



ISSN: 0067-2904

Image Signal Decomposition Using Polynomial Representation with Hybrid Lossy and Non-Lossy Coding Scheme

Zainab J. Ahmed^{1*}, Loay E. George², Raad Ahmed Hadi³

¹Department of Biology Science, College of Science, University of Baghdad, Baghdad, Iraq

²University of Information Technology and Communications, Baghdad, Iraq

³Al-Iraqia University, Baghdad, Iraq

Received: 2/2/2022

Accepted: 29/5/2022

Published: 30/10/2022

Abstract

This article presents a polynomial-based image compression scheme, which consists of using the color model (YUV) to represent color contents and using two-dimensional polynomial coding (first-order) with variable block size according to correlation between neighbor pixels. The residual part of the polynomial for all bands is analyzed into two parts, most important (big) part, and least important (small) parts. Due to the significant subjective importance of the big group; lossless compression (based on Run-Length spatial coding) is used to represent it. Furthermore, a lossy compression system scheme is utilized to approximately represent the small group; it is based on an error-limited adaptive coding system and using the transform coding scheme (discrete cosine transform or bi-orthogonal transform). Experimentally, the developed system has achieved high compression ratios with acceptable quality for color images. The performance results are comparable to those introduced in recent studies; the accomplishment of the introduced image compression system was analyzed and compared with the performance of the JPEG standard. The results of the developed system show better performance than that of the JPEG standard.

Keywords: Polynomial Representation, Image Compression, Discrete Cosine Transform, Run Length Coding, Quantization.

تحليل إشارة الصورة باستخدام تمثيل متعدد الحدود مع هيكل ترميز مختلط من فقدان البيانات وعدم فقدانها

زينب جواد احمد^{1*}, لؤي ادور جورج², رعد احمد هادي³

¹قسم علوم الحياة، كلية العلوم، جامعة بغداد، بغداد، العراق.

²جامعة تكنولوجيا المعلومات والاتصالات، بغداد، العراق.

³الجامعة العراقية، بغداد، العراق.

الخلاصة

تقدم هذه المقالة مخطط لضغط الصور بالاعتماد على متعدد الحدود، والذي يتكون من النموذج اللوني (YUV) لتمثيل محتويات اللون واستخدام متعددة الحدود ثنائي الأبعاد (من الدرجة الأولى) ذو حجم كتلة متغير والمعتمد على العلاقة بين وحدات البكسل المجاورة. يتم تحليل الجزء المتبقي من متعددة الحدود لكل الحزم إلى جزأين، الجزء الأكثر أهمية (الكبير) والأجزاء الأقل أهمية (الصغير). بسبب الأهمية للجزء الكبير؛ يستخدم الضغط بدون خسارة (على أساس Run-Length spatial coding) لتمثيله. علاوة على ذلك، يتم استخدام نظام الضغط مع الخسارة لتمثيل جزء الصغير؛ بالاعتماد على تحديد نسبة الخطأ و نظام التحويل (تحويل جيب التمام المنفصل أو التحويل المتعامد الثنائي). اظهرت النتائج ان النظام المطور حقق نسب ضغط عالية بجودة مقبولة للصور الملونة. تم مقارنة نتائج نظام الضغط المقترح مع اخر الدراسات الحديثة؛ تم تحليل إنجاز نظام ضغط الصور ومقارنته بأداء معيار JPEG. نتائج النظام المطور تظهر أداء أفضل من معيار JPEG.

1. Introduction

Image coding technology is an operation of converting a collection of characters like symbols, letters, punctuation, and numbers to a specific form to better meet demands of sending and storage. It reduces size of uncompressed image to as small as possible while keeping goodness of rebuilt image as close as original image. Encoded multimedia files are placed in an effective form named a compressed format [1, 2]. The coded file should be like the original file, however preferably a small file. The encoded file returns to the original uncompressed file by using the decoding operation. These two processes (encoding and decoding) are utilized in data communication, networking applications, and storage, to simplify the transmission operation [3].

Image compression mechanism can be divided into two parts, lossy compression technology, and lossless compression technology. In some subjects, like medical diagnosis, the user may require to rebuild the image for the original image. This kind of image compression is named lossless image compression. In other subjects, like movies and television, some data is permitted to be wasted. This kind of image compression is named lossy image compression [4]. The encoding process can be carried out in the spatial domain, or it may need to be converted to another non-spatial domain. Spatial domain coding saves conversion costs [5]. However, in the non-spatial domain (frequency), transform-based technology is based on transforming the input image into another less relevant form. Transformation does not cause data compression, but compression comes from processing and component quantization. The purpose of applying transformation is to distribute image energy in a shape that is additionally appropriate for the next efficient encoding steps. Examples of these transformations are discrete cosine transform (DCT) and discrete wave transform (DWT) [6].

Storing and transferring images has become a more difficult job in multimedia applications. Therefore, an efficient and reliable image compression system is needed to minimize the requirements for storage media and transmission bandwidth. The demand for very powerful and robust technology for compressing such images continues to grow. The remainder of this paper is organized as: Section 2 includes a literature review. Section 3 includes suggested method with an explanation in details of all involved stages in the compression module; performance results of the system are discussed in Section 4; finally, Section 5 contains conclusions.

2. Literature Review

Some studies have used this type of transformation. Among these recent studies:

D. S. S. Satyanarayana et al. 2019 [7] achieved a baseline JPEG standard with encoding and decoding of grayscale images in JPEG. The first step of encoding is to divide the input image in $8 * 8$ blocks into sub-images, perform DCT on these sub-images, and then divide the resulting matrix by the quantization matrix. The algorithm ends up making the data one-dimensional data, which is filled in with zigzag encoding and made up of arithmetic encoding or Huffman encoding. The article has introduced the influence of the coefficients on the restored image and the difference between the compression ratios. The result describes a high compression rate and allows the user to choose the quality of the output image.

S. Janaa and S. Mandal 2020[8] used both wavelet transform and quad-tree decomposition technology for better image compression performance. The discrete wavelet transform (DWT) decomposes an image into a hierarchical sub band system, then the quad-tree decomposition technology is applied to the image generated by wavelet decomposition, and the image is divided into blocks containing image information and other details according to the given threshold, and then entropy coding technology is used to finally compress the image. DWT biorthogonal technology (bior6.8) provides better PSNR than other DWT biorthogonal technologies.

With the objective of image compression, C. Ding et al. 2020 [9] suggested a grayscale image compression technique dependent on changing partial differential equations. In terms of encoding, this job at first utilizes a quadtree to division the image and then codes and conveys a few pixels. Second, the image interpolation process depends on the changing partial differential equation is utilized at the decoding end to reconstruct the image, effectively eliminating the blocking effect on the decoded image. The outcomes demonstrate that this method can perform perfect results in CR and PSNR; particularly for images with larger sizes and little texture details with better keep great changes in the grayscale of the image original. The details of the block effect can be easily removed, which is of great functional value. Despite the technique suggested in this article has a particular effect on image compression, the total PSNR stays depressed and the average stage error is high.

A systematic image compression system is proposed by Z. J. Ahmed and L. E. George 2020[10], which is based on the use of transform (discrete cosine transform or biorthogonal wavelet transform (tap9/7)) and LZW compression technology. The system is used to test the color and gray images, and then transform coding is used to separate the color and gray sub band. Perform a quantization and LZW encoding to encode the image. The obtained compression outcomes show good efficiency in increasing the compression rate and preserving the fidelity level at a suitable level.

Also, Sh. Othman et al. 2021 [11] suggested an efficient lossy compression method that uses adaptive block polynomial curve fit coding. The principal idea of this system is to decrease the number of data items to a few coefficients in the image block. The coefficient matrix in all blocks is represented with a first-order or second-order two-dimensional polynomial and variable block size of the image. The threshold is used to dynamically specify the polynomial order and the size of the coding block. The encoding situation is recognized using the two-digit prefix code. To efficiently store the coefficients matrix, a uniform quantization is implemented to these coefficients. The suggested system is implemented into two various kinds of images (gray and color). The investigational outcomes that the proposed system progress the PSNR and CR, minimum mean square error, and perfect structural similarity.

B. A. Sultan and Loay E. George 2021 [12] proposed an uncomplicated and efficient color image compression system by applying color image decomposition. Among them, the color band of the RGB image is decomposed into YUV color model, and the pixel value (amplitude) in all bands is converted into two parts; it is the most important and less important. Depending on the importance of the most important value affected by any simple modulation, a method by bit plane cutting (BP), Delta pulse code modulation, and adaptive quadtree is suggested. The partition comes next to an adaptive shift encoder. A lossy compression system was introduced to handle the least important value, which depends on an adaptive, error-limited coding structure, and utilizes a DCT compression. The results show that its achievement is equivalent to or preferable to the JPEG standard.

3. The Proposed Image Compression System

Purpose of this article is to improve image compression system using adaptive polynomial encoding supported by a high-efficiency entropy encoder to improve the performance of traditional polynomial encoding. A traditional YUV-based color polynomial coding technology uses different block sizes, which vary according to the embedded correlation between image bands while using a combination of symbol coding techniques. Each residual image pixel is divided into two values; most important (big) and least important (small). Consistent with the importance of the big value affected by any easy adjustment that occurs, an adaptive lossless image compression system dependent on run-length has been proposed. Further, lossy compression systems are designed to handle small values based on finite errors, transform coding (DCT or bi-orthogonal wavelet). Then all residue parts and polynomial coefficients are compressed by LZW coding. So the main contribution of this job is to progress the image compression system using polynomial coding and residual image decomposition. The proposed system has many successive main steps; Figure 1 shows the layout of the system.

Step 1 (loading image): Load the non-compressed original BMP image and extracts the three-band matrices: red (R), Green (G), and Blue (B).

Step 2 (color converter): Digital images are converted from RGB color systems correlated to non-correlated color space components (YUV) to generate three-band matrices Y, U, and V.

Step 3 (Polynomial coding): It is a modern compression technology that supports the idea of the image modeling, which can successfully eliminate the spatial inter-pixel redundancy that exists within the image. The essential idea of this coding is to use a mathematical model to explain each non-overlapping sub-block, with a small number of coefficients (residues) with small errors. Figure 2 shows the polynomial part of color Lena image; Figures 3 and 4 show the residual (Original-Polynomial) part data of the color Lena image for each band.

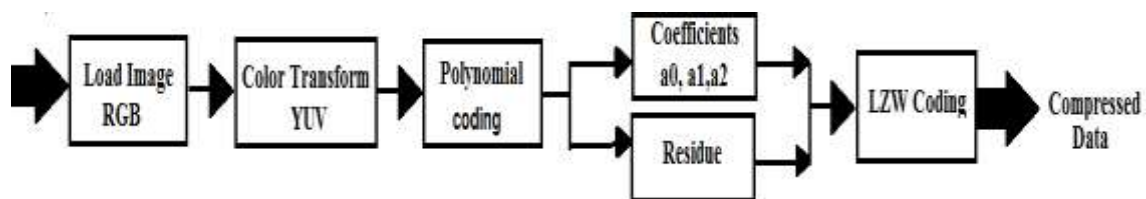


Figure 1: The main layout of the introduced system

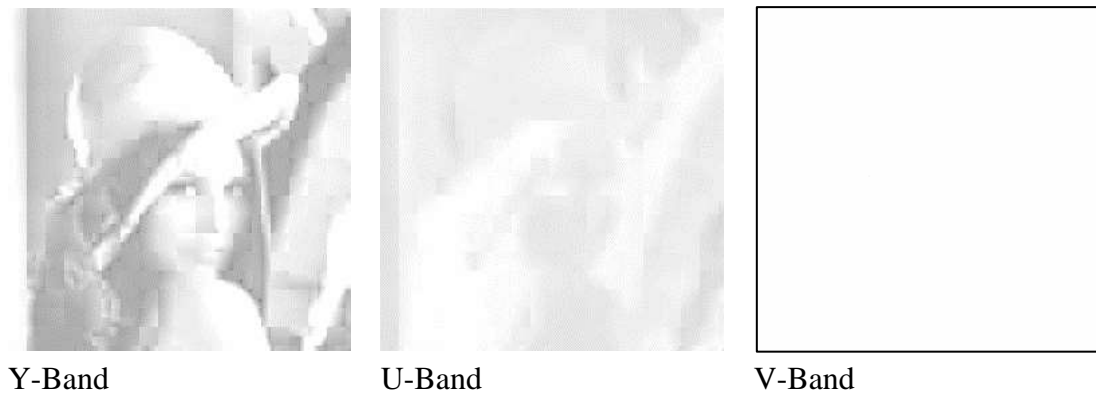


Figure 2: The polynomial part of color Lena image

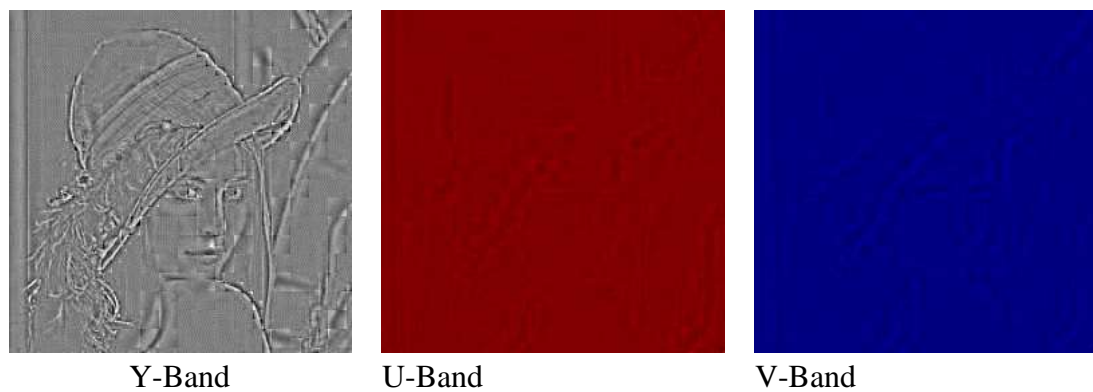


Figure 3: The residual of color Lena image

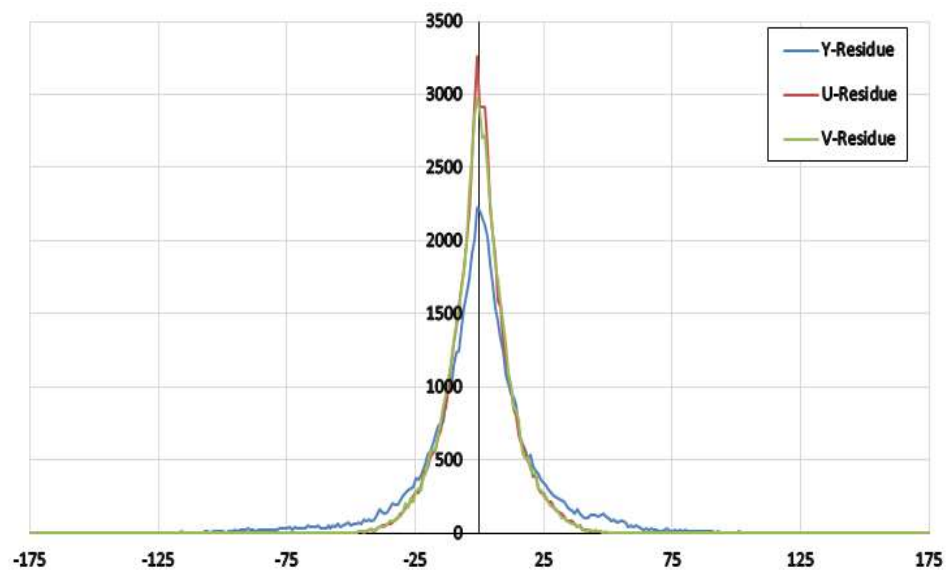


Figure 4: The histogram of residual color Lena image

The use of several blocks sizes to directly apply the polynomial coding technologies for each band. Here, the original approach contains the following sub-steps [13, 14]:

I. The bands are divided into blocks (16 x 16) pixels, and a polynomial approach is performed in each block in equations (1, 2, 3, 4).

$$a_0 = \frac{\sum_{y=0}^{L-1} \sum_{x=0}^{L-1} G(x,y)}{L^2} \tag{1}$$

$$a_1 = \frac{\sum_{y=0}^{L-1} \sum_{x=0}^{L-1} (x-x_c) \times G(x,y)}{\sum_{y=0}^{L-1} \sum_{x=0}^{L-1} (x-x_c)^2} \quad (2)$$

$$a_2 = \frac{\sum_{y=0}^{L-1} \sum_{x=0}^{L-1} (y-y_c) \times G(x,y)}{\sum_{y=0}^{L-1} \sum_{x=0}^{L-1} (y-y_c)^2} \quad (3)$$

Where $G(x,y)$ is the original value of the image element, and L is the width and height of the block.

$$\text{and, } X_c = Y_c = \frac{L-1}{2} \quad (4)$$

II. The uniform scalar quantization is used to quantize the specified polynomial coefficients to obtain compression gain.

$$a'_0 = \text{round} \left(\frac{a_0}{q_a} \right) \rightarrow \widetilde{a}_0 = a'_0 \times q_a \quad (5)$$

$$a'_1 = \text{round} \left(\frac{a_1}{q_a} \right) \rightarrow \widetilde{a}_1 = a'_1 \times q_a \quad (6)$$

$$a'_2 = \text{round} \left(\frac{a_2}{q_a} \right) \rightarrow \widetilde{a}_2 = a'_2 \times q_a \quad (7)$$

Where q_a is a polynomial quantification value.

III. Use the unquantized polynomial coefficients represented by each coding block to create a predicted image \widetilde{G} :

$$\widetilde{G} = \widetilde{a}_0 + \widetilde{a}_1 (x - x_c) + \widetilde{a}_2 (y - y_c) \quad (8)$$

Find the residual as the difference between the original image band and the predicted band created by the previous steps.

$$\text{Res}(x,y) = G(x,y) - \widetilde{G}(x,y) \quad (9)$$

Where $\text{Res}(x,y)$ is the residual value at (x,y)

IV. Compute (Chi) value that represents mean absolute error for residual of the tested block.

After that:

If $\text{Chi} > \text{threshold}$ value then applies quadtree by dividing the block size in half (8x8) and the polynomial is repeated with this new block size.

Else If $\text{Chi} < \text{threshold}$ then remain the block with no change.

Repeat the previous steps until the minimum block size (2×2) is reached. The four-tree encoding will be used to determine if a block needs to be segmented, depending on whether the Chi value generated for the block is 16x16, 8x8, or 4x4 in size.

Step 4 (LZW encoder): It is an uncomplicated, lossless compression algorithm, and depends on a dictionary. Thus, algorithm scans the file and finds data strings that appear multiple times in the file [15]. This algorithm constructs a data dictionary (string table) for the data that appears in the uncompressed data stream [16, 17]. The residuals and coefficients go through several steps; each step will be explained in sections (3.1) and (3.2).

3.1 Residue Component

Figure 5 shows the processing steps applied of residue. By using equations (10, 11) to decompose each residual value into two values: (i) large (most important) and (ii) small (least important). The large component of each frequency band is a lossless big encoder, and the small component of each frequency band is a lossy small encoder. Each basic processing step of the large and small numbers will be explained in details in two parts.

$$\text{Img}_{\text{Big}}(x,y) = \text{round} \left(\frac{\text{Res}(x,y)}{\text{Stp}} \right) \quad (10)$$

$$\text{Img}_{\text{Small}}(x,y) = \text{Res}(x,y) - \text{Img}_{\text{Big}}(x,y) \times \text{Stp} \quad (11)$$

Where: Res = refers to the residual component.

Stp = is the cut-off point value. The range of division values [32, 40, 48, 54, 64] is a very important parameter.

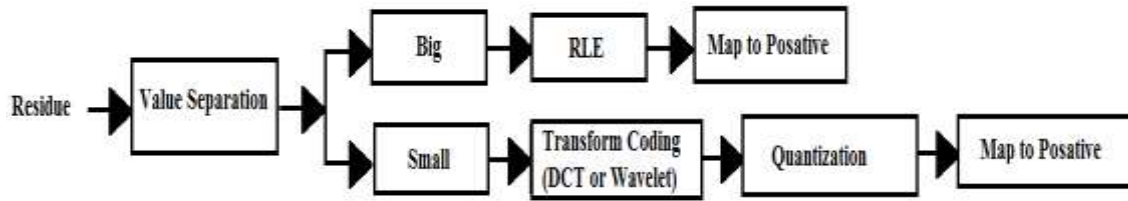


Figure 5: The residue processing steps

3.1.1 Lossless big encoder (Big component encoder)

In this section, the big component encoder has received a two-dimensional matrix of the large component of the residual for each band, as shown in Figure 5. The large encoder consists of the following processes:

I. **Run-length encoding (RLE):** It is a lossless data compression method. It is a simple shape used to compress data in which the data run is saved as a single data value and count, instead of running as a raw [18]. It calculates the value of runs and its repetition time as a run and repetition pair (I, N), where I means to run and N means repetition [19]. Figure 6 shows the structure of RLE. The RLE algorithm counts the number of repeated executions, and then sends the values to map to positive.



Figure 6: The structure of RLE

II. **Map to positive:** The mapping process needs to convert negative values to positive values to remove the negative signs, and at the same time avoid the complexity of encoding when storing these numbers. This is done by assigning all negative values to positive odd numbers and positive values to even numbers. The number of mappings is obtained by applying the following equation [19, 20]:

$$\text{Big}_{\text{map}}(x, y) = \begin{cases} 2 \times \text{Big} & \text{if } \text{Big} \geq 0 \\ (|\text{Big}| \times 2) - 1 & \text{if } \text{Big} < 0 \end{cases} \quad (12)$$

3.1.2 Lossy small encoder (Small component encoder)

In this section, the small component encoder has received a two-dimensional matrix of the small component of the residual for each band, as shown in Figure 5. The small component encoder consists of the following processes:

I. **Transform coding:** The transformation is used to map the signal from one domain representation to another (i.e.; from the time domain to frequency domain) [21]. The transformation methods used are DCT and wavelet transforms (tap9 / 7). One of these transformation methods is used to perform transformations on the small data components of the residuals.

A. DCT transforms: This transform is in control of partitioning the small two-dimensional matrix into non-overlapping blocks of size $n(n \times n)$ to perform the following block processing. For all block inputs, the well-known DCT transform is applied [22].

B. Bi-orthogonal Wavelet transforms (tap-9/7): Another very popular technique in image compression is discrete wavelet transform. In wave transform coding, the image is divided

into four sub bands, and each sub band can be easily coded separately. The tap 9/7 wavelet filters are implemented to sub bands individually [23].

II. Quantization: It is a lossy compression process that is achieved by significantly reducing the number of values. When the number of discrete symbols in a certain series decreases, the series becomes more compact [24]. The choice of the type of quantization required depends on the choice of transformation used.

A. Quantization in case of DCT transform: Non-uniform scalar quantization is applied according to [25] :

$$DCT_Q(x, y) = \text{round} \left(\frac{DCT(x, y)}{Q_s} \right) \quad (13)$$

Where Q_s is calculated by the following equation:

$$Q_s = \begin{cases} Q_0 & \text{for DC coefficient} \\ (Q_1 * (1 + \alpha(u, v))) & \text{for AC coefficients} \end{cases} \quad (14)$$

DC is the Detailed Coefficient, and AC is the Accumulative Coefficients.

B. Quantization in case of Wavelet transform: The quantized values of all wavelet coefficients are applied by the following equation [10, 26]:

$$Q_s = \begin{cases} Q\alpha^{N_{\text{pass}}-1} & \text{for LH, HL in } n^{\text{th}} \text{ Level} \\ Q\beta\alpha^{N_{\text{pass}}-1} & \text{for HH in } n^{\text{th}} \text{ Level} \end{cases} \quad (15)$$

Where N_{pass} is the wavelet level Number, α is the descending speed parameter, β is Beta multiplication:

$$\text{Wave}_Q(x, y) = \text{round} \left(\frac{\text{Wave}(x, y)}{Q_s} \right) \quad (16)$$

Where Wave () is the wavelet transform coefficient matrix, and WaveQ () is your quantization index array.

III. Map to positive: The value produced by the quantization process must be converted to a positive value. Equation (12) is used to apply it.

3.2 Encoding of a_0 , a_1 , a_2 Coefficients

The polynomial quantized coefficients go through several steps before being encoded. The steps of the polynomial coefficient explain in Figure 7. These steps are summarized below:

I. Differential Pulse Code Modulation (DPCM): DPCM means to predict the next coefficient value based on the previous coefficient value and then encode the difference between the predicted value and the actual value. This article only uses DPCM; to a_0 , decrease the number of bits desires to perform it by executing the following equation [25-27]:

$$a_0'[x] = a_0[x] - a_0[x - 1] \quad (17)$$

$a_0 []$ is array of a_0 coefficients, $a_0 []$ is array of a_0 coefficients after DPCM and $x=1 \dots a_0$ length.

II. Map to Positive: The value coefficient of each element must be always positive mapping. Equation (12) is used to apply the mapping step.

III. Histogram Packing: Due to the long tail of the mapped item values histogram. This long tail is caused by the existence of large element values, although the probability of it occurring is small. When entropy encoding is applied, this long tail can result in a significant reduction in compression gain. Therefore, it is appropriate to apply compaction in the values of the histogram. Then the element value of the positive buffer is remapped according to the packed histogram element.

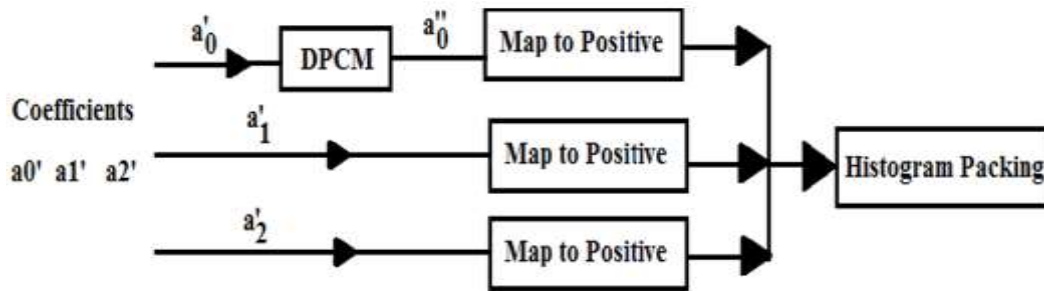


Figure 7: The processing steps of polynomial coefficients

Finally, use LZW as an effective entropy encoder to further encode the result. LZW processes all frequency bands together to increase the compression gain and encodes the information of the coefficient and residue (big and small) together once.

For decoding module, it is composed of same procedure that was submitted in the coding system but proposed in reverse order. The decoding system is responsible for the reconstruction of images extracted. Therefore, the results of decoding system are recombined to reconstruct the new Y, U, and V, and then they are converted into RGB color model to reconstruct the decompressed (RGB) images.

4. Experimental Results

The C# programming language was used to implement the proposed system. The proposed system was implemented using different test sets, and the tests are applied on five known image samples, the samples attributes are size=256x256 pixels, and color depth = 24). Image compression must achieve compatibility between image fidelity and compression ratio, thus to evaluate the variation between the original and reconstructed image, the error metric (that is, the peak signal measured in dB vs. PSNR noise). In addition to this fidelity metric, some system performance descriptions use complementary metrics, using compression ratio (CR) and compression gain (CG).

Table 1 shows the considered dominant parameters, these values have been tested in detail selection. Table 2 shows the best CR, PSNR, CG, and reconstructed color image samples in the case of DCT and wavelet transform. Table 3 explains the system parameters achieved by the proposed system when performed to Lena colors; all parameter is checked to show its effect by changing its value and keeping the other values of the coefficients constant. Table 3 shows that increasing the quantization factor (N_{pass} , q_0 , q_1 , α , β) in DCT or Wavelet will increase CR and decrease the PSNR of the restored image. Increasing the polynomial quantization step size (q_a) will result in a small improvement in CR.

Figure 8 shows the effect of Stp on CR and PSNR using Lena's test image, the test is performed in the case of DCT ($q_a = 4$, $q_0 = 17$, $q_1 = 4$, $\alpha = 0.5$, $Blk = 8$) and ($N_{\text{pass}} = 2$, $q_a = 4$, $q_0 = 9$, $q_1 = 10$, $\alpha = 1$, $\beta = 1.4$) in the case of wavelet transformation. From this figure, when the value of $Stp = 64$ in the two types of transformations affects the significant increase in compression ratio, the increase in PSNR of the restored image is within a very acceptable level. When the value of Stp is less than 64, we notice the increase in CR and PSNR, but when the value of Stp is greater than 64, the influence of the small component begins to overwhelm the influence of the large component, forming an inverse relationship. , that is, the PSNR and CR of the decompressed image. There is a significant difference, leading to a significant increase in CR and a notable decrease in PSNR, mostly when $Stp = 128$.







Figure 9 shows the impact of the block size (Blk) on DCT. The increase in the size of the block leads to an increase in CR and a reduction in PSNR to an acceptable level. The results

obtained by the suggested image compression system are compared to JPEG in Table 4 using the Lena image. The results showed a clear improvement in the compression ratio.

Table 1: The values of the control parameters







DCT		Wavelet	
Parameter	Range	Parameter	Range
Stp	[32, 40,48, 54,64]	Stp	[32, 40,48, 54,64]
q _a	[2,2.5...,5]	q _a	[2,2.5...,5]
q ₀	[10, 11,...,20]	N _{pass}	[2,3,4,5]
q ₁	[1,2,...,10]	q ₀	[1, 2,...,15]
Alpha (α)	[0.1, 0.2,...,1]	q ₁	[5,6,...,20]
Block size (Blk)	[8,16]	Alpha(α)	[0.5, 1,...,4]
		Beta(β)	[1,1.1,...1.9]

Table 2: The best CR, PSNR, CG, and the reconstructed color image samples DCT

Image Name	Original Image	CR	CG	PSNR	Decompressed Image
Lena		24.1355	95.8567	32.0095	
Barbara		22.8675	95.6269	32.5694	
Pepper		20.0815	95.0203	32.2279	

Jungle		17.7493	94.3659	32.7714	
Girl		24.5607	95.9285	32.0951	

A. Wavelet

Image Name	Original Image	CR	CG	PSNR	Decompressed Image
Lena		26.4152	96.2143	32.0203	
Barbara		22.9254	95.6380	32.2057	
Pepper		20.0477	95.0119	32.1278	



Jungle		18.2857	94.5313	32.1213	
	Girl		27.3143	96.3389	32.1724

Table 3: Test results for color Lena image

A. DCT

q_a	q_0	q_1	A	CR	CG	PSNR
2	17	4	0.5	23.01662	95.65531	31.97347
2.5	17	4	0.5	23.35844	95.71889	31.99896
3	17	4	0.5	23.24246	95.69753	31.90768
3.5	17	4	0.5	23.5177	95.74788	31.89523
4	17	4	0.5	24.13553	95.85673	32.00948
4.5	17	4	0.5	24.15332	95.85978	31.96371
5	17	4	0.5	24.32063	95.88826	31.99795
4	10	4	0.5	23.97074	95.82825	32.11481
4	11	4	0.5	24.00586	95.83435	32.10068
4	12	4	0.5	24.04991	95.84198	32.0813
4	13	4	0.5	24.07052	95.84554	32.06939
4	14	4	0.5	24.07347	95.84605	32.06087
4	15	4	0.5	24.07936	95.84707	32.04473
4	16	4	0.5	24.10002	95.85063	32.0367
4	17	4	0.5	24.13553	95.85673	32.00948
4	18	4	0.5	24.15628	95.86029	31.99281
4	19	4	0.5	24.16519	95.86182	31.97939
4	20	4	0.5	24.16816	95.86233	31.95165
4	17	1	0.5	16.15779	93.81104	39.53591
4	17	2	0.5	19.83935	94.95951	36.34098
4	17	3	0.5	22.19804	95.4951	33.87438
4	17	4	0.5	24.13553	95.85673	32.00948
4	17	5	0.5	25.85247	96.1319	30.60992
4	17	6	0.5	27.39799	96.3501	29.57644
4	17	7	0.5	28.81125	96.52913	28.74117
4	17	8	0.5	30.08539	96.67613	28.08038
4	17	9	0.5	31.22745	96.79769	27.53807
4	17	10	0.5	32.28905	96.90297	27.11043

4	17	4	0.1	19.16631	94.78251	37.77769
4	17	4	0.2	20.75676	95.18229	35.98407
4	17	4	0.3	22.01904	95.45848	34.42823
4	17	4	0.4	23.10589	95.6721	33.09615
4	17	4	0.5	24.13553	95.85673	32.00948
4	17	4	0.6	25.05837	96.00932	31.09826
4	17	4	0.7	25.96513	96.14868	30.35303
4	17	4	0.8	26.82603	96.27228	29.72267
4	17	4	0.9	27.62124	96.3796	29.20054
4	17	4	1	28.37056	96.47522	28.74218

B. Wavelet

q_a	N_{Pass}	q_0	q_1	α	β	CR	CG	PSNR
2	2	9	10	1	1.4	25.32629	96.05153	31.9981
2.5	2	9	10	1	1.4	25.69032	96.10748	31.95379
3	2	9	10	1	1.4	25.60333	96.09426	32.01701
3.5	2	9	10	1	1.4	25.93089	96.1436	31.9893
4	2	9	10	1	1.4	26.41516	96.21429	32.02026
4.5	2	9	10	1	1.4	26.51848	96.22904	31.96075
5	2	9	10	1	1.4	26.64426	96.24685	31.01047
4	2	9	10	1	1.4	26.41516	96.21429	32.02026
4	3	9	10	1	1.4	26.64787	96.24736	31.04305
4	4	9	10	1	1.4	26.68042	96.25193	30.43817
4	5	9	10	1	1.4	26.68766	96.25295	30.32346
4	2	1	10	1	1.4	24.16222	95.86131	34.04421
4	2	2	10	1	1.4	24.88709	95.98185	33.9461
4	2	3	10	1	1.4	25.32955	96.05204	33.76645
4	2	4	10	1	1.4	25.61668	96.09629	33.57133
4	2	5	10	1	1.4	25.87288	96.13495	33.28802
4	2	6	10	1	1.4	26.04424	96.16038	33.00183
4	2	7	10	1	1.4	26.19345	96.18225	32.68027
4	2	8	10	1	1.4	26.31968	96.20056	32.35399
4	2	9	10	1	1.4	26.41516	96.21429	32.02026
4	2	10	10	1	1.4	26.52206	96.22955	31.6543
4	2	11	10	1	1.4	26.62261	96.24379	31.24548
4	2	12	10	1	1.4	26.6768	96.25142	30.88075
4	2	13	10	1	1.4	26.73847	96.26007	30.53271
4	2	14	10	1	1.4	26.82237	96.27177	30.16788
4	2	15	10	1	1.4	26.87738	96.2794	29.82113
4	2	9	5	1	1.4	21.16796	95.27588	34.15606
4	2	9	6	1	1.4	22.45152	95.54596	33.71948
4	2	9	7	1	1.4	23.5939	95.76162	33.28617
4	2	9	8	1	1.4	24.63451	95.94065	32.84009
4	2	9	9	1	1.4	25.58002	96.0907	32.42439
4	2	9	10	1	1.4	26.41516	96.21429	32.02026
4	2	9	11	1	1.4	27.23857	96.32874	31.62311
4	2	9	12	1	1.4	27.94712	96.42181	31.25586
4	2	9	13	1	1.4	28.59753	96.50319	30.8816

4	2	9	14	1	1.4	29.21367	96.57694	30.51093
4	2	9	15	1	1.4	29.84335	96.64917	30.15053
4	2	9	16	1	1.4	30.46769	96.71783	29.8187
4	2	9	17	1	1.4	31.02051	96.77633	29.47218
4	2	9	18	1	1.4	31.57347	96.83278	29.14874
4	2	9	19	1	1.4	32.1465	96.88924	28.84677
4	2	9	20	1	1.4	32.59416	96.93197	28.54476
4	2	9	10	0.5	1.4	24.82738	95.97219	33.08637
4	2	9	10	1	1.4	26.41516	96.21429	32.02026
4	2	9	10	1.5	1.4	27.28014	96.33433	30.87051
4	2	9	10	2	1.4	27.84816	96.4091	29.76562
4	2	9	10	2.5	1.4	28.34193	96.47166	28.74252
4	2	9	10	3	1.4	28.73966	96.52049	27.89776
4	2	9	10	3.5	1.4	29.09263	96.5627	27.23814
4	2	9	10	4	1.4	29.42793	96.60187	26.74698
4	2	9	10	1	1	25.66684	96.10392	32.21421
4	2	9	10	1	1.1	25.87628	96.13546	32.16771
4	2	9	10	1	1.2	26.06842	96.16394	32.11773
4	2	9	10	1	1.3	26.26343	96.19242	32.07088
4	2	9	10	1	1.4	26.41516	96.21429	32.02026
4	2	9	10	1	1.5	26.57583	96.23718	31.97926
4	2	9	10	1	1.6	26.71667	96.25702	31.93429
4	2	9	10	1	1.7	26.82603	96.27228	31.90007
4	2	9	10	1	1.8	26.92154	96.2855	31.85884
4	2	9	10	1	1.9	26.99547	96.29567	31.80222

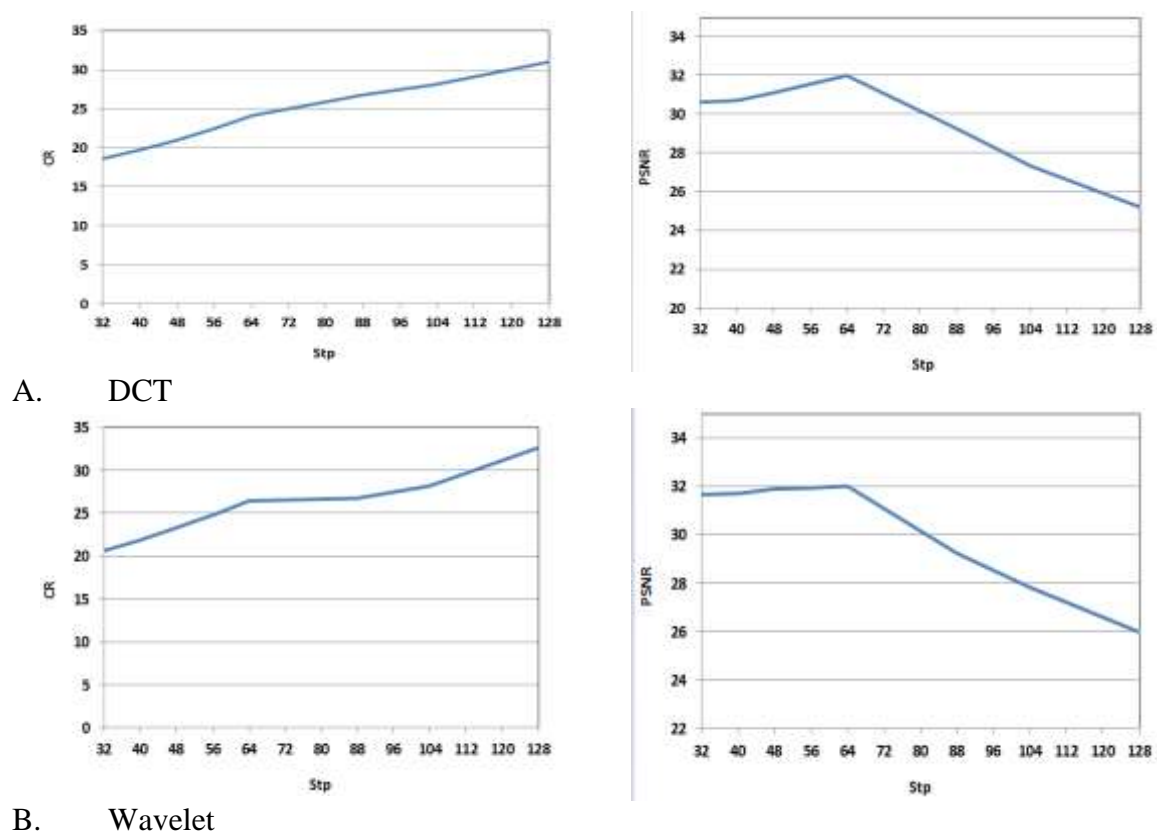


Figure 8: The effect Step on CR and PSNR

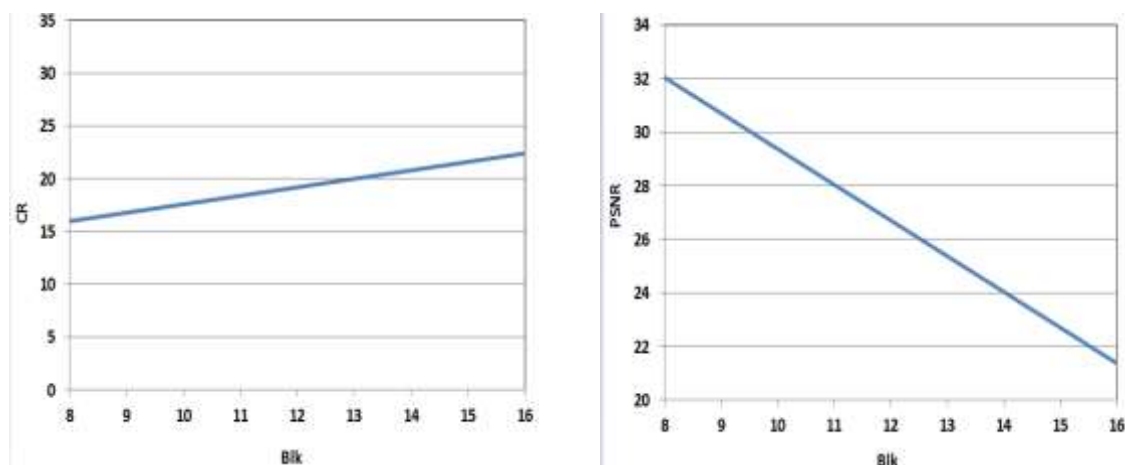


Figure 9: The effect of block size on CR and PSNR in DCT

Table 4: The proposed system with JPEG

Method	Quality	CR	PSNR (dB)
JPEG standard	20%	23.4	30.77
	40%	21.6	31.60
	80%	10.3	34.56
System with DCT		24.1355	32.0095
System with Wavelet		26.4152	32.02

4.1 Comparative Evaluation

To show the robustness of the proposed method, a comparative evaluation was carried out in this work in Table 5. In this evaluation, five different methods were compared to obtain five evaluation scores. It compares the simulation results of the proposed methods; the CR for implementing the proposed method in Lena's image is 26.4152, which is higher than the CR of other algorithms compared with the accepted PSNR of 32 db.

Table 5: Comparison with previous studies

Articles	Coding method	CR	PSNR
[11]	Adaptive encoding. An easy coding technique that does not need to be saved coding tables	6.997	31.0126
[20]	Adaptive shift coding	11.303	32.364
[27]	Arithmetic coding technique	20.4928	36.4685
[12]	Adaptive shift encoder	22.20	32.42
[10]	LZW	20.18	32.02
Proposed System	LZW	26.4152	32.02

5. Conclusion

As a result of testing the suggested system, the following observations were stimulated: considering the importance of the most important value, the system is progressive (most important value needs a little number of bits after encoding because they know they will be

encoded without errors, while the least important value is processed by lossy encoding because they need a large number of bits).

The use of polynomial with the value separation of the residue part of an image had enhanced the performance of the compression (i.e.; increase the gain in compression while keeping the quality of the image). The compression performance of the image has been improved (that is, the increase of control parameters). This increases the compression ratio and a decrease in the PSNR value.

Increasing Stp (i.e., cut-off bit) will relatively reduce the number of bits. In most cases, it is necessary to encode the most important value. When the value of Stp = 64 represents very encouraging values for CR and PSNR the system is compared with the general JPEG standard in terms of compression ratio and visual quality indicators. The system is compared with known pressure methods, and the results show that the proposed method has made great progress in CR and accuracy. It is potential to improve entropy encoder methods to improve the compression ratio or use nonlinear quantization in the future.

References

- [1] Sh. M. Othman, A. E.Mohamed, Z. Nossair and M. I. El-Adawy, "Image Compression Using Polynomial Fitting," IEEE Proceedings of the 3rd International Conference on Electronics Communication and Aerospace Technology, India, pp. 344-349, 2019.
- [2] N.S. Singh and H.J. Singh, "Data Compression Techniques in Wireless Sensor Network: A Survey," International Journal of Computer Sciences and Engineering, vol.7, no.1, pp. 697 -706, 2019.
- [3] J. Hussain, A. Al-Fayadh and N. Radi, "Image Compression Techniques: A Survey in Lossless and Lossy algorithms," Neurocomputing, Elsevier, vol.300, pp. 44-69, 2018
- [4] K. Sayood, Introduction to Data Compression, 5th Edition, 2018, p.765.
- [5] M. George, M. Thomas and J. C.K, "A Methodology for Spatial Domain Image Compression Based on Hops Encoding," Procedia Technology, vol. 25, pp. 52 – 59, 2016.
- [6] H. ZainEldin, M. A. Elhosseini and H. A. Ali, "Image compression algorithms in wireless multimedia sensor networks: A survey," Ain Shams Engineering Journal, vol.6, no.2, pp. 481-490, 2015 .
- [7] D. S. S. Satyanarayana, G. R. Reddy, R.S.P. Shanmukhanath and K. Gaurav, "JPEG Image Compression Using Discrete Cosine Transform," International Journal of New Technology and Research, vol.5, no.10, pp. 52-55, 2019.
- [8] S. Janaa and S. Mandal, "DWT Based Image Compression Using Quadtree Decomposition and Huffman Encoding," Contemporary Issues in Computing, Topics in Intelligent Computing and Industry Design (ICID), vol.2, no.1, pp. 144-147, 2020.
- [9] C. Ding, Y. Chen, Z. Liu and T. Liu, "Implementation of grey image compression algorithm based on variation partial differential equation," Alexandria Engineering Journal, vol.59, pp. 2705–2712, 2020 .
- [10] Z. J. Ahmed and L. E. George, "A Comparative Study Using LZW with Wavelet or DCT for Compressing Color Images," 2020 International Conference on Advanced Science and Engineering (ICOASE), IEEE, Iraq, pp.53-58, 2020.
- [11] Sh. Othman, A. Mohamed, A. Abouali and Z. Nossair, "Lossy Compression using Adaptive Polynomial Image Encoding," Advances in Electrical and Computer Engineering, vol. 21, no.1, pp.91-98, 2021 .
- [12] B. A. Sultan and L. E. George, "Color image compression based on spatial and magnitude signal decomposition," International Journal of Electrical and Computer Engineering, vol. 11, no.5, pp. 4069-4081, 2021 .
- [13] L. E. George and B. A. Sultan, "Image Compression Based On Wavelet, Polynomial, and Quadtree," Journal of Applied Computer Science & Mathematics, vol. 5, no.2, pp.15-20, 2011.
- [14] G. Al-Khafaji and M. Bassim, "Polynomial Color Image Compression," International Journal of Engineering Trends and Technology, vol. 61, no.3, pp. 161-165, 2018.

- [15] S. A. Abu Taleb, H. M. J. Musafa, A. M. Khtoom and I. K. Gharaybih, "Improving LZW Image Compression," *European Journal of Scientific Research*, vol.44, no.3, pp.502-509, 2010.
- [16] P.Yellamma and N. Challa, "Performance Analysis of Different Data Compression Techniques On Text File," *International Journal of Engineering Research & Technology*, vol.1, no.8, pp.1-6, 2012 .
- [17] H. H Vandra, "Image Compression with LZW (Lossless) On Different Image Formats," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 3, no.3, pp. 2153-2156, 2018.
- [18] S. M. Hardi, B. Angga, M. S. Lydia, I. Jaya and J. T. Tarigan, "Comparative Analysis Run-Length Encoding Algorithm and Fibonacci Code Algorithm on Image Compression," the 3rd International Conference on Computing and Applied Informatics, *Journal of Physics Conference Series*, vol.1235, pp.1-6, 2019.
- [19] A. C. Bovik, *The Essential Guide to Image Processing*, USA, Academic Press, 2009, p.841.
- [20] A. A. Ibrahim, L. E. George and E. Kh. Hassan, "Color Image Compression System by using Block Categorization Based on Spatial Details and DCT Followed by Improved Entropy Encoder," *Iraqi Journal of Science*, vol. 61, no.11, pp. 3127-3140, 2020.
- [21] R. H. Herrera, J. B. Tary, M. V. D. Baan and D. W. Eaton, "Body Wave Separation in the Time-Frequency Domain," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no.2, pp. 364 – 368, 2015.
- [22] D. Salomon and G. Motta, *Handbook of Data Compression*, USA, Springer, 5th Edition, 2010, p.1340.
- [23] I. Dagher, M. Saliba and R. Farah, "Combined DCT-Haar transforms for image compression," *International Journal of Imaging Systems*, vol.28, no.4, pp. 274-294, 2018.
- [24] K. Karthika and P. Damodharan, "An Efficient Image Forgery Detection Method with Dempster Shefer's Theory for Detecting Jpeg Anti-Forensics," *International Journal Of Engineering And Computer Science*, vol. 4, no.2, pp. 10539-10542, 2015 .
- [25] E. Kh. Hassan, L. E. George and F. G. Mohammed, "Color Image Compression Based on DCT, Differential Pulse Coding Modulation, and Adaptive Shift Coding," *Journal of Theoretical and Applied Information Technology*, vol.96, no.11, pp. 3160- 3171, 2018.
- [26] H. Ahmed and L. E. George, "The Use of Wavelet, DCT & Quadtree for Images Color Compression," *Iraqi Journal of Science*, vol.58, no.1C, pp. 550-561, 2017.
- [27] A. Hussain, G. K. AL-Khafaji and M. M. Siddeq, "Developed JPEG Algorithm Applied in Image Compression," 2nd International Scientific Conference of Al-Ayen University, *Materials Science and Engineering Conference Series*, vol. 928, pp. 1-17, 2020 .