A semi-lossless image compression procedure using a lossless mode of JPEG

Md. Atiqur Rahman[1] & Mohamed Hamada[2,*]
[1,2] School of Computer Science and Engineering, The University of Aizu, Aizu-Wakamatsu City, Fukushima, Japan
[1]atick.rasel@gmail.com & [2]mhamada2000@gmail.com

***Abstract*—Day by day, communication through the internet is progressing; particularly video calling for which sending a massive amount of data over the internet and saving on a computer is being a big challenge. For which, there are many compression algorithms; such as block transform, vector quantization and JPEG are used to convert a big dataset in such a format so that it can be sent over the internet at a high speed and stored in a small space in computer. In this paper, a new procedure has been proposed using a lossless mode of JPEG by removing the first and last bits from the exact binary pattern of each pixel which provides a better result than the state-of-the-art techniques. In this technique, the two bits are removed after a little bit preprocessing and then the remaining binary pattern is replaced by a fixed value each. Lastly, average code word, compression ratio and PSNR are used to assess the performance of the proposed procedure with the state-of-the-art techniques. From the experimental results, it looks that the proposed procedure provides better results than the state-of-the-art techniques.**

Keywords: Huffman coding, JPEG, lossless, lossy, average code length, compression ratio and peak signal to noise ratio.

I. Introduction

A huge amount of medical image, including MRI, computed tomography, capsule endoscope, and ultrasound images are produced from medical centers every day. As a result, a considerable memory space is required to store the images which are very challenging. One solution to this problem is to compress an image. When an image is compressed with or without a little bit distortion, it takes very less time for sending over the internet as well as less memory than an original image. Many techniques already have been developed by researchers to compression an image. Such as run-length, Shannon-Fano, LZW, Huffman, and Arithmetic coding are used in case of lossless, where as vector quantization, block transmission and JPEG for lossy. The compression techniques reduce either substantial amount of redundancies found in an image or use coding techniques to compress an image [1].

Borut et al. proposed a chain code compression procedure based on the move-to-front transform (MTFT) for entropy reduction [2]. Finally, an adaptive run-length coding is used to compress the chain code. In that paper, it has been shown that within 4 MTFT repetitions, Shannon-Fano provides the soundest reduction in case of information entropy. In [3], SVD and Huffman coding based lossy image compression technique has been proposed. First of all, an image is disintegrated using singular value decomposition (SVD) and then Huffman coding is used after neglecting some lower values to compress an image. In that technique, the results are compared with Huffman coding and JPEG2000, and shown that the technique is better than Huffman and JPEG2000. Awwal et al. proposed SVD and wavelet difference reduction (WDR) based image compression procedure in [4]. SVD provides a low compression ratio. So, WDR is applied on the SVD based decomposed image to get a higher compression ratio and shows that it works better than WDR and JPEG2000 based image compression.

In these articles [5-8], DWT, SVD and Huffman coding based image compression procedure has been proposed where most and less notable coefficients selection were the main idea and the performance of the technique is compared with the JPEG standard, and demonstrates better than JPEG. Huffman coding and the various embedded wavelet based image compression procedure is proposed in [9]. At low bitrates, it provides quite good PSNR values.

Rahman et al. proposed binary sequence based a lossless image compression procedure in [10] and histogram modification based lossy image compression technique in [11]. Both of the articles provide better results than Huffman, Shannon-Fano and run-length coding. But it takes more time than JPEG.

A JPEG based image compression technique proposed in [12] which permits a little bit loss, but it is so minute that human eyes are not possible to identify because of high-frequency degradation. It shows an improved performance with Huffman coding and JPEG. JPEG uses discrete cosine transformation (DCT) to convert an image into different frequencies first and secondly, quantization technique is applied to the DCT coefficients. Quantization is the way of reduction information of an image. Finally, Huffman coding is applied to encode the coefficients and the same reverse procedure is followed to reconstruct the image [13]. Keng et al. proposed an improved JPEG compression by changing the parameters used based on DCT in [14-16]. They compress 2D barcode images and show higher PSNR than the baseline implementation. Li Zhiqianga et al. proposed Digital Chaos compression in their paper [17]. That paper analyzed and discussed about the JPEG algorithm. It used Matlab simulation and a hardware implementation for image acquisition and compression. Finally, It has been found that the JPEG is perfect for engineering applications and secure communication. B.E. Usevitch

*Corresponding Author

978-1-7281-4882-3/19/$31.00 ©2019 IEEE
DOI 10.1109/MCSoC.2019.00028

discussed about the differences between early wavelet coding with modern wavelet coding to understand JPEG 2000 well in [18].

Kodras et al. in [19] presents JPEG 2000 and compares it with the other discovered standard methods based on scalability, region-of-interest coding, error resilience, file format and visual weighting. Here, it has been concluded that JPEG2000 works better than the existing algorithm for the still images. Awwal et al. proposed singular value decomposition (SVD) and Huffman coding based lossy image compression procedure in [20]. Here, SVD is applied to an image for decomposition and then apply Huffman coding after dropping some smaller singular values. Several medical images were taken as test images and shown that it works better than conventional Huffman coding and JPEG2000. In [21], a framework for an image compression has been proposed using Burrow Wheeler Transform (BWT). Entropy coding is applied after transformation in that technique. It shows that it works better than conventional JPEG. Bin et al. proposed an integer Discrete Tchebichef Transform (IDTT) based image compression scheme in article [22] and showed that IDTT based compression provides a higher compression ratio than the IDCT method.

In this paper, we proposed a semi-lossless image compression procedure using a lossless mode of JPEG. In this procedure, some pixels of a main image is modified first and then destroyed the first and last bits for the binary value of each pixel of the image. After that each remaining bit pattern is searched in an array that contains a list of binary patterns. Finally, the remaining pattern is replaced by the array value. Finally, a lossless mode of JPEG is applied to encode the processed data of an image. To assess the performance of the proposed technique average code length (ACL), compression ratio(CR) and peak signal to noise ratio (PSNR) have been considered. Some existing methods and the proposed technique have been discussed in section II and III respectively. Finally, the result summary and conclusion have been shown in section IV and V sequentially.

II. Background Study

A. Run-length Coding

Run-length is a lossless coding procedure that stores only a mark and a count when a series of identical values appears with consecutive times. This strategy is more cost-effective instead of encoding each pixel [23-25]. In that strategy, it chooses the first pixel from an image and then connects it to the output string and counts the number of succeeding occurrence of the selected pixels. Lastly, it appends the count to the destination string. This process continues until reading the whole pixels of an image is finished. The working procedure for run-length coding for compression and decompression are shown here:

Run-length Encoding

1) An image is read first and perform the differentiation between neighboring pixels. Secondly, each non-zero value is replaced by 1 and discover the positions of the non-zeros and find the corresponding values from the original list.
2) The list of discovered positions and values are sent to decoder for decompression.

Run-length Decoding

1) For decompression, the list of positions and values are received.
2) Read each value from the list and its corresponding position and then repeat the value till the position and continue the same process for all the positions and values until finished.

B. Huffman Coding

Huffman is a lossless data compression algorithm which uses fewer bits to encode for those pixels that happen more regularly. Huffman normally builds a tree based on probabilities of an image and then encoding is done. The compression and decompression of Huffman coding are given below [26-28].

Huffman Encoding

1) An image is read first and find all probabilities of the image, and sort the probabilities in descending order.
2) Connect two lowest probabilities and form a Huffman tree, the process is continued until the root of the tree is left with probability 1.
3) 1 and 0 is assigned to the left and right branches of the tree. Later encoding is done based on the tree.
4) the encoded matrix, code-words and the corresponding symbols are sent to decoder for decompression.

Huffman Decoding

1) The encoded matrix, code-words and the list of symbols are taken for decompression.
2) Construct the same Huffman tree as it was built in encoding procedure. Finally decompression is done based on the tree and the symbols list.

C. JPEG Procedure

JPEG is called the First international standard image compression method which can be used in case of lossless or lossy. The algorithm provides a better result for photo and painting images, and it is not adequately suited for textual graphics, line drawings and iconic graphics, where a noticeable artifact can be created in case of the sharp contrasts between adjacent pixels [29-31]. The encoding procedure of JPEG is given below.

1) An image is converted into Y'CbCr and Y' channel is broken into a block of 8x8.
2) Each block is mapped from -128 to 127 and then DCT is applied to get a list of quantized coefficient.
3) Finally, encoding is done in zigzag manner until all zero is found.

III. PROPOSED PROCEDURE

In this proposed procedure, a semi lossy image compression procedure has been recommended by dropping the first and last bit from a binary pattern of each pixel using a predefined pattern list and JPEG. After dropping the two bits, each remaining binary pattern is searched in the list and replaced by its corresponding index to the pattern list. After preprocessing, all pixel's value is in between 0-130 and then apply JPEG encoding procedure to compress the data which outperforms than the Shannon-Fano, Huffman and JPEG coding. The proposed coding procedure is demonstrated in Fig. 1.

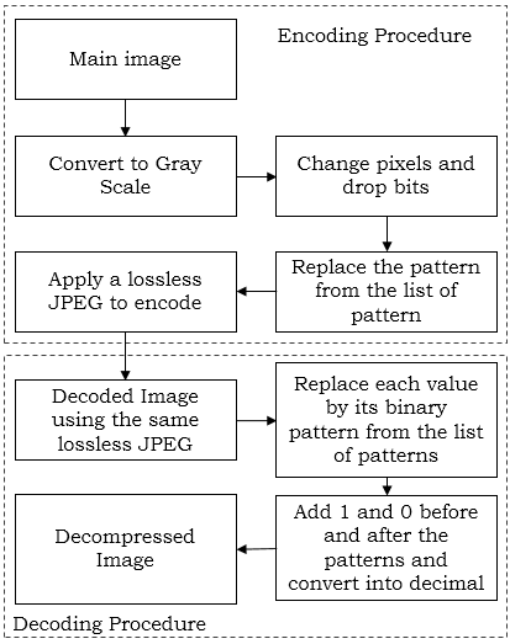

Fig. 1. Diagram for the proposed compression and decompression procedure

The compression and decompression algorithms for the proposed procedure are shown here:

Encoding procedure of the proposed coding

1) A color image is read and then convert it into a gray scale image and assign 130 number of binary pattern in a list called PATTERN_LIST where all pattern come from the 1-6 bit's binary combination. As an example, 2,4,8, 16 number of combinations are generated for 1, 2, 3 and 4 bits binary number respectively.
2) Set each odd pixel(PIX) of the image to its nearest value (EVENPIX) so that it obeys the following condition.

if $PIX \geq 4$ **then**
| *EVENPIX = PIX-1;*
else
| *EVENPIX=PIX;*
end

3) Convert each pixel's value to its corresponding binary and remove the first and last bits from the binary pattern if the decimal representation of the value is greater than 3.
4) Replace each remaining bits pattern by its index number from the array PATTERN_LIST.
5) Finally, apply a lossless mode of JPEG coding to encode.

Decoding procedure of the proposed coding

1) Read the encoded matrix and apply the lossless mode of JPEG to get pre-decoded matrix (PDM).
2) Read each data (PDD) from the PDM and search in the same array PATTERN_LIST and get the corresponding pattern.
3) if PDD is greater than 3 then add 1 and 0 before and after the binary sequence.
4) Again convert the binary sequence to its decimal value which will be the same value of the original gray scale image.
5) Repeat Step 2 and 4 until scan of all the pixels is finished.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, experimental results and an analysis has been demonstrated, and average code length (ACL), compression ratio(CR) and peak signal to noise ratio (PSNR) are used to assess the performance of the proposed scheme. The proposed scheme and the state-of-the-art techniques applied to some original images of different size shown in Fig. 2, and their corresponding compressed and decompressed images are demonstrated in Fig. 3 and Fig. 4 respectively.

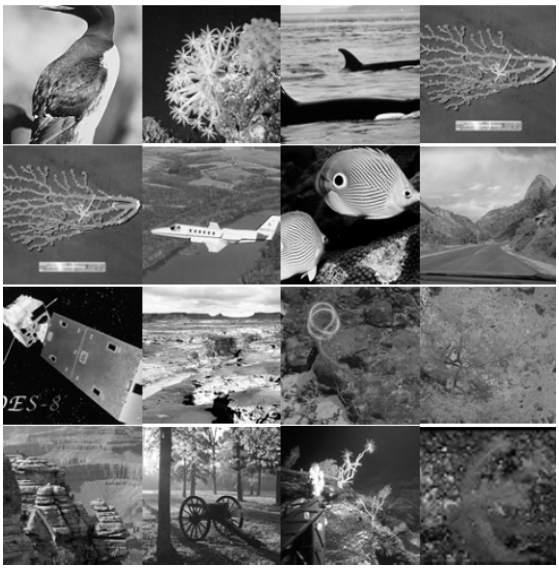

Fig. 2. Main image list

If the size of an image is MxN and the number of probabilities (Pro) of the image is NP. Then, the following equation 1, 2, 3 and 4 are applied to estimate the ACL, CR, MSE and

PSNR where MAX, CODE_LENGTH, MI and RI represent the greatest variation of the input image, length of code-word of i^{th} probability, main image and reconstructed image respectively.

$$ACL = \sum_{i=1}^{NP} Pro(i) * CODE\_LENGTH(i) \quad (1)$$

$$CR = \frac{uncompressed\,size}{compressed\,size} \quad (2)$$

$$MSE = \frac{\sum_{M,N} [MI(M,N) - RI(M,N)]^2}{MxN} \quad (3)$$

$$PSNR = 10 * log_{10} \left(\frac{MAX^2}{MSE} \right) \quad (4)$$

We have applied Huffman coding, JPEG and the proposed

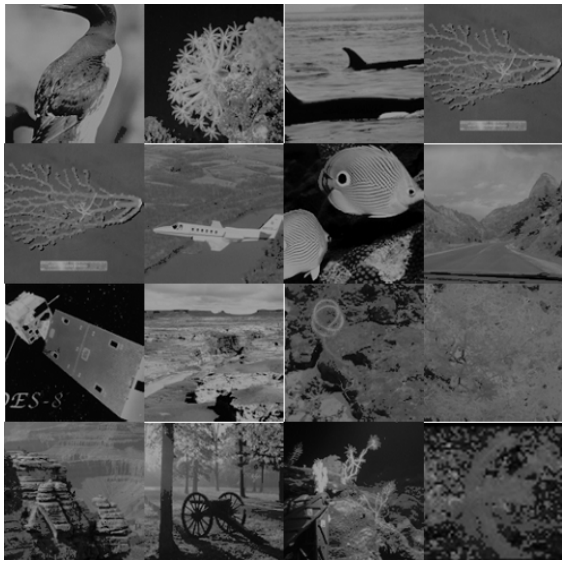

Fig. 3. Compressed image list

procedure on sixteen natural images of various size and the average code length and PSNR are shown in TABLE I. Also compression ratio is shown in TABLE II. Finally, the graphical representation of ACL and CR are shown in Fig. 5 and 6 respectively for comparison.

In this article, TABLE I represents average code length of the state-of-the-art and the proposed procedures as well as the PSNR of the proposed procedure. The graphical representation of the average code lengths IS demonstrated in Fig. 5 for comparison. TABLE I shows that the proposed algorithm reduces averagely 90.2342% more space than the Huffman coding where as JPEG use averagely 13.4042% more memory than the recommended procedure. A reconstructed image quality is good when PSNR between reconstructed and original images is greater than or equal to 30. Since, PSNR of each image for

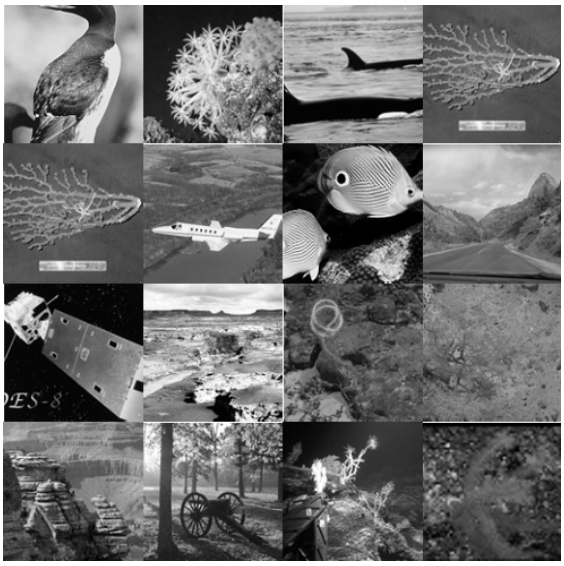

Fig. 4. Decompressed image list

TABLE I
AVERAGE CODE LENGTH AND PSNR

Images	Huffman	JPEG	Proposed	PSNR
1	7.6679	0.713	0.6123	48.2166
2	7.4768	0.9202	0.7908	46.6683
3	6.9131	0.6844	0.5721	52.5576
4	6.421	0.7642	0.64	47.9176
5	7.4949	0.6328	0.5208	52.5683
6	6.706	0.745	0.6128	47.2379
7	7.5131	0.9038	0.7924	53.0757
8	7.308	0.6555	0.5604	47.2205
9	6.2664	0.7617	0.646	50.9356
10	7.6707	0.9298	0.7641	45.72
11	6.9579	0.8706	0.8625	42.7699
12	6.3837	0.8415	0.6828	49.5757
13	7.4682	0.8949	0.8156	45.7455
14	7.6492	0.9756	0.886	44.618
15	7.3709	0.8125	0.7577	44.6281
16	6.9125	0.7213	0.6221	43.4033

proposed algorithm is greater than 40, so it can be said that the decompressed images are good. Fig. 5 shows that the line graph for the average code length of the proposed scheme is under than that of Huffman and JPEG. So, the proposed procedure uses tiny average code length than the state-of-the-art techniques.

We have shown the compression ratio of the experimental images in TABLE II which shows that Huffman coding provides 1.0433 compression ratio where JPEG and the proposed coding produce 11.2202 and 13.0655 compression ratio respectively for the first image. From the result, it has been calculated that the proposed procedure provides 92.0148%

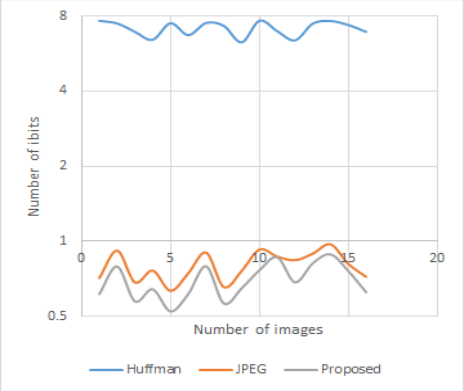

Fig. 5. Comparison of average code length

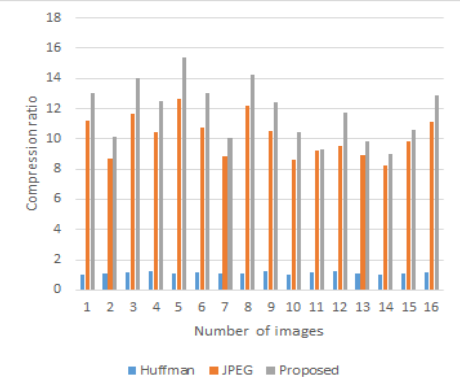

Fig. 6. Comparison of compression ratio

and 14.1235% more compression than Huffman and JPEG respectively. It is seeing for all the images in TABLE II that the proposed technique compresses an image very much than Huffman coding and a little more than JPEG. The graphical representation of compression ratio of the images are demonstrated in Fig. 6 and the highest bar of compression ratio represents the proposed coding and the lowest bar expresses Huffman coding for the same case. The second highest bar indicates JPEG. The compression bar for the proposed coding is higher than Huffman and JPEG which tells that the recommended scheme provides more compression than the state-of-the-art techniques. So, we can state that the proposed scheme provides a better result that the state-of-the-art techniques from the above-mentioned point of parameters.

TABLE II
COMPARISON OF COMPRESSION RATIO

Images	Huffman	JPEG	Proposed
1	1.0433	11.2202	13.0655
2	1.07	8.6938	10.1163
3	1.1572	11.6891	13.9836
4	1.2459	10.4685	12.5
5	1.0674	12.6422	15.361
6	1.193	10.7383	13.0548
7	1.0648	8.8515	10.0959
8	1.0947	12.2044	14.2755
9	1.2767	10.5028	12.3839
10	1.0429	8.604	10.4698
11	1.1498	9.1891	9.2754
12	1.2532	9.5068	11.7165
13	1.0712	8.9395	9.8087
14	1.0459	8.2001	9.0293
15	1.0853	9.8462	10.5583
16	1.1573	11.0911	12.8597

V. CONCLUSION

In this article, a semi lossy image compression technique has been recommended using a lossless mode of JPEG. We applied the mentioned state-of-the-art techniques and the proposed technique on sixteen various bench-marked images and estimates the performance of the proposed algorithm comparing with Huffman and JPEG based on average code length (ACL), compression ratio (CR) and peak signal to noise ratio (PSNR). It is seen that it provides a better result in case of average code length and compression ratio than Huffman coding and JPEG keeping the same quality. It has been calculated that the proposed procedure use 90.2342% and 13.4042% less memory averagely than Huffman coding and JPEG respectively. It is also very fast and more efficient than the state-of-the-art techniques and can be applied in any case of data compression.

REFERENCES

[1] Rabbani, M., Jones, P. W. (1991). *Digital image compression techniques (Vol. 7). SPIE Press.*

[2] alik, B. and Luka, N., 2014. Chain code lossless compression using move-to-front transform and adaptive run-length encoding. *Signal Processing: Image Communication, 29(1), pp.96-106.*

[3] Rufai, A.M., Anbarjafari, G. and Demirel, H., 2013, April. Lossy medical image compression using Huffman coding and singular value decomposition. *In Signal Processing and Communications Applications Conference (SIU), 2013 21st (pp. 1-4). IEEE.*

[4] Rufai, A.M., Anbarjafari, G. and Demirel, H., 2014. Lossy image compression using singular value decomposition and wavelet difference reduction. *Digital signal processing, 24, pp.117-123.*

[5] Kumar, M. and Vaish, A., 2017. An efficient encryption-then-compression technique for encrypted images using SVD. *Digital Signal Processing, 60, pp.81-89.*

[6] Rahman, M.A. and Rabbi, M.F., 2015. Non-Blind DWT-SVD based Watermarking Technique for RGB Image. *Global Journal of Research In Engineering.*

[7] Rahman, M.A. and Rabbi, M.F., 2015. DWT-SVD based New Watermarking Idea in RGB Color Space. *International Journal of Signal Processing, Image Processing and Pattern Recognition, 8(6), pp.193-198.*

[8] Gupta, D. and Choubey, S., 2015. Discrete wavelet transform for image processing. *International Journal of Emerging Technology and Advanced Engineering, 4(3), pp.598-602.*

[9] Srikanth, S. and Meher, S., 2013, April. Compression efficiency for combining different embedded image compression techniques with Huffman encoding. *In Communications and Signal Processing (ICCSP), 2013 International Conference on (pp. 816-820). IEEE.*

[10] M. A. Rahman, J. Shin, A. K. Saha and M. Rashedul Islam, "A Novel Lossless Coding Technique for Image Compression," *2018 Joint 7th International Conference on Informatics, Electronics & Vision*

(ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR), Kitakyushu, Japan, 2018, pp. 82-86. doi: 10.1109/ICIEV.2018.8641065

[11] M. A. Rahman, S. M. S. Islam, J. Shin and M. R. Islam, "Histogram Alternation Based Digital Image Compression using Base-2 Coding," *2018 Digital Image Computing: Techniques and Applications (DICTA), Canberra, Australia, 2018, pp. 1-8. doi: 10.1109/DICTA.2018.8615830*

[12] Lin, C.Y. and Chang, S.F., 2001. A robust image authentication method distinguishing JPEG compression from malicious manipulation. *IEEE Transactions on Circuits and Systems for Video Technology, 11(2), pp.153-168.*

[13] Shneier, M. and Abdel-Mottaleb, M., 1996. Exploiting the JPEG compression scheme for image retrieval. *IEEE Transactions on pattern Analysis and machine Intelligence, 18(8), pp.849-853.*

[14] Alfred Keng T. Tan, Douglas Chai, "Improving mobile color 2D-barcode JPEG image readability using DCT coefficient distributions", *Audio Language and Image Processing (ICALIP) 2012 International Conference on, pp. 123-128, 2012.*

[15] Raid, A.M., Khedr, W.M., El-Dosuky, M.A. and Ahmed, W., 2014. Jpeg image compression using discrete cosine transform-A survey. *arXiv preprint arXiv:1405.6147.*

[16] Pasquini, C., Prez-Gonzlez, F. and Boato, G., 2014, October. A Benford-Fourier JPEG compression detector. *In Image Processing (ICIP), 2014 IEEE International Conference on (pp. 5322-5326). IEEE.*

[17] Zhiqianga, L., Xiaoxin, S., Changbin, D. and Qun, D., 2013, September. JPEG algorithm analysis and application in image compression encryption of digital chaos. *In Instrumentation, Measurement, Computer, Communication and Control (IMCCC), 2013 Third International Conference on (pp. 185-189). IEEE.*

[18] Usevitch, B.E., 2001. A tutorial on modern lossy wavelet image compression: foundations of JPEG 2000. *IEEE signal processing magazine, 18(5), pp.22-35.*

[19] Skodras, A., Christopoulos, C. and Ebrahimi, T., 2001. The JPEG 2000 still image compression standard. *IEEE Signal processing magazine, 18(5), pp.36-58.*

[20] Rufai, A.M., Anbarjafari, G. and Demirel, H., 2013, April. Lossy medical image compression using Huffman coding and singular value decomposition. *In Signal Processing and Communications Applications Conference (SIU), 2013 21st (pp. 1-4). IEEE.*

[21] Pradhan, A., Pati, N., Rup, S. and Panda, A.S., 2016, January. A modified framework for Image compression using Burrows-Wheeler Transform. *In 2016 2nd International Conference on Computational Intelligence and Networks (CINE) (pp. 150-153). IEEE.*

[22] Xiao, B., Lu, G., Zhang, Y., Li, W. and Wang, G., 2016. Lossless image compression based on integer Discrete Tchebichef Transform. *Neurocomputing, 214, pp.587-593.*

[23] Golomb, S., 1966. Run-length encodings (Corresp.). *IEEE transactions on information theory, 12(3), pp.399-401.*

[24] Capon, J., 1959. A probabilistic model for run-length coding of pictures. *IRE Transactions on Information Theory, 5(4), pp.157-163.*

[25] Brown, E.F., Nokia Bell Labs, 1961. Optimal run length coding of image signals. *U.S. Patent 2,978,535.*

[26] Van Leeuwen, J., 1976, July. On the Construction of Huffman Trees. *In ICALP (pp. 382-410).*

[27] Knuth, D.E., 1985. Dynamic huffman coding. *Journal of algorithms, 6(2), pp.163-180.*

[28] M. A. Rahman, M. M. Fazle Rabbi, M. M. Rahman, M. M. Islam and M. R. Islam, "Histogram modification based lossy image compression scheme using Huffman coding," *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT), Dhaka, Bangladesh, 2018, pp. 279-284. doi: 10.1109/CEEICT.2018.8628092*

[29] Taubman, D. and Marcellin, M., 2012. JPEG2000 image compression fundamentals, standards and practice: image compression fundamentals, *standards and practice (Vol. 642). Springer Science Business Media.*

[30] Nelson, M. and Gailly, J.L., 1996. The data compression book (Vol. 199, No. 5). *New York: M t Books.*

[31] Pennebaker, W.B. and Mitchell, J.L., 1992. JPEG: Still image data compression standard. *Springer Science Business Media.*