decoded image. All the blocks are combined to get the pixel matrix from which image is generated. The analysed results shows that storage size of the image decreases and therefore compression ratio increases.

3. Image Compression Techniques

There are two ways in which we can categorize image compression techniques: Lossy and Lossless image compression [52, 53]. The former technique involves the image reconstruction as a similar calculation of original data and thus some loss of data is incurred. It wins over the later in terms of achieving higher compression ratio. The techniques taken under consideration on the basis of literature includes: Discrete Cosine Transforms (DCT), Discrete Wavelet Transform (DWT), Block Truncation Coding (BTC), and Fractal Encoding. The later technique involves the recreation of original data from the compressed form. As it uses all the information of the original image while compressing the data/image, the image received after decompressing the image is exactly identical to the original image. Four techniques has been shortlisted for consideration, which are Walsh-Hadamard, Run-Length-Encoding (RLE), Burrows-Wheeler-Transform (BWT), and Huffman Encoding.

3.1. Discrete Cosine Transform

DCT [38] is widely considered as the most effective and competent coding scheme, which was presented to solve the problem of discontinuity and to achieve better performance. The transformation is discrete, factual and orthogonal which compactly transcribe the image info to frequency domain, deriving it from spatial domain [29]. The equation for 1D DCT of a sequence $\{f(x), x = 0,1,2,3 \dots, n-1\}$ with interval $0 \leq x \leq n-1$ is given as [22]:

$$C(u) = \alpha(u) \sum_{x=0}^{n-1} f(x) cos\left(\frac{(2x+1)u}{\frac{\pi}{2n}}\right), \quad u = 0,1,2,3, \dots \dots, n-1 \qquad (1)$$

The inverse of the above equation is given as:

$$f(x) = \sum_{u=0}^{n-1} \alpha(u) C(u) cos\left(\frac{(2x+1)u\pi}{2n}\right), \quad x = 0,1,2,3 \dots \dots, n-1 \qquad (2)$$

$$where\ \alpha(0) = \frac{1}{\sqrt{2}}\ and\ \alpha(u) = 1 \qquad for\ all\ u \neq 0$$

1-D DCT vectors are obtained from:

$$C_{u,x} = \left\{\sqrt{\frac{2}{n}}\ \alpha(u) cos\frac{(2x+1)u\pi}{2n}, \quad x = 0,1,2, \dots \dots, n\right\}, for \quad u = 0,1,2,3, \dots, n-1 \qquad (3)$$

DCT being a distinct transform, 2-D DCT and its inverse is obtained in two steps through succeeding application of 1-D DCT and its inverse [41].

For images, 2D DCT of an $n \times n$ image with f(x,y) is give as [42]:

$$C(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} f(x,y) cos\left(\frac{(2x+1)u\pi}{2n}\right) cos\left(\frac{(2y+1)u\pi}{2n}\right) \qquad (4)$$

The inverse DCT is given by:

$$f(x,y) = \alpha(u)\alpha(v) \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} C(u,v) cos\left(\frac{(2x+1)u\pi}{2n}\right) cos\left(\frac{(2y+1)v\pi}{2n}\right) \qquad (5)$$

$$here\ \alpha(u) = \begin{cases} \sqrt{\frac{1}{n}} & for\ u = 0 \\ \sqrt{\frac{2}{n}} & for\ u = 1,2,3 \dots, n-1 \end{cases}$$

The transform matrix is given by:

$$C_{u,v,x,y} = \left\{\frac{2}{n}\ \alpha(u)\alpha(v) cos\left(\frac{(2x+1)u\pi}{2n}\right) cos\left(\frac{(2y+1)v\pi}{2n}\right), x, y, u, v = 0,1,2, \dots, n-1\right\} \qquad (6)$$

The coefficients of DC matrix with zero frequency are called DC coefficients and others are also addressed as AC coefficients, reflecting dissimilarities in the value of grey level [27-28].

3.2. Discrete Wavelet Transform

The discrete wavelet transform [13, 37, and 61] is a powerful technique in image processing, where the wavelet convert the image into a series of wavelets which are further stored more efficiently as compared to pixel blocks. For one dimensional, signals are split into two parts: high and low frequencies. The low pass and high pass filter of DWT is expressed as [40, 59-62]:

$$H(w) = \sum_k h_k \times e^{-jkw} \qquad (7)$$

$$G(w) = \sum_k g_k \times e^{-jkw} \qquad (8)$$

Both filters satisfy the orthogonal condition [21]:

$$|H(w)|^2 + |G(w)|^2 = 1 \qquad (9)$$

2D-DWT [16] can be obtained by applying 1D DWT row wise to produce L and H sub-bands in each row then column wise. Total four number of sub-bands (LL1, LH1, HL1, and HH1) are acquired for level 1 decomposition [23]. By repeating similar procedure in sub-band labelled LL1, we get LL2, LH2, HL2, and HH2 and so on as depicted in Figure 2.

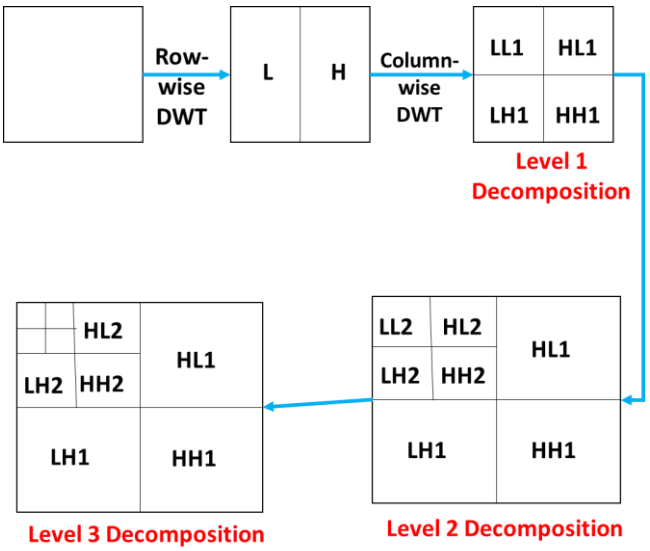

Figure 2. 2D-DWT: Three level decomposition

3.3. Block Truncation Coding

BTC compression technique was introduced in 1979 for images of grey-scale. It is often termed as moment-preserving-block-truncation since post execution, each image's first and second moments are preserved. The input image is divided into non overlapping rectangular regions where the size is assumed to be $m \times m$ for simplicity. Then, the two luminance values i.e, the mean and standard deviation are selected which represent each pixel in a block [30-31].

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (10)$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x}_i)^2} \quad (11)$$

Where x_i the value of image block's i[th] pixel while in totality, there are n number of pixels in that block. The two values, i.e., $\bar{x}$ and σ are called quantizers. Considering, $\bar{x}$ as threshold value, a 2-level bit plane is generated by comparing x_i with $\bar{x}$. A binary block, B represents the pixel values where 1 denotes that pixel whose grey level value is greater than threshold, and 0 whose grey level value is less.

$$B = \begin{cases} 1 & x_i \geq \bar{x} \\ 0 & x_i < \bar{x} \end{cases} \quad (12)$$

The image block is regenerated by converting all: 1→ H and 0→ L, expressed as:

$$H = \bar{x} + \sigma\sqrt{\frac{p}{q}} \quad (13)$$

$$L = \bar{x} - \sigma\sqrt{\frac{q}{p}} \quad (14)$$

Where p and q are the quantifications of total 0 and total 1 in compressed bit plane.

3.4. Fractal Encoding

Fractal encoding is one of the technique used for compressing the still images. A fractal code is made up of three integral parts, i.e., ranges, domains and transformations. A range is to partition the image region into portions R_k, a domain D_k – which signifies same number of image's other regions. Thereafter, there is a requirement of two transformations, for each domain-range i.e., one geometric transformation which maps domain to range, $u_k: D_k \rightarrow R_k$, and the other is affine transformation v_k, fine-tunes the value of intensity corresponding to respective range in the domain. An image is decomposed to N, non-overlapping range blocks R to M, other domain blocks D and the operator W is defined as:

$$W_x = W(\sum_{i=1}^{N} R_i) = \sum_{i=1}^{N} \omega_i (D_i) \quad (15)$$

Where $D_i = (x|D_i)$, signifying the fact that image x has a restriction upto domain D_i. ω_i denotes the mapping of transformation, to the domain block D_i on the range of block R_i. The shape and size of R_i and D_i can be dissimilar. If M (number) of block D_i is either less than, greater to or equal to N, so blocks in D_i can be overlapped or culled from parts of image.

3.5. Walsh Hadamard Transform

WHT is considered as the simplest of all transformations, that requires incrementing and deducting, and is structured on the concept of square waves / rectangular waves with peaks value of ± 1. The lowest order of transform is:

$$T_1 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (16)$$

To obtain the higher order WHT, the elements of transform matrix can be replaced by first order matrix:

$$T_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & \cdots\cdots & \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & & \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \end{bmatrix} \quad (17)$$

The general expression is [24-25]:

$$T = \frac{1}{\sqrt{N}} H_{2N} \quad (18)$$

$$where\ H_{2N} = \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix}, H_1 = 1\ and\ N = 2^n$$

The process of calculating the inverse matrix for the same is simple due to symmetry of transform matrix [26].

3.6. Run Length Encoding

RLE technique counts the occurrence of same data and stores as a single data value and single count [19, 50]. Consider a colored image that has too many long runs of red pixels and short runs of green and blue pixels. For example, while scanning any single row in an image with R representing red pixels, G representing green and B representing blue pixels.

RRRRRRGGGBBBBRRRRRRRRBBBBBGGGGGRRRRRRR

After applying run length encoding on the above row, the resultant is:

(6R) (3G) (4B) (8R) (5B) (5G) (7R).

Where 6R means 6 count of red pixels, 3G means 3 count of green pixels, 4B means 4 count of blue pixels and so on. The technique encodes only the consecutive number of same color pixels i.e., the probability of occurrence of consecutive same color is high [46, 48].

3.7. Burrows Wheeler Transform

In BWT , a presumed value T′ is produced from a text - T. It is presumed that T (text) will be concluded by an end marker "$".T′ is the resultant value of M, which is a conceptual matrix, whose rows signify the cyclic shifts of T, where F being the primary value in initial column and L is the terminal value[20]. The rows of M are in lexical order and T′ = L. The algorithm is reversible. For example, let us consider the i^{th} row of M, its terminal character L[i] presages the initial most character F[i] in T, i.e., T = L[i]F[i]……..let L[i] = c and r_i can be total number of times c has occurred in L[1,i]. The r_i^{th} row of M which is starting from c, will be M[j]. Then in the initial most column F, the character which is corresponding to L[i] can be sited at F[j] because the occurrences of c is proportional to storages of F and L. This entire procedure is known as LF mapping, where L[i] = j. The technique can be reversed as follows [17, 49]:

- Step 1: Compute an array - C [1, σ] by storing it in C[c] +1, is the first occurrence position of c in F.
- Step 2: LF mappings: LF(i) = C[L[i]] +Occ(L,L[i], i), where the total number of occurrences of c in L[1,i] is given by Occ(L,c,i).
- Step 3: Remodel T: let s = 1 for every n-1,…1, then L[s] → T[i] and LF[s] →s and set the terminal label T[n] = $.

3.8. Huffman Encoding

This is one of the oldest lossless image compression technique, developed in order to minimize the code redundancy whilst maintaining the quality of the reconstructed image. To understand the algorithm, let us consider an example, there are seven source symbols of a digital image {S1, S2, S3, S4, S5, S6, S7) with the respective probability values of {0.25, 0.25, 0.125, 0.125, 0.125, 0.0625, and 0.0625}. The process of procuring Huffman code is represented in Figure 3. The probabilities are written in decreasing order, then adding the two least probabilities and writing the result above the probability which is equal to the sum of last two digits, all other probabilities are written beneath that and thus continuing the process[33-34]. Figure 4 represents the calculation for the code word length.

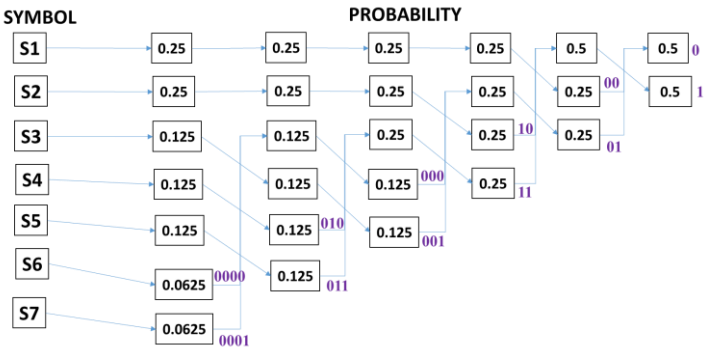

Figure 3. Huffman encoding: average word length = 2.625 bits, bit assignment process

Symbol	Probability	Code Word	Length
S1	0.25	10	2
S2	0.25	11	2
S3	0.125	1	3
S4	0.125	10	3
S5	0.125	11	3
S6	0.0625	0	4
S7	0.0625	1	4

Figure 4. Huffman encoding: code word

The average code word length is defined as ():

$$L_{avg} = \sum_{i=1}^{L} I(r_i)P(r_i) \quad (19)$$

Where I(r_i) are the total bits that represent grey level, $P(r_i)$ is the respective probability value and(r_i), i=1, 2, L represents the i^{th} grey level of an (L-grey level) image especially L-grey level [35]. Hence fore, analyzing the Figure 4 we can reach a conclusion that an average length of code word, in this example can be calculated as:

$$L_{avg} = 0.25 * 2 + 0.25 * 2 + 0.125 * 3 + 0.125 * 3 + 0.125 * 3 + 0.0625 * 4 + 0.0625 * 4 \quad (20)$$

$$L_{avg} = 2.625\ bits$$

From above example, it is evident that an average number of bits are shrunk to 2.625 bits for variable length coding. However, the algorithm is optimal, if the source of probability distribution is known prior and each of the source symbol bits are encoded in the form of integral number.

4. Performance Metrics

The performance of image compression algorithms [2] can be evaluated on the basis of parameters mentioned below:

4.1. Quality of an Image

The requirement of certain techniques in order to analyze the quality of image after the reconstruction. Better image quality is judged by minimum image distortion by the compression process. There are two ways to analyze the image after reconstruction, Qualitative and Quantitative quality measurements. Qualitative measurements are done by querying the human beings to judge and report the quality of the image by comparing it with other image with naked eye and choose the best one whereas quantitative measurements include some mathematical expression that can quantify the amount of distortion and the quality of image.

4.2. Compression Ratio

Compression ratio (CR) [32] is the ratio of the size of the file post compression, compared to the original file and is mathematically expressed in eq (21). CR is used to determine the efficiency of the compression, higher the CR better the compression [18].

$$Compression\ ratio = \frac{Compressed\ file\ size}{Original\ file\ size} \quad (21)$$

4.3. File size after compression and reconstruction

This parameter determines the size in bytes of the image after compression and the image size after the reconstruction in order to identify the loss of data in compression and reconstruction process [14]. It is measured in bytes.

4.4. Compression Speed

Speed of compression depends on compression technique adopted and the results of the parameter is dependent upon the size of memory. Lossy compression techniques [46] increases the computational complexity and storage. It is measured as ratio of compressed output file size per unit time required for compressing the file in seconds [24and is expressed mathematically in eq. (22):

$$Compression\ speed = \frac{Compressed\ file\ size}{Compression\ time} \quad (22)$$

4.5. Time required to compress and reconstruct

This parameter determines the amount of time required by a compression technique to compress and reconstruct the original file[36]. It is measured in seconds.

4.6. Compression Factor

Compression factor is the ratio of size of the original image to the size of the image after compression size. It is basically the inverse of compression ratio and is mathematically expressed in eq. (23). Lower the compression factor better the compression.

$$Compression\ factor = \frac{Original\ file\ size}{Compressed\ file\ size} \quad (23)$$

4.7. Saving Percentage

Saving percentage is the term that calculates the percentage value of the shrinkage resource file and is expressed in eq. (24):

$$Saving\ percentage = \frac{(original\ file\ size - compressed\ file\ size)}{Original\ file\ size}\ \% \quad (24)$$

5. Experimental Results & Analysis

Figure 5 and Figure 6 shows the subjective analysis of the image. Figure 5 represents the output of the images using lossy compression techniques. In the implementation part, the color image of any format (like jpeg, bmp, png) is taken into consideration and the color image is converted to the gray scale image and further image based compression algorithms are thus applied. Figure 6 indicates the resultant images after applying lossless compression techniques.

The implemented code is executed on 100 images using different compression techniques and the results achieved are shown in Table 1. It indicates the results of lossy and lossless compression algorithms on various images. The implementation is done in MATLAB. The different image compression techniques are analysed and the results are evaluated as follows:

Figure 7 shows the performance analysis of the above mentioned techniques based on compressed file size. The original size of the image is 38278 bytes, the DCT compression algorithm compresses the original image to 5096 bytes. DWT compresses the image to 777 bytes which is better than DCT. However, BTC can compress up to 6395 bytes the value of which is greater than both DCT and DWT. Fractal encoding compresses the image by just 174 bytes. Out of these 4 lossy techniques better compression is provided by Fractal encoding. However, in lossless compression algorithms best performance is shown by Huffman Encoding by compressing the original image to 215 bytes as compared to other algorithms such as Walsh Hadamard, RLE, and BWT that compresses the image to 27007, 694, and 393 bytes respectively.

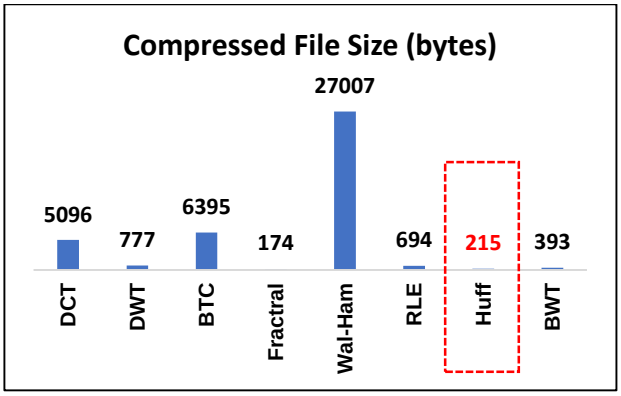

Figure 7. Performance on the basis of compressed file size offered by different image compression algorithms

Figure 8 shows the performance analysis of these algorithms on the basis of Compression Ratio. Out of lossy algorithms the best performance is shown by Fractal Encoding with compression ratio value as 0.004 whereas the algorithms such as DCT, DWT, and BTC shows the compression ratio values as 0.13, 0.02 and 0.167. In lossless algorithms the Huffman Encoding achieves the lowest value of 0.0101 where the other algorithms like Walsh Hadamard, RLE and BWT acquires the compression ratio values 0.705, 0.018, and 0.0102 respectively.

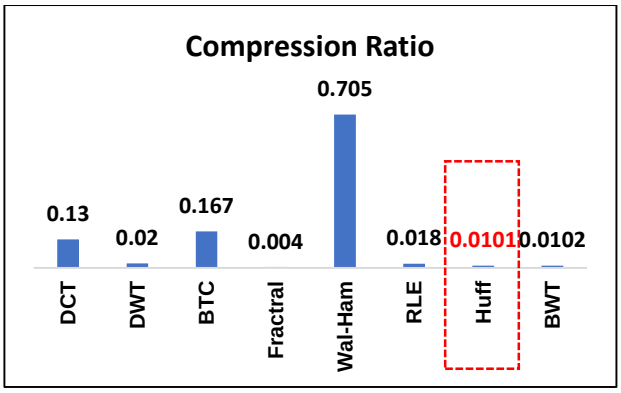

Figure 8. Performance on the basis of compressed ratio offered by different image compression algorithms

Figure 9 examines the performance of algorithms based on Compression Time parameter. In lossy techniques, BTC algorithm takes less time while compressing the image i.e., 0.04 secs whereas DCT, DWT, and Fractal Encoding takes 8.01, 0.867, 27.7 secs respectively. However, in lossless techniques BWT takes less amount of time to compress the image i.e., 0.004 secs as compared to Walsh Hadamard, RLE, and Huffman Encoding that takes 0.1, 0.83, and 0.277 secs respectively.

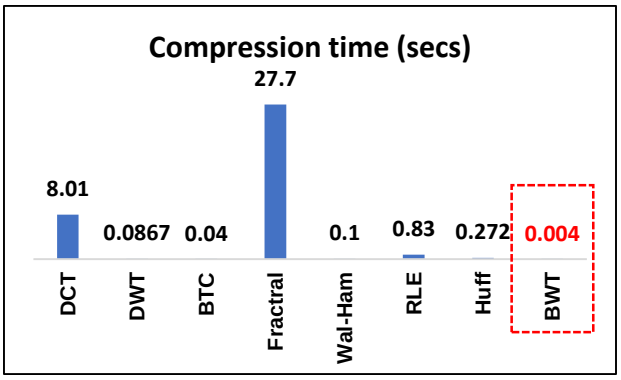

Figure 9. Performance on the basis of compressed time offered by different image compression algorithms

Figure 10 represents the performance of the compression techniques on the basis of Decompression Time. DWT takes the minimum amount of time i.e., 0.0021 secs to decompress the image where DCT takes 0.015 secs. BTC decompresses the image in 0.02 secs and Fractal Encoding takes 0.2 secs. Out of lossless techniques, the minimum amount of time required to decompress an image by an algorithm is 0.0007 secs by BWT. Walsh Hadamard takes 0.11 secs, RLE takes 6.12 secs and Huffman Encoding takes 0.06 secs.

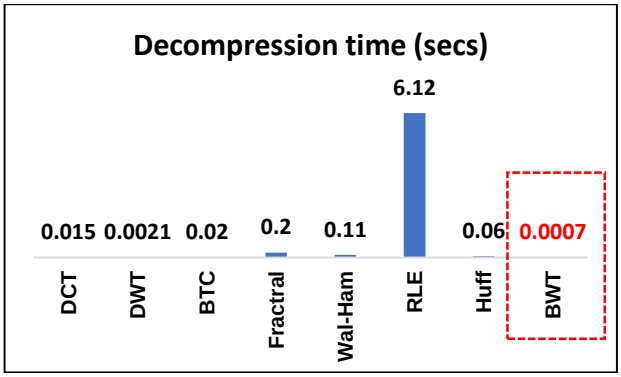

Figure 10. Performance on the basis of decompressed time offered by different image compression algorithms

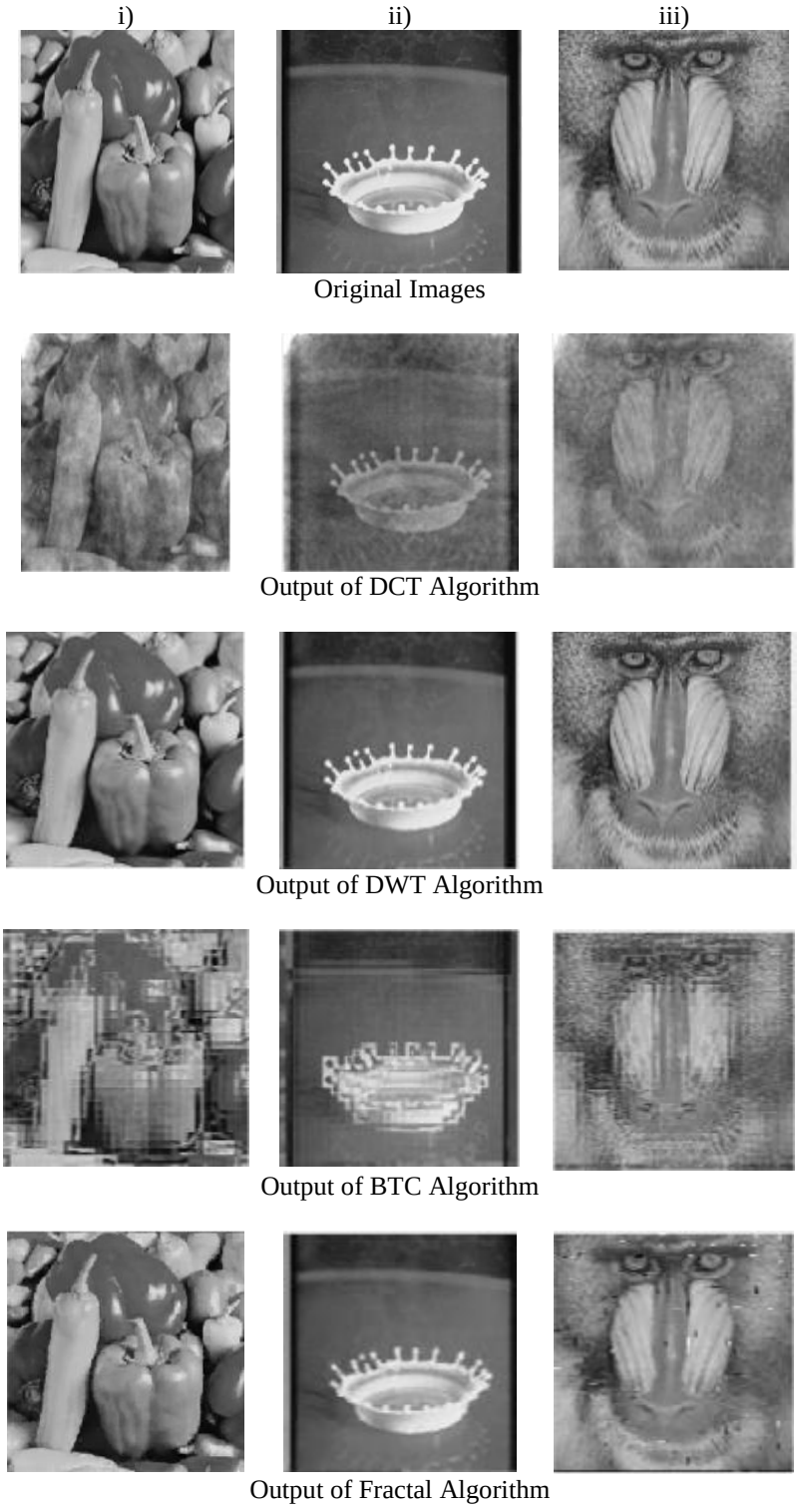

Figure 5. Output of Images using Lossy Compression Algorithms

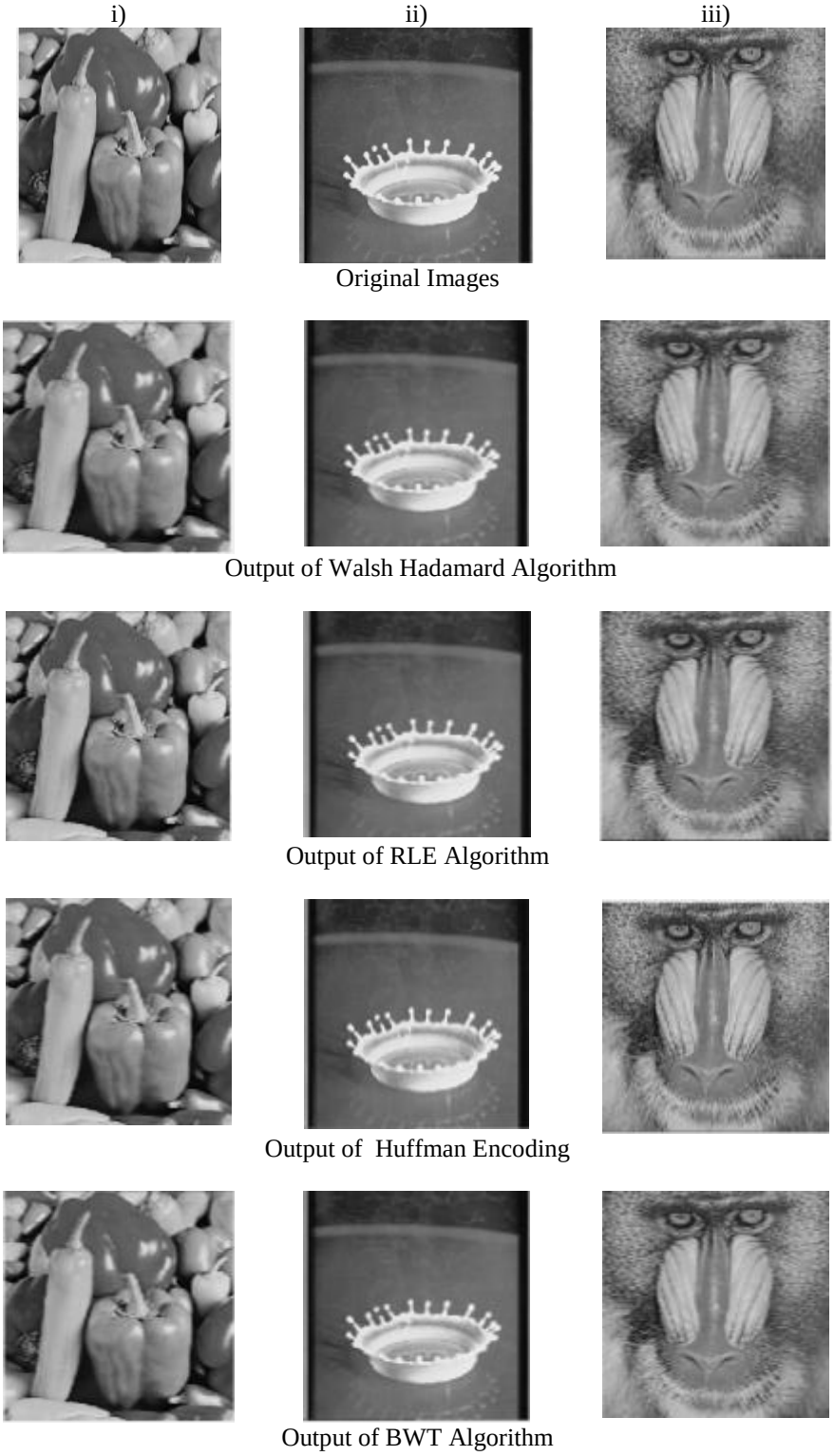

Figure 6. Output of Images using Lossless Compression Algorithms

Table 1. Comparative Analysis of Lossy and Lossless compression algorithms

Compression Algorithm	Average results of 100 images							
	DCT	DWT	BTC	Fractal	Walsh Hadamard	RLE	Huffman	BWT
Compression Techniques	Lossy	Lossy	Lossy	Lossy	Lossless	Lossless	Lossless	Lossless
Original File size	38278	38278	38278	38278	38278	38278	38278	38278
Compressed File Size	5096	777	6395	174	27007	694	215	393
Decompressed file Size	34128	38278	18446	31902	38278	38278	38278	38278
Compression Ratio	0.13	0.02	0.167	0.004	0.705	0.018	0.0101	0.0102
Compression Time	8.01	0.0867	0.04	27.7	0.1	0.83	0.277	0.004
Decompression Time	0.015	0.0021	0.02	0.2	0.11	6.12	0.06	0.0007
Compression Speed	636.205	8961.938	159875	6.282	270070	836.145	776.173	98250
Compression Factor	259.371	49.26	5.98	219.98	1.417	55.15	99.009	97.399
Saving Percentage	99.614	97.97	83.29	99.54	29.44	98.18	99.43	98.97

Figure 11 examines the performance of algorithms based on Compression Factor parameter. In lossy techniques, DCT algorithm acquires the maximum value of 259.371 whereas DWT, BTC and Fractal Encoding achieves 49.26, 5.98, and 219.98 respectively. However, in lossless techniques Huffman Encoding achieves higher compression factor value of 99.009 as compared to Walsh Hadamard, RLE, and BWT that has compression factor values as 29.44, 98.18, and 98.97 respectively.

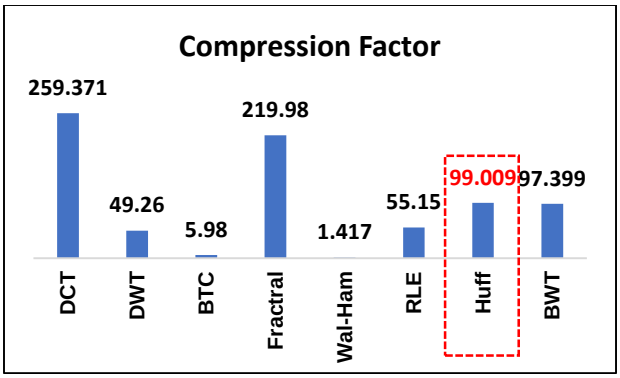

Figure 11. Performance on the basis of compression factor offered by different image compression algorithms

Figure 12 shows the performance analysis of the techniques on the basis of Saving Percentage parameter. Out of lossy algorithms the best performance is shown by DCT with 99.614% whereas the algorithms such as DWT, BTC and Fractal encoding obtains the saving percentage values as 97.97%, 83.29%, and 99.54%. In lossless algorithms the Huffman Encoding achieves the highest saving percentage i.e., 99.43 where the other algorithms like Walsh Hadamard, RLE and BWT acquires the saving percentage values 29.44%, 98.18%, and 98.97% respectively.

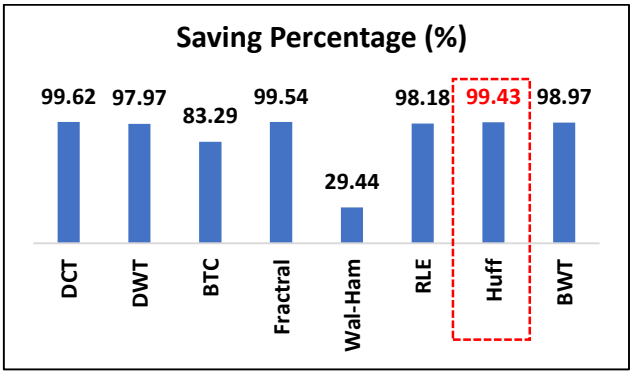

Figure 12. Performance on the basis of saving percentage offered by different image compression algorithms

6. Conclusion

In this paper, distinctive types of image compression techniques are evaluated on the basis of certain parameters such as compressed file size, compression ratio, compression time, decompression time, compression speed, compression factor and saving percentage. The two types of image compression techniques which are used predominantly are discussed and evaluated. Comparing the lossy and lossless techniques, the lossy techniques performs better than lossless based on quantitative analysis but we talk of qualitative analysis lossless performs better. So, it really depends on the requirements of the user to select a particular technique. However lossless techniques are more effective as there isn't any data or image loss during the compression and same image and data is

retrieved at the end of the process. From all the lossless techniques considered above, Huffman Encoding performs better than other algorithms through compressed file size, Compression Ratio, Compression Factor, and Saving Percentage whereas BWT performs better as per the Compression and Decompression time required.

References

[1] Uthayakumar, J., Vengattaraman, T., & Dhavachelvan, P. (2018). A survey on data compression techniques: From the perspective of data quality, coding schemes, data type and applications. *Journal of King Saud University - Computer and Information Sciences*. doi: 10.1016/j.jksuci.2018.05.006

[2] ZainEldin, H., Elhosseini, M., & Ali, H. (2015). Image compression algorithms in wireless multimedia sensor networks: A survey. *Ain Shams Engineering Journal*, *6*(2), 481-490. doi: 10.1016/j.asej.2014.11.001

[3] Thanki, R., & Kothari, A. (2019). Color Medical Image Compression Techniques. pp. 83-92.

[4] Sunil, H., & Hiremath, S. (2018). A combined scheme of pixel and block level splitting for medical image compression and reconstruction. *Alexandria Engineering Journal*, *57*(2), 767-772. doi: 10.1016/j.aej.2017.03.001

[5] Pradhan, A., Pati, N., Rup, S., Panda, A., & Kumar, L. (2016). A comparative analysis of compression techniques – the sparse coding and BWT. In *2nd International Conference on Intelligent Computing, Communication & Convergence (ICCC-2016)* (pp. 106-111). Bhubaneswar, Odisha, India: Elsevier B.V.

[6] Gong, A., Qiu, K., Deng, C., & Zhou, N. (2019). An image compression and encryption algorithm based on chaotic system and compressive sensing. *Optics & Laser Technology*, *115*, 257-267. doi:10.1016/j.optlastec.2019.01.039

[7] Deepthi, S., Rao, E., & Prasad, M. (2017). Image compression techniques in wireless sensor networks. In *International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)* (pp. 286-289). Chennai, India: IEEE.

[8] Sharma, K., & Gupta, K. (2017). Lossless data compression technique with encryption based approach. In *8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. Delhi, India: IEEE.

[9] Madhavee Latha, P., & Annis Fathima, A. (2019). Collective compression of images using averaging and transform coding. *Measurement*, *135*, 795-805. doi: 10.1016/j.measurement.2018.12.035

[10] Rawat, C., & Rao, S. (2014). Evaluation of Burrows Wheeler Transform based image compression algorithm for multimedia applications. In *International Conference on Advances in Communication and Computing Technologies (ICACACT)*. Mumbai, India: IEEE.

[11] Patil, R., & kulat, K. (2017). Audio compression using dynamic Huffman and RLE coding. In *2nd International Conference on Communication and Electronics Systems (ICCES)*. Coimbatore, India: IEEE.

[12] Agarwal, N., Jha, S., & Mishra, A. (2017). An Analysis Of Variation In Lossless Color Image Compression using n-MM Algorithm. *International Journal of Computer & Mathematical Sciences IJCMS*, *6*(8), 173-179.

[13] Khalifa, O. (2019). Wavelet Coding Design for Image Data Compression. *The International Arab Journal of Information Technology*, *6*(2), 118-127.

[14] Vimalachandran, P., Wang, H., Zhang, L, Zhuo, G. (2016). *The Australian PCEHR System: Ensuring Privacy and Security through an Improved Access Control Mechanism.* EAI Endorsed Trans. Scalable Information Systems 3(8) 3. e4. 10.4108/eai.9-8-2016.151633.

[15] Zhang, Y., Xu, B., Zhou, N., (2017). *A novel image compression–encryption hybrid algorithm based on the analysis sparse representation.* Optics Communications, vol. 392, 223–233.

[16] Setyaningsih, E., Wardoyo, R., (2017), *Review of image compression and encryption techniques.* International Journal of Advanced Computer Science and Applications, vol. 8, no. 2.

[17] Sharma, R., Bollavarapu, S., (2015). *Data Security using Compression and Cryptography Techniques*. International Journal of Computer Applications, vol. 117, no. 14, pp. 15–18.

[18] Ogiela, M., Ogiela, L., (2016). *On using cognitive models in cryptography*, International Conference on Advanced Information Networking and Applications, AINA, pp. 1055–1058.

[19] Pizzolante, R. and Carpentieri, B. (2013). *Lossless, low-complexity, compression of three-dimensional volumetric medical images via linear prediction.* Proceedings of the 18th International Conference on Digital Signal Processing.

[20] Pizzolante, R. and Carpentieri, B. (2012). *Visualization, band ordering and compression of hyperspectral images.* Algorithms, vol. 5, no. 1, pp. 76–97.

[21] Sophia, P.E. and Anitha, J. (2017). *Contextual Medical Image Compression using Normalized Wavelet-Transform Coefficients and Prediction*, IETE Journal of Research, 63:5, 671-683.

[22] Vidhya, K. (2016). *Medical Image Compression using Adaptive Subband Threshold.* Journal of Electrical Engineering & Technology, 11(2), pp. 499-507.

[23] Ravichandran, D., Ahamad, M.G., Dhivakar, A. (2016). *Performance Analysis of Three- Dimensional Medical Image Compression Based on Discrete Wavelet Transform.* Proceedings of the International Conference on Virtual Systems & Multimedia (VSMM), pp. 351-358.

[24] Lucas, L. F.R., Rodrigues, N. M. M., Cruz, L.A.D, and de Faria, S.M.M., (2017). *Lossless Compression of Medical Images Using 3-D Predictors.* IEEE Transactions on Medical Imaging, Volume 36, Issue 11, pp. 2250-2260.

[25] Haddas, S., Coatrieus, G., Cozic, M., and Bouslimi, S. (2017). *Joint Watermarking and Lossless JPEG-LS Compression for Medical Image Security*. Innovation and Research in BioMedical engineering, Volume 38, Issue 4, pp. 198-206.

[26] Narmatha, C.; Manimegalai, P.; Manimurugan, S. (2017). *A Lossless Compression Scheme for Grayscale Medical Images Using a P2-Bit Short Technique.* Journal of Medical Imaging and Health Informatics, Volume 7, Issue 6, pp. 1196-1204.

[27] AbuBaker, A., Eshtay, M., AkhoZahia, M., (2016). *Comparison Study of Different Lossy Compression Techniques Applied on Digital Mammogram Images.* Volume 7, No. 12, pp. 149-155.

[28] Hagag, A., Fan, X.P., Abd El-Samie, F.E., (2016). *The Effect of Lossy Compression on Feature Extraction Applied to Satellite Landsat ETM plus Images*. International Conference on Digital Image Processing, Chengdu, China.

[29] Kozhemiakin, R., Abramov, S., Lukin, V., Djurovic, B., Djurovic, I., and Vozel, B., (2016). *Lossy Compression of Landsat Multispectral Images*. Mediterranean Conference on Embedded Computing, Montenegro, pp. 104-107.

[30] Thanki, Rohit M., Kothari, Ashish. (2019). *Data Compression and Its Application in Medical Imaging*. Hybrid and Advanced Compression Techniques for Medical Images, Pages 1-15.

[31] kumar, S., Bhatnagar, G. (2019). *SIE: An Application to Secure Stereo Images Using Encryption*, Handbook of Multimedia Information Security: Techniques and Applications, Pages 37-61.

[32] Singh, S., Devgon, R., (2019). *Analysis of Encryption and Lossless Compression Techniques for Secure Data Transmission* International Conference on Computer and Communication Systems (ICCCS), DOI: 10.1109/CCOMS.2019.8821637, pp.120-124.

[33] Thakur, Nileshsingh & Kakde, Omprakash. (2018). *Compression Mechanism for Multimedia System in consideration of Information Security*.

[34] Shunmugan, S., Arockia P., (2016). *Encryption-then-Compression Techniques: A Survey*. International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 978-1-5090-5240-0/16.

[35] Tsai, C., Shih, W., Lu, Y Huang, J., and Yeh, L. (2019). *Design of a Data Collection System with Data Compression for Small Manufacturers in Industrial IoT Environments.* Asia-Pacific Network Operations and Management Symposium (APNOMS), pp. 1-4.

[36] Dhanawe, S., Doshi, V., (2016). *Hiding file on Android Mobile and Sending APK file through whats app using Steganography and Compression techniques.* International conference on Signal Processing, Communication, Power and Embedded System (SCOPES), pp. 106-110.

[37] Anand, A., Singh, A.K. (2020). *An improved DWT-SVD domain watermarking for medical information security*. Computer Communications Volume 152, Pages 72-80.

[38] Li, P., Tung. (2019). *Joint image encryption and compression schemes based on 16 × 16 DCT.* Journal of Visual Communication and Image Representation Volume 58, Pages 12-24.

[39] Yuan, S. Hu, J., (2019). *Research on image compression technology based on Huffman coding.* Journal of Visual Communication and Image Representation, Volume 59, Pages 33-38.

[40] Setyaningsih, E., Wardoyo, R., KartikaSari, A., (2020). *Securing color image transmission using compression-encryption model with dynamic key generator and efficient symmetric key distribution*. Digital Communications and Networks, pp. 1-21.

[41] Zhang, G., Wang, J., Yan, C., Wang, S., (2019). *Application research of image compression and wireless network traffic video streaming*. Journal of Visual Communication and Image Representation, Volume 59, pp. 168-175.

[42] Kester QA. (2018). *A Hybrid Lossy Compression Using 2-D Discrete Cosine Transform and Visual Cryptographic Technique for Security of Multimedia Image Data Communications in Internet of Things*. In: Bissyande T., Sie O. (eds) e-Infrastructure and e-Services for Developing Countries. AFRICOMM 2016. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 208, pp 292-303.

[43] Chen, T., Chang, T. (2018). *On the security of a BTC-based-compression image authentication scheme*. Multimed Tools and Applications vol 77, pp. 12979–12989.

[44] Lin CC, Huang Y, Tai WL (2017*). A novel hybrid image authentication scheme based on absolute moment block truncation coding*. Multimedia Tools and Applications 76(1), pp. 463–488.

[45] Guo JM, Prasetyo H, Wong K., (2016). *Halftoning-based block truncation coding image restoration*. Journal of Visual Communication and Image Representation, vol 35, pp.193–197.

[46] Song, Y., Zhu, Z., Zhang, W., (2019). *Joint image compression–encryption scheme using entropy coding and compressive sensing*. Nonlinear Dynamics, vol 95, pp. 2235–2261.

[47] Tong, X., Zhang, M., Wang, Z. (2016). *A joint color image encryption and compression scheme based on hyper-chaotic system*. Nonlinear Dynamics, vol 84, pp. 2333–2356.

[48] Raghavendra, C., Sivasubramanian, S., Kumaravel, A. 92019). *Improved image compression using effective lossless compression technique*. Cluster Computing, vol 22, pp. 3911–3916.

[49] Preston, C., Arnavut, Z., and Koc, B., (2015*). Lossless compression of medical images using Burrows-Wheeler Transformation with Inversion Coder*. 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Milan, pp. 2956-2959.

[50] Jeromel, A., Žalik, B. (2020). *An efficient lossy cartoon image compression method.* Multimedia Tools and Application, vol 79, pp. 433–451.

[51] Salomon D, Motta G (2010) Data compression: The complete reference, 5th edn. Springer, New York.

[52] Hussaina, A., Al-Fayadh, A., Radi N., (2018). *Image compression techniques: A survey in lossless and lossy algorithms*. Neurocomputing, Volume 300, Pages 44-69.

[53] Ahmad, N., Iqbal, K., Han, L., Iqbal, N., Abid, M., (2019). Zone Based Lossy Image Compression Using Discrete Wavelet and Discrete Cosine Transformations. Security and Privacy in New Computing Environments. Second EAI International Conference, SPNCE.

[54] Vimalachandran, P., Wang, H., Zhang, H., Zhuo, G., Kuang H.,(2017) Cryptographic access control in electronic health record systems: a security implication. International Conference on Web Information Systems Engineering, 540-549.

[55] Chenthara, S., Ahmed, K., Wang, H., Whittaker, F., (2019) *Security and privacy-preserving challenges of e-Health solutions in cloud computing.* IEEE access 7, 74361-74382.

[56] Vimalachandran, P., Wang, H., Zhang, L, Zhuo, G. (2016). *The Australian PCEHR System: Ensuring Privacy and Security through an Improved Access Control Mechanism.* EAI Endorsed Trans. Scalable Information Systems 3(8) 3. e4. 10.4108/eai.9-8-2016.151633.

[57] Ballé, J., (2018). *Efficient Nonlinear Transforms for Lossy Image Compression*. Picture Coding Symposium (PCS), San Francisco, CA, pp. 248-252.

[58] Qin, C., Zhou, Q., Cao, F., Dong J., and Zhang, X., (2019). *Flexible Lossy Compression for Selective Encrypted Imagewith Image Inpainting*. IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 11, pp. 3341-3355.

[59] Dagher, I., Saliba, M., Farah, R., (2018). *Combined DCT-Haar transforms for image compression*. International Journal of Imaging Systems and TechnologyVolume 28, Issue 4.

[60] Kahu, Y. S., Raut, B. R., Bhurchandi, M. K., (2018). Review and evaluation of color spaces for image/video

compression. Color Research & Application, volume 44, Issue 1.

[61] Zhang, Ji. (2013). Advancements of Outlier Detection: A Survey. ICST Transactions on Scalable Information Systems. 13. e2. 10.4108/trans.sis.2013.01-03.e2.

[62] Bano N., Alam M., Ahmad S. (2018) Performance Evaluation of Wavelet-Based Image Compression Techniques. In: Singh R., Choudhury S., Gehlot A. (eds) Intelligent Communication, Control and Devices. Advances in Intelligent Systems and Computing, vol 624.