Advances and Applications in Mathematical Sciences
Volume 20, Issue 9, July 2021, Pages 2031-2044

AN OPTIMIZED BTC IMAGE COMPRESSION TECHNIQUE BASED ON SINGULAR VALUE THRESHOLDING IN THE WAVELET DOMAIN

U. NAVEENAKUMARA and A. PADMANABHA REDDY

Department of Studies in Mathematics
V. S. K. University, Ballari-583105, India
E-mail: navee98765@gmail.com
apreddy@vskub.ac.in

Abstract

In this paper, an optimal discrete wavelet transform - singular value thresholding based on the image compression scheme with block truncation coding (BTC) algorithm is presented. A discrete wavelet transform is applied to the cover image is decomposed into sub-bands of different frequencies at one level. Low-frequency sub-band (cA) decomposed using singular value decomposition. Adaptively adjust the low -rank approximation precision threshold as singular value thresholding progresses, which is approximated in compressed form ($\widetilde{c}A$). After that high-frequency component below a certain threshold is removed and reconstruct the approximation matrix via inverse discrete wavelet transform.

The role of precision threshold (k) in the frequency domain to achieve the best performance in terms of compromising the quality of the image and energy retained. Using a block truncation coding method to reduced bit rate of compressed image and compared to the MPQ-BTC method.

1. Introduction

With the exponential popularization of multimedia applications like digital images, audio, and videos, a huge amount of data is transmitted over communication networks and the internet which increases the demand for transmission bandwidth, storage capabilities, and high computational time [1, 2]. Nowadays, digital images have made high-resolution with the high visual quality of getting trendier. These digital images consume a large

2010 Mathematics Subject Classification: 15A18, 65F35.
Keywords: Image compression, Discrete wavelet transform, Singular value thresholding, Block truncation coding, Energy retained.
Received March 19, 2020; Accepted February 14, 2021

amount of storage space due to their big data size, which creates an obstacle in data storage, and transmission. Therefore, an efficient representation of images is strongly needed that consumes a fewer amount of storage space, which can be achieved by data compression process that represents a big data using suitable compression scheme whereas subjectively human perceived quality and energy retention (Er) of the input image is main tained even after compressed [3]. The most successful digital image compression is based on transformed-based technique(s) like Karhumen Loeve Transform [4], JPEG2000 standard based on the discrete wavelet transform (DWT) increased compression ratio with high quality than JPEG [5, 7].

The quantum image compression technique performs a search to determine the most significant discrete cosine transform coefficients [8]. These methods provide a great compress of images with an acceptable loss of data. Singular value decomposition (SVD) is an effective numerical technique used in diagonalizing and splitting the system of matrix into a set of linearly independent sub-matrices, where every sub-matrices has its own energy contribution [9, 10]. It is one of the most commonly effective methods used for minimizing data storage and transfer [11]. Later, BTC method used a two-level (one-bit) non-parametric quantizer that adopts to local properties of the image: low computational complexity and no large data storage is required [12, 13]. Moreover, SVD was combined with other compression techniques in order to improve the compression ratio and PSNR between the compressed and the original images [14, 15]. An adaptive lossy color and grayscale image compression method using SVD and wavelet difference reduction was proposed [2]. Hybrid approaches BTC method combined VQ [16], and singular value decomposition [17] then DWT [18].

The main purpose of this paper is to introduce another scope of image compression by employing "k" in the combined domain of DWT and singular value thresholding (SVT) and to get low-rank approximation matrix. Furthermore, to enhance image quality and acceptable loss of digital image information with prominent compression ratio (Cr), is done using BTC image compression. The rest of this paper is organized as follows. In Section 2 we review of mathematical formulation work. Proposed algorithm presented in Section 3. Simulation results and discussions are given in Section 4, followed by conclusion in Section 5.

2. Preliminaries of Mathematical Work

This section provides a brief introduction to the techniques used in designing the proposed algorithm

2.1. Discrete wavelet transform

Lebesgue spaces are given by:

$$\mathbb{L}^p(\mathbb{R}) = \left\{ x : \| x \|_p = \left(\int_{\mathbb{R}} | x(t) |^p dt \right)^{\frac{1}{p}} < \infty \right\}, \ p \geq 1.$$

A Hilbert space $\mathbb{L}^2(\mathbb{R})$ is formed by the functions which have a finite norm. In these functions the energy $E(f)$ is defined as the square of the norm.

A sequence $\{V_j\}$ where $j \in \mathbb{Z}$ of the closed subspace of $\mathbb{L}^2(\mathbb{R})$ is a multiresolution approximation (MRA) [7, 19]. In two-dimensional DWT, the image is decomposed into four sub-images and each of them having $\frac{M}{2} \times \frac{N}{2}$ 2 size of the original image [20] in first level. One of the sub-images is a low-frequency sub-band (cA) which can be decomposed continually and the others are high-frequency sub-band in the horizontal (cH), vertical (cV) and diagonal (cD) directions.

The DWT of an image $g(n1, n2)$ of size $M \times N$ is defined as

$$W_\varphi(j_0, l1, l2) = \frac{1}{\sqrt{MN}} \sum_{n1=0}^{M-1} \sum_{n2=0}^{N-1} g(n1, n2) \varphi_{j0, l1, l2}(n1, n2), \tag{1}$$

$$W_\psi^i(j_0, l1, l2) = \frac{1}{\sqrt{MN}} \sum_{n1=0}^{M-1} \sum_{n2=0}^{N-1} g(n1, n2) \psi^i_{j0, l1, l2}(n1, n2). \tag{2}$$

The original image can be obtained using inverse discrete wavelet transform (IDWT) expressed as below. For $j \leq j_0$,

$$g(n1, n2) = \frac{1}{\sqrt{MN}} \sum_{l1} \sum_{l2} W_{\varphi}(j0, l1, l2) \varphi_{j0, l1, l2}(n1, n2) +$$

$$\frac{1}{MN} \sum_{i=cH, cV, cD} \sum_{j=j_0} \sum_{l1} \sum_{l2} W_{\psi}^{1}(j, l1, l2) \psi_{j, l1, l2}^{i}(x, y) \tag{3}$$

where $i = cH, cV, cD$ indicate the directional index of the wavelet function and j_0 represents arbitrary scale.

$$\varphi_{j, l1, l2}(n1, n2) = 2^{\frac{j}{2}} \varphi(2^j x - n1, 2^j y - n2), \tag{4}$$

and

$$\psi_{j, l1, l2}^{i}(n1, n2) = 2^{\frac{j}{2}} \psi(2^j x - n1, 2^j y - n2), \; i = cH, cV, cD, \tag{5}$$

here index i identifies the directional wavelets that assumes the values cH, cV and cD. Equations (4) and (5) define the scaled and translated basis functions respectively. $g(n1, n2)$, $\varphi_{j, l1, l2}(n1, n2)$, and $\psi_{j, l1, l2}^{i}(n1, n2)$ are functions of the discrete variable $n1 = 0, 1, 2, \ldots, M - 1$ and $n2 = 0, 1, 2, \ldots, N - 1$. The coefficients defined in Equations (1) and (2) are usually called approximation and detail coefficients, receptively. $W_{\varphi}(j_0, l1, l2)$ coefficients define an approximation of $g(n1, n2)$ at scale j_0 $W_{\psi}^{i}(j, l1, l2)$ coefficients add horizontal, vertical and diagonal details for scale $j \geq j_0$. We normally let $j_0 = 0$ and select $M = N = 2^j$ so that $j = 0, 1, 2, \ldots, J - 1$ and $l1 = l2 = 0, 1, 2, 3, \ldots, 2^j - 1$. Equation (3) shows that $g(n1, n2)$ is obtained via the inverse DWT for given W_{φ} and W_{ψ}^{i} of Equations (1) and (2). In the proposed algorithm we have decomposed the image using "Haar" and "Daubechies" wavelet function [21, 22].

2.2. Singular Value Decomposition

Let $I \in \mathbb{R}^{M \times N}$. The null space and column space of I and I^t provide sufficient details to understand I. Basis and dimension of each subspace can

be obtained by knowing its SVD. SVD of I is given in the following [23]

$$I = \begin{cases} P\begin{pmatrix}\Sigma\\0\end{pmatrix}Q^T & \text{where}\,\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_N), \\ & \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_N \geq 0, \text{ if } M \geq N \\ P(\Sigma\ 0)Q^T & \text{where}\,\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_N) \\ & \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_M \geq 0, \text{ if } M \geq N. \end{cases} \tag{6}$$

Here $P = [p_1, p_2, \ldots, p_M] \in \mathbb{R}^{M\times M}$ and $Q = [q_1, q_2, \ldots, q_n] \in \mathbb{R}^{N\times N}$ are orthogonal. The columns of P and Q are called left and right singular vectors respectively. The non-negative real numbers $\{\sigma_i\}_{i=1}^{\min(M, N)}$ are called singular values. Singular values are unique, but the singular vectors are not. The number of non-zero singular values is called the rank of matrix A and denoted by rank(I) [24].

$$I = \sum_{i=1}^{rank(I)} \sigma_i p_i q_i^t.$$

Truncated singular value decomposition or approximation of a matrix by lower ranks (decaying of smaller singular values to zero) has prominent role in lossy image compression i.e., optimality of matrix I.

2.2.1. Singular value thresholding

SVT computes the singular values exceeding user defined threshold i.e., pre-determined to catch the top k singular values of $I \in \mathbb{R}^{M\times N}$ and associated singular vectors. It can also be used for top singular value decomposition [25].

For any k with $1 \leq k < \min(M, N) = rank(I)$ then define

$$I'_k = \sum_{i=1}^{k} \sigma_i p_i q_i^T. \tag{7}$$

A rank-revealing randomized SVD algorithm is used to adaptively carry out partial SVD to fast approximate the SVT operator given a desired, fixed precision, where the computation cost for partial SVD at each SVT iteration is significantly reduced. A simulated annealing style cooling mechanism is employed to adaptively adjust the low-rank approximation precision

threshold (k) as SVT progresses [26]. The advantage of SVT is encompassing both top singular value decomposition and thresholding, handles both large sparse matrices and structured matrices, and reduces the computation cost in matrix learning algorithms [25]. The approximation of I, i.e., I_k' which can be stored only $k(M + N + 1)$ [3, 27] coefficients to store in computer memory while in frequency domain $k(\frac{M}{N} + \frac{N}{2} + 1)$.

2.3. Block truncation coding

BTC is a simple lossy image compression technique to compress monochrome image data, originally introduced by Delp and Mitchell [12]. It achieves 2 bits per pixel (bpp) with low computational complexity.

Suppose that the size of the original image size $M \times N$. Before BTC compression, the original image is partition into N non-overlapping blocks, $\{O_i\}_{i=1}^{N}$ of size $l \times l$. The number of blocks N where $N = \frac{M \times N}{l \times l}$.

Let $n = l \times l$, and let $z_1, z_2, z_3, \ldots z_n$, be the intensity values in a original image $I \in \mathbb{R}^{M \times N}$. The BTC preserves the two moments average of block $(\bar{z})$ and standard deviation (σ) of each digital block. The description is as follows:

$$\bar{z} = \frac{1}{n} \sum_{i=1}^{n} z_i \tag{8}$$

Where z_i indicates the pixel value in the i^{th} position of the block.

$$\sigma = \sqrt{\frac{1}{n} \sum_{1}^{n} (z_i - \bar{z})^2}. \tag{9}$$

The output data of BTC for an image block contains one bitmap $B(m \times n)$ bit plane and two 8-bit quantization levels (Y_L and Y_H): The two quantization levels Y_L and Y_H of an image block are defined as follows:

$$Y_L = \bar{z} - \sigma \sqrt{\frac{p_s}{n - p_s}}. \tag{10}$$

$$Y_H = \bar{z} + \sigma\sqrt{\frac{n - p_s}{p_s}}. \tag{11}$$

Here p_s denotes the number of pixels with values greater than or equal to the mean value $\bar{z}$. As for the bitmap $B = \{b_1, b_2, \ldots, b_n \mid b_i \in \{0, 1\}, i = 1, 2, \ldots, n\}$ of an encoded image block, if a pixel value z_i is greater than the mean value $\bar{z}$, then b_i is set to be 1; otherwise, it is set to be 0. Each image block in turn is encoded using the BTC uses a triple (Y_L, Y_H, B).

The decoding procedures BTC uses a triple data (Y_L, Y_H, B) to decode an image block. If a value bi of B is 0, then the reconstructed value $\hat{z}_i$ is Y_L; otherwise, the reconstructed value is Y_H [28, 29]. The decoding rules are listed as follows:

$$\hat{o}_i = \begin{cases} Y_L & b_i = 0, \\ Y_H & b_i = 1. \end{cases} \tag{12}$$

An example of BTC compression is shown below. Assume that the block size used in BTC is set to be 4×4 pixels.

Let 4×4 block be

$$O = \begin{bmatrix} 114 & 117 & 121 & 102 \\ 118 & 115 & 121 & 141 \\ 120 & 120 & 126 & 79 \\ 117 & 139 & 75 & 65 \end{bmatrix}.$$

The mean value of this block $\bar{z} = 111.875$, and standard deviation $\sigma = 20.7993$ and two reconstructed values $Y_i = 75.8493$, $Y_H = 123.8835$, receptively. Otherwise, it is 0. The binary bitmap and reconstructed block are listed as follows:

$$B = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}.$$

$$\hat{O} = \begin{bmatrix} 123.8835 & 123.8835 & 123.8835 & 75.8496 \\ 123.8835 & 123.8835 & 123.8835 & 123.8835 \\ 123.8835 & 123.8835 & 123.8835 & 75.8496 \\ 123.8835 & 123.8835 & 75.8496 & 75.8496 \end{bmatrix}.$$

The reconstructed image obtained from applying this technique have a bit rate of 2 bits/pixel. The PSNR is used as a measure of the reconstructed image quality by block wise compared to original image. Here, a bit map of 16-pixel values is derived from a block of 4×4 pixels, and reconstruction levels and respectively may hold an eight-bit storage space. As a result, every block keeps a feature of 16×2 bits. In total, the size of all features may accommodate $l \times l \times 16 \times 2$ bits [29].

3. Proposed Algorithm

This section is devoted to describing our contribution to the implementation of the image compression algorithm based on DWT-SVT and optimized BTC method that is given below:

Algorithm

Input : Digital image I and $k(1 \le k < \min(M, N))$.

Output : $I_k \approx I$.

1. Apply 1- discrete wavelet transform to $I \in \mathbb{R}^{M \times N}$.

2. We next quantize high frequency components (cH, cV, cD) by choosing threshold T is median of the absolute value of the detail coefficients.

3. Low-frequency sub-band (cA) compressed by SVT with different k Equation (7).

4. Construct I_k' via inverse discrete wavelet transform with smooth coefficients: low $(\tilde{c}A)$ and high frequency components $(\tilde{c}H, \tilde{c}V, \tilde{c}D)$.

5. Apply BTC method to I_k' of block size: $2 \times 2, 4 \times 4, 8 \times 8, 16 \times 16$.

6. Repeat the step 5 until $1, \ldots, N$ to get I_k.

4. Simulation Results and Discussion

Performance analysis of the proposed algorithms have been discussed in this section. We have used RGB image(s) as shown in Figure 1, for the effectiveness and comparison of the proposed algorithm.

Pepper Lena

Figure 1. Two test RGB images of 512×512 pixels.

Efficiency of the proposed scheme can be evaluated by the following fidelity assessment parameters are Er, Cr, Mean square error (MSE), Peak signal to noise ratio (PSNR) is given by

$$Er = \frac{\| I'_k \|^2}{\| I \|^2} \tag{13}$$

$$qc = \frac{Nz}{Nc} \times 100 \tag{14}$$

$$Cr = \frac{100}{100 - qc} \tag{15}$$

where I'_k represents the compressed image, I is the original image, Nz is the number of zeros and Nc is the number of coefficients [7]. Distortion between the compressed image (I'_k) and the cover image (I) to verify the quality of the approximate image, which is defined as [30]:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} [(i, j) - I'_k(i, j)]^2 \tag{16}$$

$$PSNR = 10 \times \log_{10}\left(\frac{255^2}{MSE}\right). \tag{17}$$

A comparative analysis results of the image quality (PSNR) and compression ratio (Cr) at different threshold k are enlisted in Table 1. The hybrid compression scheme (independently using Haar and Daubechies) compared both transform, obtained outcomes that explain the recovery process to examine the compressed image quality and achieves 99:99% energy retained Equation (13) and image quality retains 30-60 dB of PSNR.

Table 1. Comparative numerical results of image quality parameters tested on pepper image (512×512).

Threshold precision(k)	Haar wavelet			Daubechies Wavelet		
	MSE	PSNR	Cr	MSE	PSNR	Cr
10	204:0567	25:0333	1:5561	210:9051	24:9051	1:4437
50	19:4050	35:2517	1:5183	20:7835	34:9536	1:4166
100	5:5732	40:6698	1:4778	5:6024	40:6471	1:3870
150	3:0416	43:2997	1:4441	2:7763	43:6962	1:3614
200	2:4350	44:2659	1:4131	2:1027	44:9030	1:3389

Figure 2. Approximate image when $k = 100$.

Table 2. Comparative study of proposed scheme outcomes to author [17] when $k = 100$ of Lena's image (512×512).

MPQ-BTC						Proposed Method							
Block size	Average of MSE	PSNR			Average of PSNR	MSE			Average of MSE	PSNR			Average of PSNR
		R	G	B		R	G	B		R	G	B	
2×2	2.81e-4	35.57	36.38	35.48	35.81	0.75	0.25	0.25	0.4166	49.3802	54.1514	49.3802	50.9706
4×4	3.77e-4	33.87	34.71	34.14	34.24	1.75	1.1250	0.75	1.2066	45.7004	47.6193	49.3802	47.566
8×8	5.70e-4	31.77	32.88	32.72	32.46	1.3125	1.2656	5.3125	2.6302	46.9498	47.1078	40.8778	44.9784
16×16	9.57e-4	29.24	30.57	30.93	30.25	7.7422	6.7188	16.5156	15.3082	39.2422	39.8579	35.9519	38.3506

Table 2 shows compared the performance of the proposed BTC compression scheme (using Haar) and obtained approximate image shows in Figure 2. Proposed work compared to Abbadi et al. [17] compressed $I \in \mathbb{R}^{M \times N}$ with SVD (rank equal to 100), then approximate image compressed with MPQ-BTC for different block size: 2×2, 4×4, 8×8, 16×16. However, for each non-overlapping approximate block of I'_k to obtained average MSE (see C_{10}) and PSNR (see C_{14}). Evidently, the above discussion optimized BTC scheme is better performance regarding image quality as per human vision system (HVS). Image quality performance of proposed algorithm by block wise are shows in Figure 3.

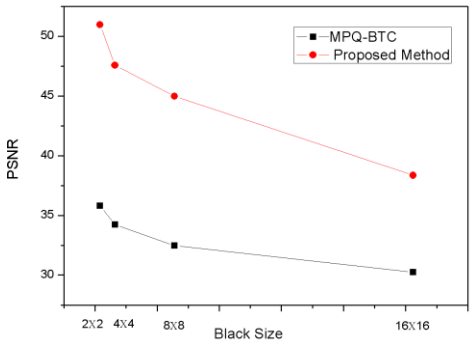

Figure 3. Comparison of algorithms in term of block size and PSNR values of Lena's image. Block size represents on the primary axis and PSNR on the secondary axis.

Proposed scheme maintains a pleasing image quality as compared to MPQ-BTC method (shown in Figure 3). Micro block size (2×2 and 4×4) having better average PSNR value compare to macro block size (8×8 and 16×16). The processing of proposed algorithm(s) using MATLAB R2016a on

a personal computer with i5-2450M CPU @2.50 GHz, 4GB of RAM and 64-bit operating system.

5. Conclusion

In this paper, we proposed a novelty of improved BTC image compression algorithm on both DWT and SVT for compressed JPEG and PNG images. First, proposed scheme comparison between the different wavelet's Haar and Daubechies, numerical outcomes shown in Table 1 was significantly pleasing Cr and without compromising much on the quality of the image (PSNR). Second, comparisons were also made on the SVT in the frequency domain between the proposed improved BTC image compression scheme and Abbadi et al. [17] showed that improved scheme performed significantly promised results of average PSNR values (see Table 2). Those schemes are very simple and do not require complex computations. For future studies, it may be applied to image stenography for hybrid compression images.

References

[1] K. J. Anil, Fundamentals of digital image processing, Englewood Cliffs, NJ: Prentice hall, 1989.

[2] A. M. Rufai, G. Anbarjafari and H. Demirel, Lossy image compression using singular value decomposition and wavelet difference reduction, Digital Signal Processing 24 (2014), 117-123. doi: 10.1016/j.dsp.2013.09.008.

[3] R. Kumar, U. Patbhaje and A. Kumar, An efficient technique for image compression and quality retrieval using matrix completion, Journal of King University-Computer and Information Sciences, (2019). doi: https://doi.org/10.1016/j.jksuci.2019.08.002.

[4] R. M. Harlick and K. Shanmugam, Comparative study of a discrete linear basis for image data compression, IEEE Transactions on Systems, Man and Cybernetics 4(1) (1974), 121-126. doi: https://doi.org/10.1109/TSMC.1974.5408516

[5] A. Skodras, C. Christopoulos and T. Ebrahimi, The JPEG-2000 still image compression standard, IEEE Signal Processing Magazine, 18(5) (2001), 36-58. doi: https://doi.org/10.1109/79.952804

[6] S. J. Lawson and J. Zhu, Image compression using wavelets and JPEG2000: a tutorial, IEEE Electronics and Communication Engineering Journal, 14(3) (2002), 112-121. doi: https://doi.org/10.1049/ecej:20020303

[7] M. Boix and B. Canto, Wavelets transform applications to the compression of images, Mathematical and Computer Modelling, 52(7-8) (2010), 1265-1270. doi: https://doi.org/10.1016/j.mcm.2010.02.019

[8] Y. C. Pang, R. G. Zhou, B. Q. Hu, W. W. Hu and A. E. Rafei, Signal and image compression using quantum discrete cosine transform, Information Sciences 473 (2019) 121-141. doi: https://doi.org/10.1016/j.ins.2018.08.067 10

[9] H. C. Andrews and C. L. Patterson, Singular value decompositions and digital image processing, IEEE Trans. on Acoustics, Speech, and Signal Processing 24 (1976), 26-53. doi: https://doi.org/10.1109/TASSP.1976.1162766

[10] M. Tian, S. W. Luo and L. Z. Liao, An investigation into using singular value decomposition as a method of image compression, Prc. of Fourth Int. Conf. on Machine Learning and Cybernetics 8 (2005), 5200-5204. doi: https://doi.org/10.1109/ICMLC.2005.1527861

[11] H. B. Razafindradina, P. A. Randriamitantsoa and N. R. Razafindrakoto, Image compression with SVD : A new quality metric Based On Energy Ratio, International Journal of Computer Science and Network 5(6) (2016), 960-965.

[12] E. J. Delp and O. R. Mitchell, Image compression using block truncation coding, IEEE Trans. on Communications, 27(9) (1979), 1335-1341. doi: https://doi.org/10.1109/TCOM.1979.1094560

[13] K. Sau, R. K. Basak and A. Chanda, Image Compression based on block truncation coding Using Clifford Algebra, Procedia Technology 10 (2013), 699-706. doi: https://doi.org/10.1016/j.protcy.2013.12.412

[14] J. F. Yang and C. L. Lu, Combined techniques of singular value decomposition and vector quantization for image coding, IEEE Transactions on Image Processing 4(8) (1995), 1141-1146.

[15] A. M. Rufai, G. Anbarjafari and H. Demirel, Lossy medical image compression using Hufiman coding and singular value decomposition, IEEE Signal Processing and Communications Applications Conference (2013), 1-4. doi:10.1109/SIU.2013.6531592.

[16] S. A. Mohamed and M. M. Fahmy, Image compression using VQ-BTC, IEEE Transactions on Communications 43(7) (1995), 2177-2182. doi: https://doi.org/10.1109/26.392959

[17] N. K. E. Abbadi, A. A. Rammahi, D. S. Redha and M. A. Hammeed, Image compression based on SVD and MPQ-BTC, Journal of Computer Science 10(10) (2014), 2095-2104. doi:10.3844/jcssp.2014.2095.2104

[18] S. Lal, M. Chandra and G. K. Upadhyay, A novel image compression algorithm based on discrete wavelet transform with block truncation coding, International Journal of Computer Science and Emerging Technologies 1(2) (2010), 25-34.

[19] S. Mallat, A wavelet tour of signal processing, Elsevier, 1999.

[20] D. K. Ruch and P. J. V. Fleet, Wavelet Theory: An elementary approach with applications, A John Wiley and Sons, INC, 2009.

[21] M. Kumar, D. C. Mishra and R. K. Sharma, A first approach on an RGB image encryption, Optics and lasers in engineering 52 (2014), 27-34.

[22] R. C. Gonzalez and R. E. Woods, Digital image processing, Saddle River, NJ: Prentice Hall, 2008.

[23] I. C. F. Ipsen, Numerical matrix analysis: linear systems and least squares, SIAM, 2009.

[24] G. H. Golub and C. F. Van Loan, Matrix computations. Hindustan Book Agency, 2015.

[25] L. Cai and Z. Hua, SVT: Singular value thresholding in MATLAB, Journal of Statistical Software 81(2) (2017), 1-13. doi: http://dx.doi.org/10.18637/jss.v081.c02

[26] Y. Li and W. Yu, A fast implementation of singular value thresholding algorithm using recycling rank revealing randomized singular value decomposition, arXiv preprint (2017), 1704-05528.

[27] K. Elasnaoui and C. Youness, Two new methods for image compression, International Journal of Imaging and Robotics 15(4) (2015), 1-11.

[28] J. Chen, W. Hong, T. S. Chen, P. A. and C. W. Shiu, Stegnography for BTC compressed images using no distortion technique, International Journal of Computer Science and Network 58 (2010), 177-185.

[29] C. C. Chang and J. C. Chuang, Using a simple and fast image compression algorithm to hide secret information, International Journal of Computers and Applications 28(4) (2015) 329-333. doi: https://doi.org/10.1080/1206212X.2006.11441818

[30] M. Mrak, S. Grgic and M. Grgic, Picture quality measures in image compression systems, IEEE EURCON, (2003), 233-237. doi: 10.1109/EURCON.2003.1248017.12