International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 9, No. 4, August, 2022, pp. 919-927
Journal homepage: http://iieta.org/journals/mmep

Effectiveness of Low-Numerical Rank Approximation to Image Compression in Wavelet Domain

Arikera Padmanabha Reddy[1*], Naveenakumara Uddagatta[2]

[1] Department of Studies in Mathematics, Vijayanagara Sri Krishnadevaraya University, Ballari 583105, Karnataka, India
[2] Department of Mathematics, MVJ College of Engineering, Bengaluru 560067, Karnataka, India

Corresponding Author Email: apreddy@vskub.ac.in

https://doi.org/10.18280/mmep.090408

Received: 5 January 2020
Accepted: 23 August 2021

Keywords:
image compression, TSVD, numerical rank, discrete wavelet transform, PSNR, CR

ABSTRACT

In this paper, we report a modeling of approximation for images by finding numerical rank in the wavelet domain through singular value decomposition of approximation coefficients. Firstly, the digital image is transformed into the frequency domain. Then high-frequency sub-bands are quantized to zero. This is quite obvious in wavelet-based image compression. Simultaneously, the low-frequency sub-bands are compressing by using truncated singular value decomposition (TSVD) through a numerical rank. Finally, reconstruct the approximation matrix via inverse discrete wavelet transform with low computational intricacy. This mathematical model is more adequate for solving engineering problems arises in digital image processing such as the transmission of image (reducing the bandwidth size of a communication channel) and storage capacity (space saving). The simulation results on gray and color images show that there is a gain in: (i) the compression ratio with acceptable visual quality as per human vision system; (ii) balancing of performance measures over conventional SVD methods.

1. INTRODUCTION

With the rapid development of the digital world, extensive applications of multimedia technology and the internet, the huge size of digital image(s) is represented by a matrix ($A \in \mathbb{R}^{m\times n}$). This carries all kind of information's are produced and transmitted over the network [1]. The size of digital multimedia data (images, videos) constantly increasing and these image files can be very large and they require more memory. For example, a grayscale (8-bit) image is 512×512 pixels have 262144 elements to be stored, and a typical RGB (24-bit) image 1024×768 has nearly a million. Therefore, data compression in the form of an image(s) plays a crucial in the era of the digital world.

The problem of image compression is to achieve a high compression ratio (CR) in the digital representation of an input image ($A \in \mathbb{R}^{m\times n}$) with minimum perceived loss of image quality. Ultimately, the criterion of image quality is usually that judged or measured by the human receiver on the basis of human vision system [2]. There are two approaches to compression methods: Lossless and lossy compression. Lossless compression is generally used for medical database management and telediagnosis applications [3] because it cannot tolerate any difference between original and reconstructed data [4]. An image can be lossy-compressed by removing irrelevant information even if the original image does not have any redundancy various methods of lossy image compression algorithms have been proposed [5]. One of the popular methods is singular value decomposition (SVD) which achieves a sequence of good approximation images with low compression ratios. On the other hand, JPEG2000 is based on a discrete wavelet transform (DWT) and it provides a better compression ratio [6]. The author(s) Stoica et al. [7] introduce the integration method of that knowledge for the improvement of perceptual JPEG2000 image compression quality.

During past few years, immense data compression schemes and their applications in digital image processing have been proposed. Numerous efficient lossy image compression techniques [8, 9] motivated us to contribute in this field to combining SVD and DWT in different prospective.

SVD of a matrix proposed by Eugenio Beltrami, Camille Jordan helps to approximate a matrix by lower rank matrix [10]. It is a lossy image compression technique that achieves compression by truncation of smaller singular values to zero [11]. Tian et al. [12] discuss the usage possibility of SVD in image compression. Rufai et al. [13] described image compression carried out by Hoffman coding with SVD which gives the image, better quality with low CR. Khalid et al. [14] introduced two new approaches for image compression. Further, Khalid et al. [15] progressed the compression studies to present an application of linear algebra to image compression using SVD and also recently, proposed by Khalid [16] image compression method based on the block SVD power method (BSPM). Liu et al. [17] to apply sequence image-based fractal compression method to compression three-dimensional MRI images. Xu et al. [18] propose a novel image compression method based on linear regression to the domain of pixel prediction of the image. Bascones et al. [19] designs aimed toward FPGA implementation and focus on the low complexity predictive lossy compression. Bouida et al. [20] an investigation to examine and evaluate image compression degradation by the use of new tendency concept of image quality assessment based on texture and edge analysis.

Wavelet's theory has placed a prominent role in image processing applications (Especially, image compression). It is characterized by high CR and maintains essentially the same characteristics of moments after the compression [21]. The author Mulcahy [22] has developed a compression method using the Haar wavelet transform. Grgic et al. [23] examine a set of wavelet functions for implementation of image compression. Usevitch [24] wavelet-based coding has been adopted as the underlying method to implement the JPEG 2000 standard. Extends investigation of improvement in compression techniques, researcher(s) Yang et al. [25] proposed novel hybrid techniques of SVD and vector quantization for image coding. Rufai et al. [6] also described lossy image compression based on SVD and wavelet difference reduction. Krishnaraj et al. [26] introduce a wavelet transform-based deep learning-based image compression model especially for communication in IoUT.

Based on the study of prevailing image compression techniques, an alternative method is proposed by adopting a relatively less computational complexity method to meet the requirements of getting a superior image consuming a lesser memory.

2. THE HILBERT SPACE $L^2(\mathbb{R})$

In order to create a mathematical model with which to build wavelet transform, it is important that to work in a vector space that lends itself to applications in digital imaging or signal processing. We can view a digital image ($A \in \mathbb{R}^{m\times n}$) as a function of two variables where the function value is the grey-level intensity and the audio signal as functions of time where the function values are the frequencies of the signal. Since rows or columns of A usually are of finite dimension and 1D signals (audio) taper off, we want to make sure that the functions $f(t)$ in space decay sufficiently fast $t \to \pm\infty$. The rate of decay must be fast enough to ensure that the energy of the signal is finite. Finally, it is desirable from a mathematical standpoint to use a space where the inner product of a function with itself is related to the size (norm) of the function [27]. Following is the Hilbert space $L^2(\mathbb{R})$ defined as

$$L^2(\mathbb{R}) = \left\{ f: \mathbb{R} \to \mathbb{C} \mid \int_{\mathbb{R}} |f(t)|^2 \, dt < \infty \right\} \tag{1}$$

with the inner product $f(t)$ and $g(t)$ as $\langle f(t), g(t) \rangle = \int_{\mathbb{R}} f(t)\, \overline{g(t)} dt$.

2.1 Wavelet transform

The familiar Fourier function $f(t)$ is sinusoids of changing frequency and unbounded duration [28] and too much Fourier data is needed to reconstruct the signal locally and it is a useful tool to analyze the frequency component of the signal. In these cases, the wavelet investigation is often very effective because it provides a simple approach for dealing with the local aspects of a signal. Therefore, the wavelet transforms expansion functions are little waves of finite duration and varying frequency. One of the basic and prominent wavelet techniques is Haar and it is a good representation of the image with fewer coefficients [29].

2.2 The Haar function

Haar functions have been utilized from 1910 when they were presented by the Hungarian mathematician Alfred Haar [30].

Haar scaling function defined as

$$\phi(t) = \begin{cases} 1, & 0 \le t < 1 \\ 0, & otherwise. \end{cases} \tag{2}$$

Definition: (The Haar Space V_j) Let $\phi(t)$ be the Haar function in Eq. (2) and $j \in \mathbb{Z}$. We define

$$V_j = span\{\phi(2^j t - l)\}_{l\in\mathbb{Z}} \cap L^2(\mathbb{R}) \tag{3}$$

2.3 Haar wavelet function

The Haar wavelet function $\psi(t)$ is given by

$$\psi(t) = \phi(2t) - \phi(2t-1)\,;$$

$$\psi(t) = \begin{cases} 1, & 0 \le t < \frac{1}{2} \\ -1, & \frac{1}{2} \le t < 1 \\ 0, & otherwise \end{cases} . \tag{4}$$

Definition: (The Haar wavelet Space H_j) Let $\psi(t)$ be the Haar wavelet function in Eq. (4), $j \in \mathbb{Z}$. We define

$$H_j = span\{\psi(2^j t - l)\}_{l\in\mathbb{Z}} \cap L^2(\mathbb{R}) \tag{5}$$

Therefore, the sets $\{\phi_{j,l}(t) = 2^{\frac{j}{2}}\phi(2^j t - l) : l \in \mathbb{Z}\}$ and $\{\psi_{j,l}(t) = 2^{\frac{j}{2}}\psi(2^j t - l) : l \in \mathbb{Z}\}$ are orthonormal basis for $\boldsymbol{V_j}$ and $\boldsymbol{H_j}$ [25]. Scale j determines dilation or the visibility in frequency and l represents translation, $2^{\frac{j}{2}}$ controls their amplitude of the signal. The wavelet subspaces $\boldsymbol{H_j}$ fill the gaps between successive scales:

$$V_{j+1} = V_j \oplus H_j \tag{6}$$

We can start with an approximation on some scale V_0 and then use wavelets to fill in the missing details on finer and finer scales. The finest resolution levels includes all square-integrable functions

$$L^2(\mathbb{R}) = V_0 + \oplus_{j=0}^{\infty} H_j \tag{7}$$

One of the enormous discoveries for wavelet analysis was that perfect reconstruction filter banks could be formed using the wavelet coefficient sequences for 1D or 2D signal.

2.4 Discrete wavelet transform

Suppose n is an even positive integer. We define the discrete Haar wavelet transform (DHWT) matrix as

$$H_n = \left[\frac{W}{G}\right] = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 & & 0 & 0 \\ 0 & 0 & 1 & 1 & & 0 & 0 \\ \vdots & & & & \ddots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 1 \\ \hline 1 & -1 & 0 & 0 & & 0 & 0 \\ 0 & 0 & 1 & -1 & & 0 & 0 \\ \vdots & & & & \ddots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -1 \end{bmatrix} \tag{8}$$

For each decomposition level, it generates the $\frac{n}{2} \times n$ block W is called the averages block and the $\frac{n}{2} \times n$ block G is called the detail block and other family of matrices can be easily obtained using filter coefficients.

The *filter*

$$h = (h_0, h_1) = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right) \tag{9}$$

is called the Haar (low-pass) filter and then we will call

$$g = (g_0, g_1) = \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right), \tag{10}$$

the Haar wavelet (high pass) filter. Similarly, we can obtain discrete higher order Daubechies wavelet transformation matrix obtained as [27]

$$H_n = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & \cdots & h_{L-1} & h_L & \cdots & 0 & 0 \\ 0 & 0 & h_0 & h_1 & \cdots & h_{L-3} & h_{L-2} & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & h_{L-5} & h_{L-4} & \cdots & 0 & 0 \\ & & & & \ddots & & & \ddots & & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & h_{L-1} & h_L \\ h_{L-1} & h_L & 0 & 0 & \cdots & 0 & 0 & \cdots & h_{L-3} & h_{L-2} \\ & & & & \ddots & & & \ddots & & \\ h_2 & h_3 & h_4 & h_5 & \cdots & 0 & 0 & & h_0 & h_1 \\ \hline g_0 & g_1 & g_2 & g_3 & \cdots & g_{L-1} & g_L & \cdots & 0 & 0 \\ 0 & 0 & g_0 & g_1 & \cdots & g_{L-3} & g_{L-2} & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & g_{L-5} & g_{L-4} & \cdots & 0 & 0 \\ & & & & \ddots & & & \ddots & & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & g_{L-1} & g_L \\ g_{L-1} & g_L & 0 & 0 & \cdots & 0 & 0 & \cdots & g_{L-3} & g_{L-2} \\ & & & & \ddots & & & \ddots & & \\ g_2 & g_3 & g_4 & g_5 & \cdots & 0 & 0 & & g_0 & g_1 \end{bmatrix} \tag{11}$$

Here, $L = 2l$ where l is natural number. For l=1, we get Haar matrix.

The matrix H_n is orthogonal and still essentially computes averages and differences. The benefit, of course, is that we have an easy formula for the inverse:

$$H_n^{-1} = H_n^T \tag{12}$$

A two-dimensional DHWT of a discrete image A can be performed whenever an even number of rows (m) and an even number of columns (n). 1-level wavelet transform of an image A is defined as

$$\tilde{A} = H_m A H_n^T, \tag{13}$$

$$\tilde{A} = \left[\frac{W}{G}\right] A \left[\frac{W}{G}\right]^T, \tag{14}$$

$$\tilde{A} = \left[\frac{LL}{LH} \,\Big|\, \frac{HL}{HH}\right]. \tag{15}$$

For each level of decomposition, wavelet coefficients are decimated by a factor two, which achieves better compression ratio [31]. In Haar wavelet decomposition produces, approximate wavelet co-efficient (*LL*), horizontal (*HL*), vertical (*LH*) and diagonal (*HH*) detail coefficients as in Eq. (15). Then block matrix (*LL*) compressed by SVD method and other detail coefficients block(s): *LH, HL, HH* truncated to zero [32].

Remark: When an image A has an odd number of m or n, it can be extended by appending 0's so that it has an even number of rows and columns.

3. SINGULAR VALUE DECOMPOSITION

Let $A \in \mathbb{R}^{m \times n}$. The full space and comlun space of A and A^T provide sufficient details to understand A. Basis and dimension of each subspce can be obtained by knowing its SVD. SVD of A is given in the following [33].

$$A = \begin{cases} P\left(\frac{\Sigma}{0}\right)Q^T, & \text{Where } \Sigma = diag(\sigma_1, \cdots, \sigma_n), \\ & \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0, \ \ if \ \ m \geq n, \\ p(\Sigma \ \ 0)Q^T, & \text{Where } \Sigma = diag(\sigma_1, \cdots, \sigma_m), \\ & \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_m \geq 0, \ \ if \ \ m \leq n. \end{cases} \tag{16}$$

Here $P = [p_1, p_2, \cdots, p_m] \in \mathbb{R}^{m\times m}$ and $Q = [q_1, q_2, \cdots, q_n] \in \mathbb{R}^{n\times n}$ 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 A and denoted by $rank(A)$ [11].

$$A = \sum_{i=1}^{rank(A)} \sigma_i p_i q_i^T \tag{17}$$

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, because it supports psychovisual redundancy related to human eye [6].

3.1 Optimality of SVD

We consider the SVD of matrix A of size $m\times n$ as in Eq. (16). For any k with $1 \le k < r = rank(A)$, then define.

$$A_k = \sum_{i=1}^{k} \sigma_i p_i q_i^T \tag{18}$$

Ekart-Young theorem says A_k is the best approximation to A among all $k - rank$ matrices with respect to Frobenius norm [34]. This result has been used to compress an image (matrix) A of size $m \times n$ has initially $m \times n$ entries to store in computer memory disk space and measured in kilobytes (KB). If we consider A_k , instead of A, then we have an approximation of A which can be stored with $k(m + n + 1)$ values in spatial domain, i.e., the entries of the vectors p_i, q_i and singular values $\sigma_i, i = 1,2,\cdots,k$. For the point of image compression choosing k value is a challenging task and this leads to limitation of SVD. Khalid et al. [14] considered new upper bound for k in terms of size of original image instead of its rank,

$$\text{i.e., } k \le \frac{mn}{m+n+1} \tag{19}$$

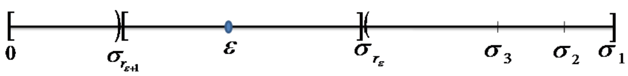
According to Eq. (19) choosing k value has a lot of choices between 1to $\frac{mn}{m+n+1}$. The concept of numerical rank of matrix help us to finalize suitable value for k.

3.2 Estimating numerical rank

In matrix computations, the numerical rank of $A \in \mathbb{R}^{m\times n}$, can be defined as the number of singular values greater than a specified tolerance $\varepsilon > 0$ and it is denoted by r_ε [35].

The singular values of matrix A by Eq. (16) with the numerical r_ε satisfy

$$\sigma_{r_\varepsilon+1} \le \varepsilon < \sigma_{r_\varepsilon} \tag{20}$$



It is important to note that the notion of numerical rank r_ε is useful only when there is a well-defined gap between σ_{r_ε} and $\sigma_{r_\varepsilon+1}$ [36], as well. Then decaying singular values of A to get low numerical rank and its needful to SVD compression method [8]. Clearly, the little value of $\varepsilon > 0$, we get A_{r_ε} with rank r_ε such that $\|A - A_{r_\varepsilon}\|_2 = \sigma_{r_\varepsilon+1} < \varepsilon$.

3.3 Methodology to image compression

SVD methods heavily depend on a small number of dominant singular values to reconstruct the image, because there is a large number of singular values close to zero represents redundancy information of the digital image. A small number of dominant singular values are closely associated with numerical rank (r_ε) for a specified tolerance $\varepsilon > 0$. This fact is utilized for finding r_ε in the wavelet domain because dominant singular values of the original image are found in SVD of its smooth / average coefficients.These facts are presented in the form of an algorithm:

ALGORITHM:

Input: Digital image A and tolerance $\varepsilon(0 < \varepsilon < \sigma_1)$
Output: $B \approx A$ with $rank(B) = r_\varepsilon$.
1. Apply 1-level discrete Haar wavelet transform to A to obtain $\tilde{A} = H_m A H_n^T$.
2. We next quatize $\tilde{A}$ by replacing each HL, LH and HH by the $\frac{m}{2} \times \frac{n}{2}$zero matrix, i.e.,

$$\tilde{A}_q = \left[\begin{array}{c|c} LL & 0 \\ \hline 0 & 0 \end{array}\right].$$

3. Compute r_ε from SVD of LL. Here LL represents coaser representation of $\tilde{A}$,
4. Obtain the best r_ε rank matrix which approximations LL i.e., $\tilde{LL} = \sum_{i=1}^{r_\varepsilon} \sigma_i p_i q_i^T$ by Eq. (18).
5. Construct B by applying 1-level inverse Haar wavelet transform with smooth coefficients as $\tilde{LL}$ and all other detail coefficients are zero.

---

4. IMAGE QUALITY PARAMETERS

A picture quality in image compression systems is important phase in describing the type and amount of degradation in reconstructed image. Among many objective numerical measures of image quality, that are based on comparable distortion measures is defined as follows:

4.1 Mean-square error (MSE) and peak signal-to-noise ratio (PSNR)

Let $A \in \mathbb{R}^{m\times n}$be a given digital image and the approximate matrix $B \in \mathbb{R}^{m\times n}$. To evaluate MSE and $PSNR$ between A and B defined as [37].

$$MSE(A,B) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij} - \tilde{b}_{ij}|^2}{mn} \tag{21}$$

$$PSNR(A,B) = 10\log_{10}\left(\frac{(L^{'}-1)^2}{MSE(A,B)}\right), \quad (22)$$

$$PSNR(A,B) = 10\log_{10}\left(\frac{(L^{'}-1)^2}{\|A-B\|_F^2}\right) \quad (23)$$

where, $L^{'}$ is the number of gray levels.The unit of $PSNR$ is decibel (dB). MSE is inversely proportional to $PSNR$. Smaller values of MSE means better approximation.

4.2 Compression ratio

Compresssion ratio defined as the ratio of size of original image to the size of compressed image.

$$CR(A,B) = \frac{\text{Required memory space to store entries of sample image } A}{\text{Required memory space to store entries of sample image } B}. \quad (24)$$

Vigor of image compression is measured by CR.

5. RESULTS AND DISCUSSION

We implemented the proposed algorithm using MATLAB R2009a on machine with: i5-2450M CPU @2.50 GHz, 4GB RAM and 64-bits operating system.

An innovative study of image compression with respect to numerical rank (r_ε) constructed in the wavelet domain for a given tolerance ε is presented. For comparison purposes, we have designed/considered various tolerance values in such a way that the proposed algorithm produces a platform to compare other related works. It is implemented on test images and obtained observations are noted in Tables and Figures.

Table 1 is showing the comparision between the proposed technique test with grayscale image pigeon and spatial domain (SVD alone method). In order to see the performance of the method is test with fruits image shown in Table 2.

Table 3 reveals that proposed algorithm meets the reasonable MSE (C_2&C_7) and PSNR (C_3&C_8) with high CR (C_4&C_9). The behavior of compression ratio for Lena image with respect to tolerances as shown in Figure 1.

The performance of proposed routine as compared to scheme BSPM for color image (Desert.jpg, 1024×768) and results are shown in Table 4.

Remark: In the Table 4, *PSNR* value (see column 2) to mention 92.97 for $MSE = 0.36$, its apparently correct answer is 52.56 [16].

We designed various tolerances for three different channels ($\varepsilon_R, \varepsilon_Gand\varepsilon_B$) to match r_ε for comparison with existing methods Khalid et al. [15] and Khalid [16]. Observations in Table 4 report that proposed algorithm achieves a higher compression ratio by reasonably balancing PSNR. Also, PSNR values are increased as numerical rank (r_ε) increases (see C_7&C_8 in Table 4). In the context of a human visual comparison between sample image(s) and compressed image(s) as shown in Figure 2.

Visual analysis of compressed images obtained from the proposed compression methodology is illustrated in Figure 2. Original (a, b, c and d) and compressed images (e, f, g and h) have been shown, which illustrates the compressed images with fewer amounts of data. Different visual characteristics are tested with different images. From this analysis, illustration clearly demonstrates the proposed method is evident from simulation results that better compression is achieved with acceptable visual quality (as per HVS) as compared to conventional SVD methods [9].

Table 1. PSNR and CR values for SVD and proposed mathod on pigeon image (768×512)

SVD Compression				Proposed method				
k	*MSE*	*PSNR*	*CR*	*ε*	*r_ε*	*MSE*	*PSNR*	*CR*
08	0.0118	67.4267	38.3700	20.1	08	0.0119	67.3785	76.6802
16	0.0065	70.0304	19.1850	11.1	16	0.0067	69.8859	38.3401
32	0.0036	72.5418	9.5925	5.7	32	0.0040	72.1188	19.1700
64	0.0019	75.2900	4.7963	3.0	64	0.0026	74.0490	9.5850
128	0.0008	78.9971	2.3981	1.4	128	0.0018	75.4607	4.7925

Table 2. Comparison of SVD techniques with proposed method using fruits image (512×512)

SVD Compression							Proposed method			
k	*MSE*	*PSNR*	*CR*	ε_R	ε_G	ε_B	r_ε	*MSE*	*PSNR*	*CR*
08	0.0185	65.4489	1.7838	14.1	13.5	13.5	08	0.0077	69.2828	3.9741
16	0.0150	66.3756	1.5503	7.0	7.1	7.4	16	0.0044	71.7090	3.4402
32	0.0114	67.5643	1.3509	4.0	4.1	4.5	32	0.0029	74.2223	3.0733
64	0.0088	68.6738	1.2000	1.9	2.0	2.1	64	0.0014	76.6132	2.8456
128	0.0017	75.7361	1.0974	0.71	0.79	0.76	128	0.0010	78.1028	2.7604

Table 3. Obtained numerical results of image quality parameters tested on Lena image (512×512) and comparison with the work of Tian et al. [12]

SVD Compression				Proposed method				
k	*MSE*	*PSNR*	*CR*	ε	r_ε	*MSE*	*PSNR*	*CR*
08	0.0070	69.6843	31.48186	14.1	08	0.0071	69.6333	63.8752
16	0.0038	72.3836	15.98439	8.1	16	0.0039	72.2171	31.9376
32	0.0018	75.6716	7.992195	3.9	32	0.0020	75.0684	15.9688
64	0.0006	80.0204	3.996098	1.9	64	0.0011	77.7585	7.8744
128	0.0001	86.2687	1.998043	0.61	128	0.0007	79.4549	3.9922

Table 4. Comparision of obtained outcomes against block SVD power method (BSPM) for various tolerances [15, 16]

BSPM			Proposed method					
k	$PSNR$	CR	ε_R	ε_G	ε_B	r_ε	$PSNR$	CR
50	48.44	6.98	6.38	2.61	1.12	50	75.5155	10.8256
100	51.20	6.94	3.65	1.57	0.69	100	77.2613	9.6046
122	92.97(52.56)	6.92	3.02	1.31	0.6	122	77.7329	9.2808
150	53.60	6.92	2.36	1.07	0.5	150	78.1662	9.1879
200	54.43	6.92	1.51	0.71	0.36	200	78.6212	9.0471
250	58.88	6.89	0.93	0.44	0.24	250	78.8227	8.9978
300	63.26	6.88	0.47	0.24	0.15	300	78.8946	8.5507
350	67.19	6.87	0.17	0.095	0.08	350	78.9100	8.5243

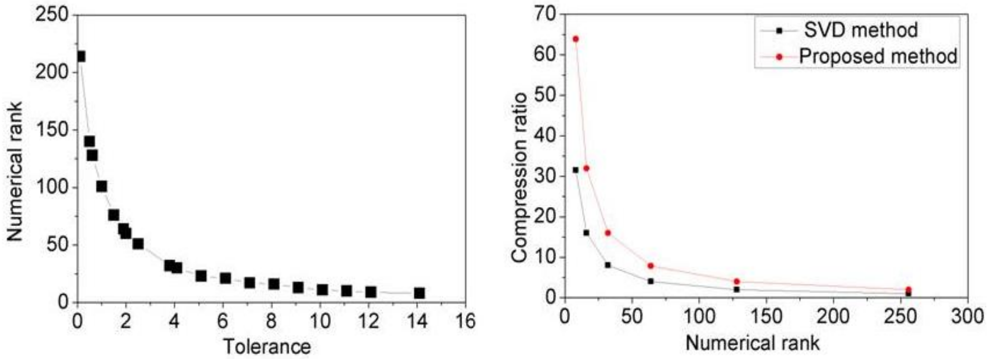


Figure 1. Variation of compression ratio (CR) with different numerical rank (r_ε) for Lena image

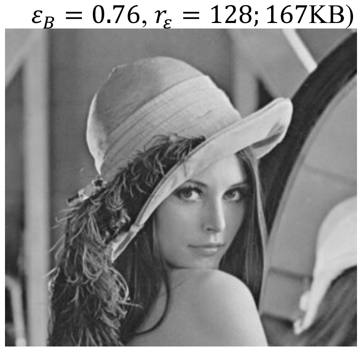
a (pigeon, 234 KB)

e ($\varepsilon = 1.4$, 85.5KB)

b (fruits, 461KB)

f ($\varepsilon_R = 0.71$, $\varepsilon_G = 0.79$, $\varepsilon_B = 0.76$, $r_\varepsilon = 128$; 167KB)

c (Lena, 148 KB)



g ($\varepsilon = 0.6$, $r_\varepsilon = 128$; 136 KB)

d (Desert, 826 KB)

h ($\varepsilon_R = 1.52$, $\varepsilon_G = 0.71$, $\varepsilon_R = 0.36$, $r_\varepsilon = 200$; 91.3KB)

Figure 2. Human visual comparison of a, b,c,d are sample images and e,f,g, h are after compression with suitable ε and acquired memory space in KB (Kilobytes)

6. COMPUTATIONAL INTRICACY

Let A be a given digital image of size $m \times n$. In the case of SVD, the required complexity to compute the compressed image with numerical rank r_ε, it found to be $O(m \times n \times min\{m, n\})$ [38]. Whereas the proposed algorithm requires half of both because we are applying SVD on a matrix size $\frac{m}{2} \times \frac{n}{2}$. For this half (50%) reduction, the proposed method has to pay $O(m \times n)$ [12] operations to obtain the wavelet (Haar takes fewer operations compare to other wavelets) transform of A. Whenever the image database is large, proposed algorithm places a crucial role in reducing complexity (hence CPU time) and memory compared to conventional SVD.

The CPU time is taken to construct compressed image with various numerical rank (r_ε) corresponding to conventional SVD and the proposed method for Lena image is shown in the following Figure 3. In this figure, x and y axes represent the numerical rank and corresponding processing time in seconds respectively. Further, this figure demonstrates the above discussed reduction in half computational process characterized in terms of CPU time concerned with the work of Tian et al. [12] presented a SVD alone method.

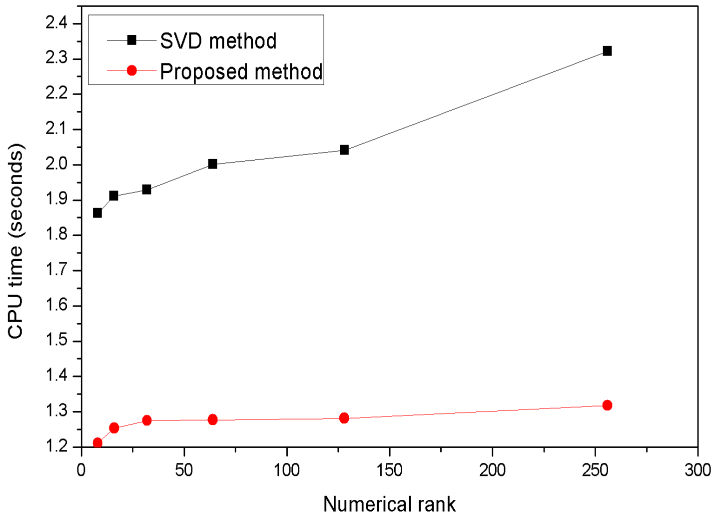


Figure 3. Shows the plot of CPU time rate across the numerical rank

7. CONCLUSIONS

The algorithm proposed with the significant role of numerical rank (r_ε) for image compression in the wavelet domain. It has two advantageous over conventional SVD and its variants: (i) the comparison clearly illustrates that the proposed model achieves high compression ratio with tolerable visual quality as per human vision system; (ii) less computational complexity because of fast wavelet transform. Test images considered in results and discussion act as testimonials to demonstrate the features of the proposed algorithm over the conventional SVD methods. Haar wavelet transform outperforms to reduce the computational processing time and speeds up the compression and decompression process as compared to the higher-order level of wavelet families. The proposed algorithm applied to video compression justifies the high computational complexity. Future researches may lead to continuing on developing more effective transform-based algorithms and explore a better performance on data compression.

REFERENCES

[1] Jain, A.K. (1989). Fundamentals of digital image processing. Englewood Cliffs, NJ: Prentice Hall.

[2] Jayant, N., Jounson, J., Safranek, R. (1993). Signal compression based on models of huaman perception. Proc, IEEE, 81(10): 1385-1422. https://doi.org/10.1109/5.241504

[3] Munteanu, A., Cornelis, J., Cristea, P. (1999). Wavelet-based lossless compression of coronary angiographic images. IEEE Transactions on Medical Imaging, 18(3): 272-281. https://doi.org/10.1109/42.764904

[4] Sayood, K.. (2006). Introduction to Data Compression. Elsevier, Third Edition.

[5] Salomon, D. (2007). Data compression. Springer, Fourth Edition.

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

[7] Stoica, A., Larabi, M.C., Fernandez-Maloigne, C. (2004). Amélioration de la qualité visuelle d'images couleur dans le cadre du standard de compression JPEG2000 - Visual quality enhancement for color images in the framework of the JPEG2000 compression standard. Traitement du signal, 21(6): 661-677.

[8] Ashino, R., Morimoto, A., Nagase, M., Vaillancourt, R. (2005). Comparing multiresolution SVD with other methods for image compression. Advance in Analysis, pp. 457-470. https://doi.org/10.1142/9789812701732_0042

[9] Kumar, R., Patbhaje, U., Kumar, A. (2019). An efficient for image compression and quality retrieval using matrix completion. Journal of King Saud University-Computer and Information Sciences, 1-9. https://doi.org/10.1016/j.jksuci.2019.08.002

[10] Stewart, G.W. (1993). On the early history of the singular value decomposition. SIAM Review, 35(4): 551-566. https://doi.org/10.1137/1035134

[11] Golub, G.H., Van Loan, C.F. (2015). Matrix Computations. Hindustan Book Agency.

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

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

[14] Khalid. E.A., Youness, C. (2015). Two new methods for image compression. International Journal of Imaging and Robotics, 15(4): 1-11.

[15] Khalid, E.A., Ouhda, M., Aksasse, B., Ouanan, M. (2018). An application of linear algebra to image compression. Homological and Combinations Methods, Springer Proceedings in Mathematics and Statistics, 228: 41-54. https://doi.org/10.1007/978-3-319-74195-6_4

[16] Khalid E.A. (2019). Image compression based on block SVD power method. Journal of Intelligent Systems, 1-15. https://doi.org/10.1515/jisys-2018-0034

[17] Liu, S., Bai, W., Zeng, N., Wang, S. (2019). A fast fractal based compression for MRI images. IEEE Access, 7: 62412-62420. https://doi.org/10.1109/ACCESS.2019.2916934

[18] Xu, S., Chang, C.C., Liu, Y. (2020). A novel image compression technology based on vector quanisation and linear regression pridiction. Connection Science, 33(6): 1-18. https://doi.org/10.1080/09540091.2020.1806206

[19] Bascones, D., Gonzalez, C., Mozos, S. (2020). An extremely pipelined FPGA implementation of a lossy hyperspectral image compression algorithm. IEEE Transactions on Geoscience and Remote Sensing, 58(10): 7435-7447. https://doi.org/10.1109/TGRS.2020.2982586

[20] Bouida, A., Beladgham, M., Bassou, A., Benyahia, I., Ahmed-Taleb, A., Haouam, I., Kamline, M. (2020). Evaluation of textural degradation in compressed image texture feature and edges. Traitement du Signal, 37(5): 753-762. https://doi.org/10.18280/ts.370507

[21] Hu, S.G., Wu, X.F., Xi, Z.F. (2012). Application of 2-D wavelet transform in image compression based on Matlab. In Computer, Informatics, Cybernetics and Applications, Springer, Dordrecht, 491-499. https://doi.org/10.1007/978-94-007-1839-5_52

[22] Mulcahy, C. (1997). Image compression using the Haar wavelet transform. Spelman Science and Mathematics Journal, 1(1): 22-31.

[23] Grgic, S., Grgic, M., Cihar, B.Z. (2001). Performance analysis of image copression using wavelets. IEEE Transactions on Industrial Electronics, 48(3): 682-695. https://doi.org/10.1109/41.925596

[24] Usevitch, B.E. (2001). A tutorial on modern lossy wavelet image compression: Foundations of JPEG 2000. IEEE Signal Processing Magazine, 18(5): 21-35. https://doi.org/10.1109/79.952803

[25] Yang, J.F., Lu, C.L. (1995). Combined techniques of singular value decomposition and vector quantization for image coding. IEEE Transactions on Image Processing, 4(8): 1141-1146. https://doi.org/10.1109/83.403419

[26] Krishnaraj, N., Elhoseny, M., Thenmozhi, M., Selim, M.M., Shankar, K. (2020). Deep learning model for real-time image compression in internet of underwater things (IoUT). Journal of Real-Time Image Processing, 17(6): 2097-2111. https://doi.org/10.1007/s11554-019-00879-6

[27] Ruch, D.K., Fleet, P.J.V. (2009). Wavelet theory: An elementary approach with applications. A John Wiley & Sons, INC.

[28] Gonzalez, R.C., Woods, R.E., Eddins, S.L. (2016). Digital Image Processing Using MATLAB. McGraw Hill Eucation.

[29] Dan, L., Li, J., Gu, X., Liao, J., Zhan, S. (2006). The application of image compression based on discrete wavelet transform. In Wavelet Active Media Technology and Information Processing, 2: 291-298. https://doi.org/10.1142/9789812772763_0045

[30] Stanković, R.S., Falkowski. B.J. (2003). The Haar wavelet transform: Its status and achievements. Computers and Electrical Engineering, 29(1): 25-44. https://doi.org/10.1016/S0045-7906(01)00011-8

[31] Kumar, R.N., Jagadale, B.N., Bhat, J.S. (2019). A lossless image compression algorithm using wavelets and fractional fourier transform. SN Applied Sciences, 1(3): 1-266. https://doi.org/10.1007/s42452-019-0276-z

[32] Fleet, P.J.V. (2011). Discrete Wavelet Transformations: An Elementary Approach with Applications. John Wiley

and Sons.
[33] Ipsen, I.C.F. (2009). Numerical Matrix Analysis: Linear Systems and Least Squares, SIAM.
[34] Trefethen, L.N., Bau III, D. (1997). Numerical Linear Algebra (Vol. 50). SIAM.
[35] Foster, L.V., Davis, T.A. (2013). Algorithm 933: Reliable calculation of numerical rank, null space bases, pseudoinverse solutions, and basic solutions using suitesparseQR. ACM Transactions on Mathematical Software, 40(1): 1-24. https://doi.org/10.1145/2513109.2513116
[36] Ubaru, S., Saad, Y. (2016). Fast methods for estimating the numerical rank of large matrices. In International Conference on Machine Learning, 48: 468-477.
[37] Mrak, M., Grgic, S., Grgic, M. (2003). Picture quality measures in image compression systems. In The IEEE Region 8 EUROCON 2003. Computer as a Tool., 1: 233-236. https://doi.org/10.1109/EURCON.2003.1248017
[38] Liu, W., Lin, W. (2011). Additive white Gaussian noise level estimation in SVD domain for images. IEEE Transactions on Image Processing, 22(3): 872-883. https://doi.org/10.1109/TIP.2012.2219544