International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August, 2019

Modified Firefly Algorithm for Vector Quantization Codebook Design in Image Compression

D. Preethi, D. Loganathan

***Abstract**: In the recent days, the importance of image compression techniques is exponentially increased due to the generation of massive amount of data which needs to be stored or transmitted. Numerous approaches have been presented for effective image compression by the principle of representing images in its compact form through the avoidance of unnecessary pixels. Vector quantization (VA) is an effective method in image compression and the construction of quantization table is an important process is an important task. The compression performance and the quality of reconstructed data are based on the quantization table, which is actually a matrix of 64 integers. The quantization table selection is a complex combinatorial problem which can be resolved by the evolutionary algorithms (EA). Presently, EA became famous to resolve the real world problems in a reasonable amount of time. This chapter introduces Firefly (FF) with Teaching and learning based optimization (TLBO) algorithm termed as FF-TLBO algorithm for the selection of quantization table and introduces Firefly with Tumbling algorithm termed as FF-Tumbling algorithm for the selection of search space. As the FF algorithm faces a problem when brighter FFs are insignificant, the TLBO algorithm is integrated to it to resolve the problem and Tumbling efficiently train the algorithm to explore all direction in the solution space. This algorithm determines the best fit value for every bock as local best and best fitness value for the entire image is considered as global best. When these values are found by FF algorithm, compression process takes place by efficient image compression algorithm like Run Length Encoding and Huffman coding. The proposed FF-TLBO and FF-Tumbling algorithm is evaluated by comparing its results with existing FF algorithm using a same set of benchmark images in terms of Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR), Signal to Noise Ratio (SNR). The obtained results ensure the superior performance of FF-TLBO and FF-Tumbling algorithm over FF algorithm and make it highly useful for real time applications.*

***Index Terms**: Bio-inspired algorithm, DCT, image warping, SIFTS matching, Transform coding*

Revised Manuscript Received on October 30, 2019.
* Correspondence Author
Ms. D. Preethi*, Dept. of Computer Science and Engineering, Pondicherry Engineering College, Puducherry.
Dr. D. Loganathan, Professor, Dept. of Computer Science and Engineering, Pondicherry Engineering College, Puducherry.

I. INTRODUCTION

Image compression is highly essential for effective storage and transmission of images. The need of communication via telecommunication network and accessing the multimedia data using Internet is tremendously increasing. The advancements in the digital camera lead to the generation of larger size images requires more communication bandwidth for transmission and massive amount of memory for storage[1]. It is useful in various applications such as medical imaging, satellite imaging, teleconferencing, etc. For instance, a medium size color image (512x512 pixels) needs a storage area of 0.75megabytes (MB); 35mm digital slide with 12µm resolution consumes a storage area of 18MB. A 12-bit X-ray image of 2048x2560 pixels requires a storage area of 13MB. A 16-bit mammogram image of 4500x4500 pixels requires 40megabytes of disk storage. The original video for 1 second needs around 20MB of storage space. So, image compression techniques are developed to effective store and transmit data to utilize the available resources in an efficient manner. Basically, data compression techniques work on the principle of eliminating repeated and unwanted data[2]. As the images are composed of pixels, image compression techniques are based on the idea of removing redundant and irrelevant pixels in an image[3]. For highly correlated images, better compression performance can be achieved when compared to less correlated images. Generally, image compression techniques are classified to lossy and lossless compression techniques[4]. The compression of images with no loss of information comes under lossless compression, which is useful in applications where loss of information is not bearable[5]. Medical imaging and satellite imaging follows the concept of lossless image compression. By Contrast, sometimes, the loss of information in an image during compression is tolerable in various applications. In multimedia, graphics and browsing internet, lossy compression techniques can be utilized. On the other hand, image compression techniques can be partitioned to predictive and transform coding techniques. In predictive coding, the existing information can be utilized for the prediction of upcoming data, and the obtained difference is encoded. This type of coding is simple, easier to implement and can be adaptable to different local image features. Next, transform coding, converts an image from one kind of representation to another and the transformed values (coefficients) are encoded by compression techniques. The performance of transform coding is much higher than predictive coding but at the cost of high computational complexity.

An image compression model under transform coding comprises of three components namely transformer, quantizer and encoder. It is a reversible, liner mathematical transform which maps the pixels to a set of coefficients, which undergoes quantization and encoding process. Transform coding techniques partitions the reference image to sub-images (blocks) of smaller sizes (8*8). Then, the transform coefficients are determined for each block, efficiently transforming the reference 8*8 array of pixel values to an array of coefficients inside which the coefficients at the top left corner contains more information, which needs to be quantized and encoded with less distortion. The resultant coefficients are quantized and symbol encoding methods are employed to achieve output bit stream, which represents the compressed image. During decompression, at the decoder size, the reversible operation takes place. The advantage of transform coding is that most of the resultant coefficients of natural images have smaller magnitude, which can be easily quantized with no distortion in the reconstructed image. A better transform coding technique has the ability to compress images using less number of coefficients. DCT and DWT are the most widely used type of transform coding techniques [6]. DCT is a popular image compression technique used in JPEG. It partitions an image into various portions of distinct frequencies where less significant parts are removed by quantization and significant frequencies are employed to reconstruct the image in the decompression process. DCT has many benefits: easily implemented to an IC, capability to store information in lesser number of coefficients and reduces the blocking artifacts[7].

Quantization is an important source of compression along with some loss of information. An important characteristic of JPEG is that diverse levels of image compression and quality can be achieved by the selection of quantization tables. Consequently, the quality ratio can be altered based on the application requirements. Every coefficient in the 8x8 DCT matrixes is divided by a weight present in the quantization table and less important DC coefficients are eliminates. When all the weights are found to be 1, the transformation process involves zero compression. JPEG suggested quantization table for brightness component, which is available in the information annexure of the JPEG standard. Table 1.1 shows the quality level of 50 quantization matrix which provides better compression with high reconstructed data quality. The user can choose the quantization level ranges between 1 to 100 where 1 implies worst image quality with best compression performance and vice versa.

For every application, the quantization table may vary and a universal quantization table applicable for all application is not available. So, the selection of quantization table is a combinatorial optimization problem which can be solved by meta-heuristic algorithms. Several methods like statistical and metaheuristic algorithms have been proposed to determine the transform coefficients in DCT for image compression[8], [9], [10], [11]. Some of the metaheuristic algorithms used in image compression techniques are particle swarm optimization (PSO)[12], quantum particle swarm optimization (QPSO)[13], genetic algorithm (GA)[14], differential evolution (DE)[15], honey bee mating optimization (HBMO)[16], pollination based optimization (PBO)[17], cuckoo search optimization [18]and so on.

The contribution of the chapter is summarized as follows: This chapter presents a Firefly (FF) with Teaching and learning based optimization (TLBO) algorithm termed as FF-TLBO algorithm for the selection of quantization table. The efficiency of the proposed is validated using a set of benchmark images. The obtained results are compared with PBO method in terms of Mean Square Error (MSE), Root mean square error (RMSE), Peak signal to noise ratio (PSNR), Signal to Noise Ratio (SNR).

Table 1.1 Quantization Matrix Q50

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	130	99

The succeeding part of the chapter is arranged as follows: Section 2 reviews the existing meta-heuristic based image compression techniques in detail. The basic concepts of DCT and FF algorithm are given in section 3. Section 4 presents the proposed FF-TLBO and FF-Tumbling algorithm with necessary steps and diagrams.

The validation of the proposed method and discussion of results takes place in section 5. Finally, in section 6, the chapter is ended with concluding remarks and future studies.

II. RELATED STUDIES

Numerous lossless image compression methods are proposed by the utilization of mathematical models and metaheuristic algorithms. As the proposed method employs one of the bio-inspired algorithms, the image compression techniques based on optimization algorithms are reviewed and the comparison is tabulated in Table 2. A fast fractal encoding technique using PSO algorithm is presented in [19], to decrease the amount of time required for the encoding process. The usage of PSO algorithm increases the fractal encoding speed and also conserves the image quality. This method is tested on medical images and the results are analyzed interms of encoding time and PSNR. An author in [20] used HBMO algorithm to build the codebook of vector quantization (VQ). It produces reliable results with higher quality when compared to conventional Linde–Buzo–Gray (LBG)[21], PSO-LBG and QPSO-LBG algorithms. Thepresented HBMO–LBG method resulted to the construction of better codebook with smaller distortions. A firefly (FF) algorithm is also used in [22]to build the codebook of VQ. The author employed LBG algorithm as the initialization of FF algorithm to design VQ algorithm. The presentedFF-LBGtechnique implemented VQ and improves the results of LBG method. The simulation results of FF-LBG is compared with LBG, PSO, QPSO and HBMO algorithms using MSE, PSNR and bit rate. The FF-LBG algorithm is found to be faster and attain better quality than LBG, PSO and QPSO. However, it showed no superior performance than HBMO algorithm. In the year 2013, [23] integrated Super-Spatial Structure Prediction with inter-frame coding to attain better CR. At first, Super-Spatial Structure Prediction algorithm is employed with a fast block-matching process (Diamond Search method). Head code compression algorithm is used to further enhance the CR. The proposed method is evaluated on medical images and found that it outperforms JPEG-LS interms of CR.A histogram based image compression method using multi-level image thresholding is presented in [24]. The gray scale image is split to crisp group of probabilistic partitions. Shannon's Entropy is utilized to calculate the level of randomness of the crisp grouping.The entropy function is maximized by Differential Evolution (DE) to decrease the computation complexity and standard deviation of optimizedobjective value. The proposed method is experimented by the use of benchmark images fromUC Berkeley and CMU dataset. The simulated results verify that the proposed DE algorithm is efficient when compared to PSO and GA.

Ming-Sheng Wu 2014 presented a GA based DWT model to reduce the fractal encoding time[25]. Initially, at every range blocks, two wavelet coefficients areemployed to determine the fittest Dihedral block of the domain block. Next, DWT is embedded to GA to attain fast encoding and maintains better image retrieval quality.This GA method operates at much faster rate than full search technique, but at the cost of relatively acceptable reconstructed image quality. In [26], a lossless image compression approach is presented by incorporating integer wavelet transform (IWT) with prediction step. Initially, the transformation of image takes place and a difference image is produced. Then, the difference image is passed to IWT and computes the transform coefficients employed in the lossless codeword assignment. This method attained better compression performance and the computational complexity is mostly nearer to its competitors. Omari and Salah Yaichi[27]exploited the relativity between fractional numbers and their respective quotient representation. Every individual sub-image is mapped to a fractional number by RGB representation and then decreased to an effective quotient. This technique reported better CR, when the least significant bits of every bytes is changed, hence, the image quality is saved with higher CR.HarpeetKaru et al. devised lossless image compression technique for compressing significant parts by the extraction of a region of interest in DICOM images[28]. Then, Huffman coding is used to compress the extracted region and GA further enhances the compression performance. This presented model involves several steps like ROI extraction, GA, Huffman coding and finally compresses the image.Mohammed Ismail also reduced the fractal image encoding time by the use of cuckoo inspired fast search (CIFS) technique[29]. CIFS technique makes use of vectors of range blocks which are arranged by the level of resemblance and coordinate distance respectively. The cuckoo search is altered in a way that the searching process takes place on limited nests (maximum six) and initialization of nest selection searching process is done by levy flights strategy. The overall results revealed that the CIFS method is found to be robust and attained significantly less MSE. Priyanka Jindal et al. [30]introducedpollination based optimization (PBO) algorithm in image compression based on the fitness value of a DCT block of image data. By employing PBO algorithm, the local best and global best values of several DCT blocks are determined. Based on global best values, the compression process will be carried out by the use of RLE and Huffman coding. This method is validated by comparing its results with JPEG interms of MSE and PSNR. Though several image compression techniques have been proposed and found in the literature, we believe that there is more room for enhancement to attain even better compression performance.

Table 2.1 Comparison of reviewed image compression techniques

Reference	Year	Objective	Algorithm used	Performance measure	Compared with	Merits
[9]	2010	To decrease the amount of time required for fractal encoding process	PSO	Encoding time, PSNR	Full search technique	High speed and preserves image quality
[10]	2011	To build the codebook of VQ	HBMO algorithm	-	LBG, PSO–LBG, QPSO–LBG	Reliable with higher quality
[11]	2012	To build the codebook of VQ	FF algorithm	MSE, PSNR and bit rate	LBG, PSO, QPSO, HBMO	faster and attain better quality
[12]	2013	To integrate Super-Spatial Structure Prediction with inter-frame coding for better CR	Head Code Compressi on	CR	JPEG-LS	Low computation complexity
[13]	2014	To compress images based on multi-level image thresholding	Shannon Entropy and DE	PSNR, Weighted PSNR, storage size and standard deviation	PSO and GA	Better CR
[14]	2014	To reduce the fractal encoding time	GA ad DWT	MSE, PSNR, encoding time	GA	100 times faster than full search method
[15]	2015	To perform compression by incorporating IWT with prediction step	Median edge detector	CR	DPCM	better compression performance
[16]	2015	To propose a compression algorithm by exploiting the relationship between fractional numbers and their quotients	GA	CR	-	image quality is saved with higher CR
[17]	2015	To perform lossless compression of significant parts by extracting ROI in DICOM images	GA	PSNR	Hybridization of RLE and Huffman coding	Better CR
[18]	2016	To reduce the fractal image encoding time	CIFS	CPU time, GPU time, MSE, PSNR	PSO with Wavelet Classification, GA with RSM	Robust, lower MSE
[19]	2016	To compress image using the fitness value of a DCT block	PBO	MSE, PSNR	JPEG	Better CR

III. BACKGROUND INFORMATION

A. Discrete Cosine Transform (DCT)

DCT is the fundamental concept of various image processing techniques. DCT is a type of mathematical transformation, intends to transform a signal from one type of representation to another[31]. In general, images are 2D signal which is based on the perception of human visual system (HVS). DCT is defined as the process of converting a signal (spatial information) to numeric data (frequency or spatial information), so that the information of the image exists in a quantitative form which can be manipulated for compression. The general form of 1D-DCT (N data items) is represented in Eq. (1).

$$F(u) \triangleq \begin{cases} \sum_{i=0}^{N-1} 2.f(i).\cos\left[\frac{\pi.u}{2.N}(2i+1)\right], & u \in [0, N-1] \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$\text{where} \; w(i) \triangleq \begin{cases} \frac{1}{\sqrt{2}} & \text{for } i = 0 \\ 1 & \text{otherwise} \end{cases}.$$

For every N point signal f (i) having support [0, N-1], the corresponding inverse DCT (IDCT) can be computed as,

$$f(i) = \begin{cases} \frac{1}{N}\sum_{u=0}^{N-1} w(u).F(u)\cos\frac{\pi.u}{2N}(2i+1), & i \in [0, N-1] \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Assume that the data has finite rectangular support on [0, M-1] x [0, N-1].The basic representation of 2D-DCT of images is given in Eq. (3).

$$F(u,v) \triangleq \sum_{i=0}^{M-1}\sum_{j=0}^{N-1} 4.f(i,j).\cos\left[\frac{\pi.u}{2.M}(2i+1)\right]\cos\left[\frac{\pi.v}{2.N}(2j+1)\right] \quad (3)$$

For $(u,v) \in [0, M-1] \times [0, N-1]$, otherwise $F(u,v) \triangleq 0$. Next, IDCT also exist and given in Eq. (4).

$$f(i,j) \triangleq \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} w(u)w(v).F(u,v).\cos\left[\frac{\pi.u}{2.M}(2i+1)\right]\cos\left[\frac{\pi.v}{2.N}(2j+1)\right] \quad (4)$$

where $w(i) \triangleq \begin{cases} \frac{1}{\sqrt{2}} & \text{for } i = 0 \\ 1 & \text{otherwise} \end{cases}$ the weight function of 2D-DCT is same as 1D-DCT.

B. Firefly Algorithm

FF algorithm was originally introduced by Xin-She Yan in the year of 2007 and 2008 at Cambridge University, inspired by the flashing pattern and behavior of FFs[32]. The flashing pattern of FFs are quite fascinating and there are around 2000 species exists. For every individual FF species, the flashing pattern is different and many FFs show short and rhythmic flashes. These flashlights are caused by the process of bioluminescence and the original reason for those signaling system are still unexplored. There are two basic functions of flashes: attraction of mating partners (communication) and potential preys. Sometimes, it is also used as a protective warning signal from the predators. The rhythmic flash, flashing rate and time period constitute a part of signal system which makes both sexes contact with each other. It is a known fact that light intensity I reduces at some distance r from the light source as it follows the inverse square law, i.e. light intensity I reduces as the distance r increases, $I \propto (1/r^2)$. In addition, when r increases, the absorption of light in the air weakens the brightness of the flashes. These two characteristics limit the visual distance of FFs. During night, FFs can easily communicate over several hundred meters. The pseudo code of FF algorithm is given in Algorithm I. The flashes can be formulated in such a way that it can be integrated with the objective function to be optimized, which makes it possible to formulize new optimization algorithm. FF algorithm follows three ideal rules which are listed below.

- FFs are unisex, they attract with each other independent of their sex
- Attractiveness is related to brightness, the lesser bright FF will move towards a brighter FF
- Brightness of an FF is influenced or calculated by the landscape of the objective function

For maximization problems, flashing brightness is directly proportional to the value of objective function. Basically, there are two issues in FF algorithm: variation in light intensity and formulation of attractiveness. For the sake of simplicity, the attractiveness of an FF is calculated by the brightness, which is integrated with the encoded objective function. In those problems, the brightness I of an FF at specific position x is selected as

$$I(x) \propto f(x) \quad (5)$$

Additionally, the attractiveness β is a relativity parameter which is determined by other FFs. Hence, for two FFs i and j, β is varied with respect to the distance r_{ij}. At the same time, light intensity diminishes as the distance from the source increases and is absorbed by the medium. Hence, it is noted that the attractiveness varies with the degree of absorption. In general, light intensity $I(r)$ modifies using the inverse square law as equated in Eq. (6).

$$I(r) = \frac{I_s}{r^2} \quad (6)$$

Where I_s represents the intensity at the source. When the light intensity I changes with distance r, in presence of predefined light absorption coefficient γ, I can be calculated as

$$I = I_0 e^{-\gamma r} \quad (7)$$

Where I_0 indicates initial light intensity. To eliminate singularity at point $r = 0$ in Eq. (6), the Eq. (6) and Eq. (7) undergo approximation as the Gaussian form given in Eq. (8).

$$I(r) = I_0 e^{-\gamma r^2} \quad (8)$$

When the FF's attractiveness β is based on the light intensity perceived by neighboring FFs, the value of β is determined as given in Eq. (9).

$$\beta = \beta_0 e^{-\gamma r^2} \qquad (9)$$

where β_0 represents the attractiveness at $r = 0$. Since, it can be easier to compute $\frac{1}{1+\gamma r^2}$ than exponential function, Eq. (9) can be rewritten as

$$\beta = \frac{\beta_0}{1+\gamma r^2} \qquad (10)$$

The above two equations define a characteristic distance $\Gamma = 1/\sqrt{\gamma}$, where β is significantly changed from β_0 to $\beta_0 e^{-1}$ in Eq. (9) or $\beta_0/2$ in Eq. (10). For implementation purposes, attractiveness function $\beta(r)$ is monotonically decreasing as given in Eq. (11).

$$\beta(r) = \beta_0 e^{-\gamma r^m}, \quad (m \geq 1) \qquad (11)$$

The characteristic length is computed as

$$\Gamma = \gamma^{\frac{-1}{m}}, \quad m \to \infty \qquad (12)$$

By contrast, for a given Γ in optimization problem, γ is represented as a conventional initial value.

$$\gamma = \frac{1}{\Gamma^m} \qquad (13)$$

The cartesian distance between two FFs i and j $x_i$and$x_j$can be computes as

$$r_{ij} = \| x_i - x_j \| = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2 \frac{1}{\Gamma^m}} \qquad (14)$$

where$x_{i,k}$ is the kth component of the spatial coordinate x_i of the ithFF. The r_{ij} in 2D space is calculated in Eq. (15).

$$r_{ij} = \sqrt{(x_i - x_j)^2 - (y_i - y_j)^2} \qquad (15)$$

The movement of a FFi is attracted to a brighter FF and can be equated as

$$x_i = x_i + \beta_0 e^{-\gamma r_{i,j}^2}(x_j - x_i) + \alpha \in_i \qquad (16)$$

Where 2nd term represents attraction and 3rd term indicates the randomization. Here, α is the randomized parameter and $\in_i$ is a vector of random numbers derived from Gaussian or uniform distribution. As each FF works in an independent way, it can be applicable for parallel implementation. It is superior to GA and PSO due to the nature of FFs aggregation more closely around every optimum.

Algorithm I. Firefly Optimization

1. **Begin Algorithm**
2. **Step 1: Initialize,** *f(x)*←Objective Function, x_i←Initial Population (i=1,2,3…n), *δ*←Light Intensity, *I*← Formulate Light Intensity *(I α f(x))*, γ←Absorption Coefficient.
3. **Step 2:Repeat** through step
4. **Step 2.2:until***T<Max_Generation*
5. **Step 2.1:For***i = 1*to*n* do
6. **Step 2.1.1:For***j = 1* to *n* do
7. **Step 2.1.1.1:If**(I_j>I_i)
8. **Step 2.1.1.1.1:**Vary attractiveness with distance r
9. **Step 2.1.1.1.2:**Shift Firefly *i* approaching towards *j*
10. **Step 2.1.1.1.3:**Assess current solution and update *δ*
11. **Step 2.1.1.2:End If**
12. **Step 2.1.2:EndFor**
13. **Step 2.2:End For**
14. **Step 3:**Rank Fireflies and Find the Best cost
15. **Step 4:**Post-Process the solution
16. **End Algorithm**

IV. THE PROPOSED LOSSLESS COMPRESSION MODEL

A. Overview

The overall operation of the proposed lossless image compression model is shown in Fig 4.1 Initially, the reference image needs to be compressed is partitioned into sub-images (8*8 blocks). The blocks of image undergo the quantization process where the proposed FF-TLBO algorithm constructs the quantization table based on the maximization of fitness function. FF-TLBO algorithm is employed to compute the transform coefficients, where the coefficients closer to top left corner holds the most significant information. The resultant coefficients are quantized by the use of quantization table. Next, encoding process takes place where the AC coefficients are encoded by RLE[33] and DC coefficients are encoded by Huffman coding [34]technique to generate the compressed image. The proposed method follows symmetric compression in which the decompression process is exactly same as compression process, but in the opposite direction.

B. FF-TLBO algorithm

As explained above, the input image is partitioned into 8*8 blocks of sub-images which is given as input to DCT. The FF-TLBO algorithm computes the best fitness value for every DCT block. This algorithm determines the best fit value for every block is called as local best whereas the best fitness value for the entire image is considered as global best. The fitness function is defined in Eq. (17) assigns a fitness value for transforming the array of coefficients.

$$f(x) = (x_1, x_2, \ldots x_d)^T \qquad (17)$$

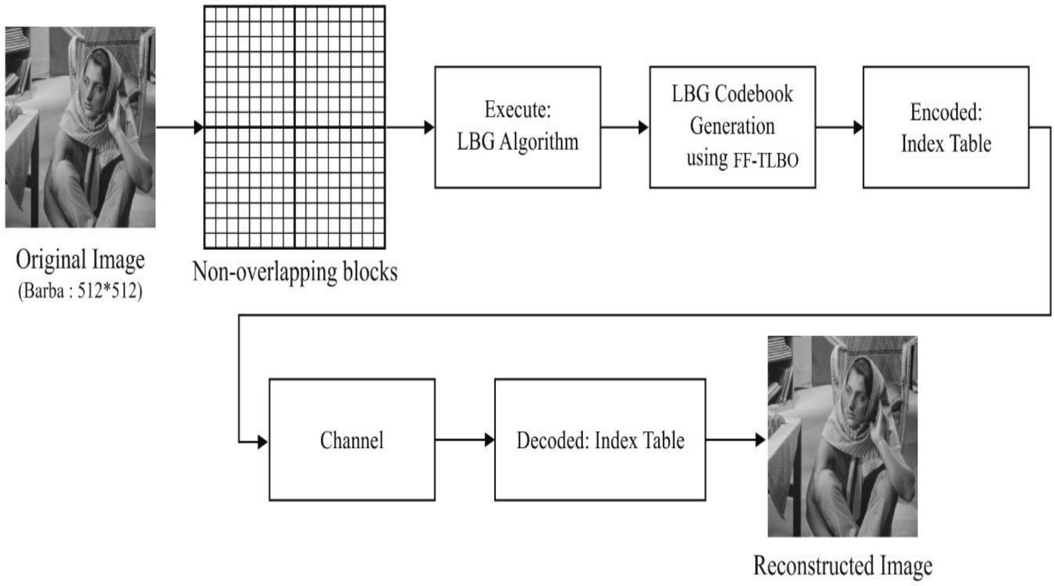


Fig 4.1 Overall process of FF-TLBO method

Teacher phase

- Initialization: In this step the initial population x_i, light intensityI_i at $x_i$and γ are initialized.
- Choose the current best solution: This step chooses the best solution from all the solutions and is defined as x_i^{max},

$$x_i^{max} = \arg\max_i f(x_i) \quad (18)$$

- Attractiveness: The movement of FF $x_i$is attracted to another FF x_j. Every solution x_j calculates the fitness values with respect to the brightness of the FFs as given in Eq. (17).
- Termination condition: When the number of iterations exceeded, then the FF algorithm stops its execution and give the best solution.

The FF algorithm performs well when the brighter FF is available in the search space. In some cases, when none of the brighter FF appears in the search space, the FFs start moving randomly. This is the major drawback of the FF algorithm. To resolve this algorithm, we introduce the FF-TLBO algorithm which integrates the TLBO algorithm with FF algorithm to explore the search space efficiently.

The TLBO algorithm is stimulated from the knowledge transfer between the teachers and students in the learning and it depends on the influence of the teacher on the outcome of the learners in the class [35]. The two main phases in TLBO algorithm is 'Teacher Phase' (learns from teacher) and 'Learner Phase' (learns via their interaction).

The nature of the good teacher is they should try to improve the learner's knowledge level to a maximum level or atleast to his/her level. In practical, it is difficult and the teacher can attain the mean of the class to a certain level based on different dimensions. For instance, $M_i$indicates the mean of the class and T_i is the teacher at any iteration i. The teacher T_i will try to move the mean M_i closer to its own level, therefore the new mean become T_i named as M_{new}.

The solution will be updated using the differences between the present and new mean M_{new} as given in Eq. (19).

$$\text{Difference\_Mean}_i = r_i(M_{new} - T_F M_i) \quad (19)$$

Where T_F denotes a teaching factor which calculates the mean value to be modified and r_i is a random number lies between [0, 1]. The T_F value will be either 1 or 2, which is arbitrarily decided as T_F= round [1 + rand (0, 1) {2 − 1}]. This modification will alter the existing solution using Eq. (20) as represented below.

$$X_{new,\ i} = X_{old,\ i} + \text{Difference\_Mean}_i \quad (20)$$

Learner phase

The learners can improvise the knowledge by the use of two ways: the former one is getting input from the teacher and latter one is their interaction between them. A learner can improve the knowledge by random interaction with other learners. In general, the knowledge of a learner improves once the learner will interact with the more knowledgeable learner. In those situations, the learner modification can be equated as

$For\ i = 1 : P_n$
  $Randomlyselecttwolearners$ $X_i$andX_j, where i≠j
  $If\ f(X_i) < f(X_j)$
    $X_{n,i}= X_{o,i}+ r_i(X_i - X_j)$
  Else
    $X_{n,i}= X_{o,i}+ r_i(X_j - X_i)$
  End If
End For

Accept $X_n$when a better function value is obtained.

The TLBO algorithm aims to maximize the fitness function by the construction of the quantization table at desiredcompression efficiency. At the initial state, the quantization table generated by the FF algorithm is used as the initial point. Every quantization table generated by the FF algorithm denotes a student in the TLBO algorithm. At the end, the optimized quantization table for the applied images has been attained by maximizing the fitness function by the utilization of the phases involved in the TLBO algorithm.

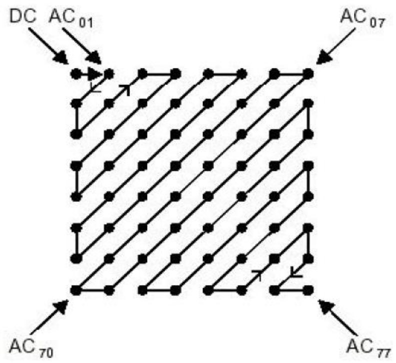


Fig 4.2 Zigzag Scanner

After the execution of the quantization process, the zigzag scanner scans all the quantized coefficients as shown in Fig. 4.2 In the zigzag sequence, the coefficients with lower frequencies (DC coefficients) are encoded first and the higher frequencies (AC coefficients) are encoded. The AC coefficients are encoded using RLE and the DC coefficients are encoded by Huffman coding. Finally, the compressed image with reduced file size from reference image is generated. When the compressed image is received, decoding process will take place using Huffman decoding and RLE decoding techniques. Next, the decoded image undergoes dequantization process and then IDCT operation is performed. Once the IDCT operation is completed, all the individual sub-images (8*8 blocks) are merged and finally, the reconstructed image is generated.

C. FF-Tumbling algorithm

As explained above, the input image is partitioned into 8*8 blocks of sub-images which is given as input to DCT. The FF-Tumbling algorithm computes the best fitness value for every DCT block. In the modified bat algorithm, the selection of bat movement is decided by the value offitness function. If bat moves towards the optimum value of fitness function then type ofbat movement is swimming. Otherwise bat follows the chemotactic movement of bacterium. The chemotactic movement of bacterium is represented by the following Eq. (21)

$$x_i^t = x_i^{t-1} + v_i^t \frac{\Delta_i}{\sqrt{\Delta_i^T \times \Delta_i}} \qquad (21)$$

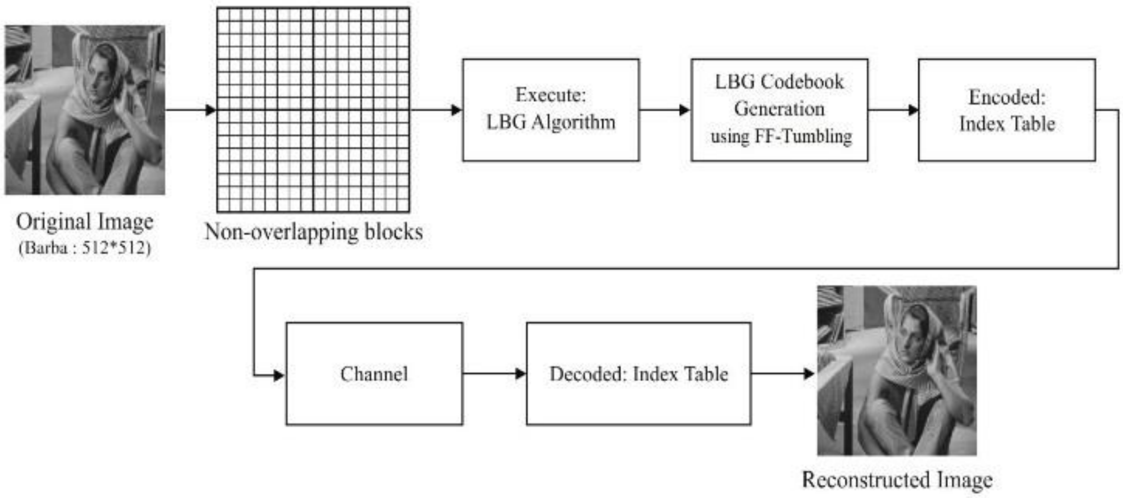


Fig 4.3 Overall process of FF-Tumbling method

V. PERFORMANCE EVALUATION

To ensure the efficiency of the proposed lossless compression algorithm, it is tested against a set of 40 benchmark images from LIVE database [36]. The obtained results are compared with existing PBO method, which is one of the popular bio-inspired algorithm employed in the area of lossless image compression.

A. Metrics

MSE, PSNR, SNR are used as performance measures to validate the results of the proposed and existing methods[37], [38]. MSE is commonly employed to calculate the difference between the reference and reconstructed images. It can be equated as

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(P_i - P_j)^2 (21)$$

Where n represent the total number of pixels in the image, P_i and P_j is the pixel values of the reference and reconstructed images. The value of MSE should be lower to produce better compression performance. Root Mean Square Error (RMSE) is the square root of MSE, which is used to calculate PSNR.s

PSNR is the ratio between maximum possible power of signal and power of error signal which influences the fidelity of its representation. It can be computed as

$$\mathrm{PSNR} = 20\log_{10}\frac{\max_{i,j}|P_{i,j}|}{\mathrm{RMSE}} \quad (22)$$

Where $\max_{i,j}|P_{i,j}|$ represents the maximum pixel value in the image. For better similarity among two images, the typical value lies in the range of 20 and 40. When the reference and reconstructed images are exactly identical, the value of MSE will be zero and PSNR will be infinity.

SNR is defined as the ratio of the power of a signal to the power of background noise.

$$\mathrm{SNR} = \frac{P_{signal}}{P_{noise}} \quad (23)$$

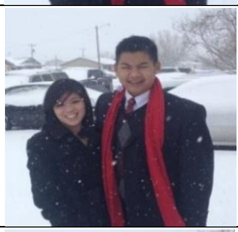
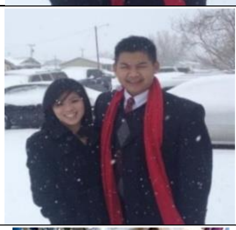
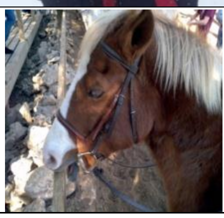
B. Results and discussion

Table 5.1and Fig. 5.1 provides the comparative results of proposed and existing FF algorithms interms of MSE, PSNR, SNR respectively. The same set of 10 images is applied to both the existing and proposed methods. The tabulated results revealed that the average MSE of FF method is worse when compared to proposed method. The existing method attains an average MSE of 0.910975 whereas the proposed method achieves an efficient MSE of 0.64205. The obtained results show that the proposed method produces lesser MSE which indicates the better compression performance. Likewise, the average PSNR of FF method is 48.565, but the proposed method attains a PSNR of 53.1322, which is much higher than FF method. In the same way, average SNR values of the proposed and existing FF methods are 0.86925 and 0.67025 respectively. The higher value of SNR by proposed method notifies that the better reconstructed image quality of the proposed method when compared to FF method.

Table 5.1 Comparison of FF-TLBO and FF-Tumbling algorithm with DCT algorithm in terms of MSE, PSNR, SNR

LIVE DATASET	MSE			SNR			PSNR		
	FF-Tumbling	FF-TLBO	DCT	FF-Tumbling	FF-TLBO	DCT	FF-Tumbling	FF-TLBO	DCT
Image 11	13.18	53.754	60.545	35.406	29.302	33.278	46.474	40.369	39.852
Image 12	1.9002	9.866	20.367	43.915	36.762	44.389	54.885	47.732	44.584
Image 13	4.7308	57.135	69.394	41.228	30.409	39.489	50.924	40.104	39.26
Image 14	17.948	53.414	65.389	31.798	27.061	35.478	45.133	40.397	39.518
Image 15	8.568	44.631	62.384	37.232	30.065	40.378	48.344	41.177	39.722
Image 16	1.641	37.52	70.378	46.161	32.57	40.378	55.522	41.931	39.199
Image 17	8.632	33.509	45.289	40.384	34.494	44.389	48.312	42.422	41.113
Image 18	1.133	20.822	49.389	48.119	35.479	43.347	57.131	44.488	40.737
Image 19	4.7004	19.536	34.589	39.638	33.451	42.189	50.952	44.765	42.284
Image 20	15.337	73.801	90.347	34.428	27.605	31.289	45.816	38.993	38.114

LIVE DATASET			MSE	PSNR	SNR
(a)		**DCT algorithm**	69.394	39.26	39.489
		**FF-TLBO algorithm**	57.135	40.104	30.409

		**FF-Tumbling algorithm**	4.7308	50.924	41.228
(b)		**DCT algorithm**	65.389	39.518	35.478
		**FF-TLBO algorithm**	53.414	40.397	27.061
		**FF-Tumbling algorithm**	17.948	45.133	31.798
(c)		**DCT algorithm**	49.389	40.737	43.347
		**FF-TLBO algorithm**	20.822	44.488	35.479
		**FF-Tumbling algorithm**	1.133	57.131	48.119
(d)		**DCT algorithm**	90.347	38.114	31.289

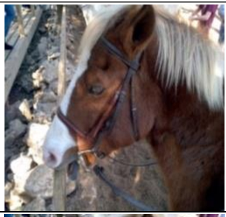
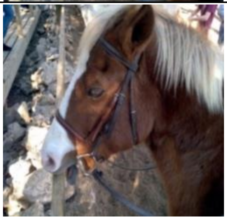
		**FF-TLBO algorithm**	73.801	38.993	27.605
		**FF-Tumbling algorithm**	15.337	45.816	34.428

Fig 5.1 Evaluation of LIVE Image Dataset (a) Image 13, (b) Image 14, (c) Image 18and (d) Image 20

To further facilitate the highlights of the proposed method, some interesting results of the applied benchmark images are shown in Fig. 5.1 This figure shows the obtained values of four images from LIVE database include Image 13, Image 14, Image18 and Image 20 respectively. From Fig. 5.1a, the results of Image 13 show that the proposed method attains better performance than DCT method. It can be shown from the values MSE= 69.394, PSNR = 39.26, SNR= 39.489 respectively. However, the FF method fails to achieve a closest performance, achieved an MSE= 57.135, PSNR = 40.104, SNR= 30.409 and MSE=4.7308, PSNR=50.924, SNR= 41.228 respectively. Finally, in Fig. 5.1d, the results of existing and FF-TLBO and FF-Tumbling algorithms are shown. Likewise, in Fig. 5.1b, the results of Image 14 are shown where the proposed method is superior to DCT method in all the performance measures involved. It is clearly shown from the values MSE=65.389, PSNR = 39.518, SNR= 35.478 respectively. However, the FF method fails to attain maximum performance, achieved a MSE=53.414, PSNR =40.397, SNR=27.061 and MSE=17.948, PSNR=45.133, SNR=31.798 respectively. the results of existing and FF-TLBO and FF-Tumbling algorithms are shown.Similarly, for Image 18 in Fig 5.1c, the attained values revealed that the existing method outperforms the FF-TLBO and FF-Tumbling method. The existing FF method reported the values of MSE=49.389, PSNR = 40.737, SNR=43.347respectively. But, the proposed method produced enhanced results with the MSE=20.822, PSNR =44.488, SNR=35.479 and MSE=1.133, PSNR=57.131, SNR=48.119 respectively. Finally, in Fig. 5.1d, the results of existing and FF-TLBO and FF-Tumbling algorithms are shown. The obtained values indicated that the effectiveness of proposed method over FF method. The proposed method attained an MSE= 90.347, PSNR =38.114, SNR=31.289 respectively. At the same time, FF method fails to manage maximum compression performance and better reconstructed image quality with the MSE=73.801, PSNR =38.993, SNR=27.605 and MSE=15.337, PSNR=45.816, SNR=34.428 respectively.

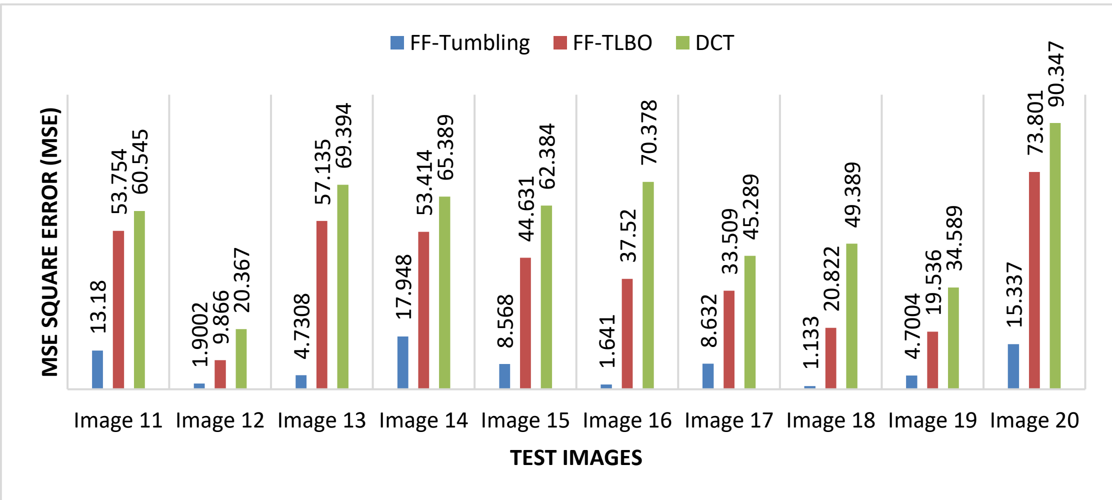


Fig 5.2 Comparative analysis of proposed method with FF method interms of MSE

It shows that the proposed FF-TLBO and FF-Tumbling algorithm is reliable and robust for all the applied images.These values depict that maximum reconstructed image quality and better compression performance is produced by the proposed method. The obtained values imply that the proposed method manages to retain the image quality as well as the better compression performance in a reasonable amount of time.

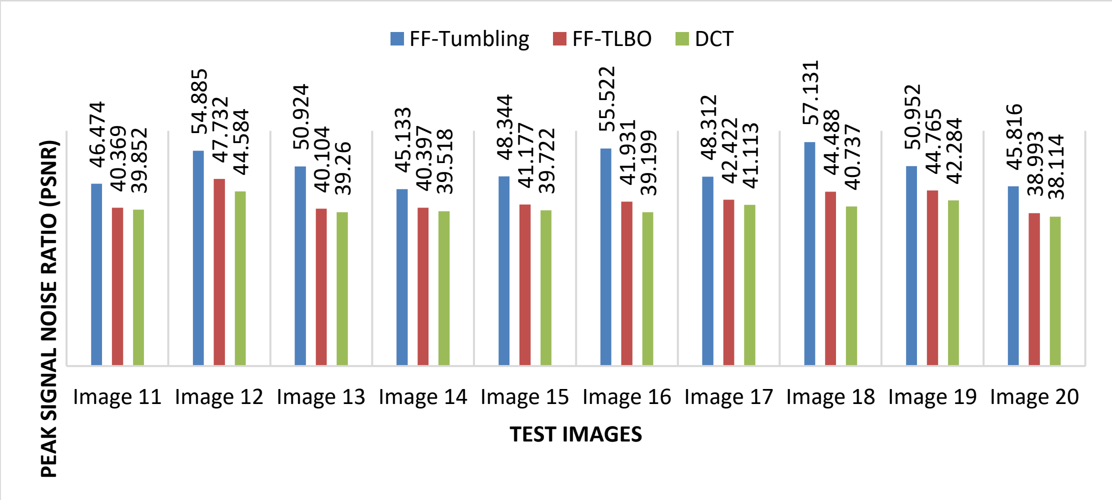


Fig. 5.3Comparative analysis of proposed method with FF method interms of PSNR

Fig. 5.2-5.4 shows the comparison results of proposed and FF method interms of MSE, PSNR, SNR respectively. Fig. 5.2 illustrates the comparison results of proposed and existing FF methods interms of MSE. From this Fig., it is noted that the MSE of the FF-TLBO and FF-Tumbling algorithm is significantly better than FF method. Fig. 5.3 demonstrates the performance of the proposed and existing FF methods interms of PSNR. Fig. shows that the maximum value of PSNR is achieved by proposed method, which reveals the maximum performance of the proposed method. The existing method achieves a minimum PSNR of 48.15 and maximum PSNR of 49.71 whereas the proposed method reported a minimum PSNR of 50.12 and maximum PSNR of 57.58 respectively. The performance of the proposed and compared method interms of SNR is depicted in Fig.5.4. From the Fig, it is interesting that the proposed method reaches maximum resemblance with the SNR for Image 13, Image 14, Image 18 and Image 20 respectively.The existing method achieves a minim PSNR of 48.15 and maximum PSNR of 49.71 whereas the proposed method reported a minimum PSNR of 50.12 andmaximumPSNR of 57.58 respectively.

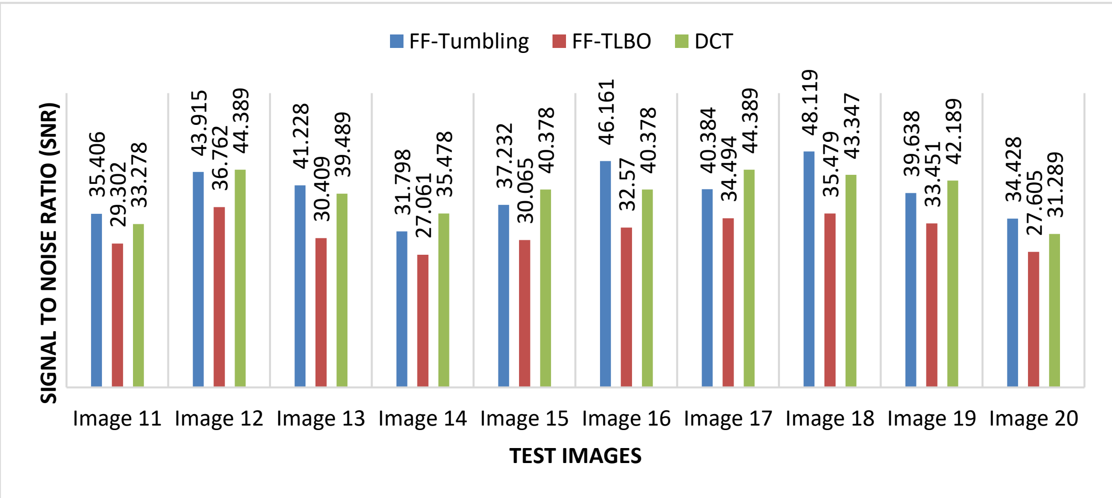


Fig. 5.4Comparative analysis of proposed method with FF method interms of SNR

For better understanding, an additional experiment is carried out to analyze the visual similarities by comparing the results obtained by FF algorithm and FF-TLBO algorithm. Figure 5.7 and 5.8 shows the original and compressed images with quality coefficient with standard quantization matrix attained by FF algorithm is given in Table 5.2 It is found that the CR is high with degraded image quality. The average pixel intensity distance between the original and compressed image is 5.9.

Fig 5.7 Original image "Lena", 512*512

Table. 5.2Quantization matrix by FF

120	56	60	129	190	225	255	255
60	66	76	156	154	255	255	255
78	78	129	198	223	255	255	255
70	86	178	143	255	255	255	255
90	123	185	255	255	255	255	255
126	189	255	255	255	255	255	255
250	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255

The proposed FF-TLBO algorithm obtained the optimal quantization matrix for the same level of compression. This matrix is shown in Table 5.3 and the decompressed image with that quantization matrix is shown in Fig. 5.9 It is found that same level of compression is attained with better reconstructed image quality and the average pixel intensity distance was reduced to 5.1.

Fig. 5.8FF Algorithm

Fig. 5.9FF-TLBO algorithm

Fig 5.10FF-Tumbling algorithm

Table 5.3 Quantization matrix by FF-TLBO algorithm

16	26	68	124	96	255	255	255
16	22	124	143	178	255	255	255
16	34	187	165	255	255	255	255
234	228	18	122	255	255	255	255
16	42	65	255	255	255	255	255
245	16	255	255	255	255	255	255
255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255

From these figures, it is clear that the FF-TLBO algorithm obtained the quantization table with same number of bits for nonzero frequency coefficients.

Table 5.4 Quantization Matrix By FF-Tumbling Algorithm

122	156	60	129	190	225	255	255
160	66	76	156	255	255	255	255
78	178	129	198	255	255	255	255
170	86	178	143	255	255	255	255
90	123	185	255	255	255	255	255
126	189	255	255	255	255	255	255
250	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255

This illustration verifies the significance of metaheuristic algorithms on the selection of quantization tables.

VI. CONCLUSIONS

This chapter presented a detailed explanation of how the FF-TLBO algorithm finds useful to construct the quantization table and enhances the performance of the compression techniques. All of the experimentation results reported that the proposed method achieved better compression performance and also increased the reconstructed image quality with respect to compared FF algorithm. Thisensures that FF-TLBO algorithm is potentially powerful in achieving near lossless compression performance which will be concentrated more in future studies. Additionally, further studies on the application of different metaheuristic algorithm may create an interesting field for upcoming research in image compression.

REFERENCES

1. Bookstein A, A.Storer J. Data Compression. Inf Process Manag 1992;28.
2. Salomon D. Data Compression The Complete Reference. 4th ed. Springer; 2007.
3. Rehman M, Sharif M, Raza M. Image compression: A survey. Res J Appl Sci Eng Technol 2014;7:656–72.
4. Drost SW, Bourbakis N. A Hybrid system for real-time lossless image compression. Microprocess Microsyst 2001;25:19–31. doi:10.1016/S0141-9331(00)00102-2.
5. Holtz K. The Evolution of Lossless Data Compression Techniques 1999:140–5.
6. Tarek S, Musaddiqa M, Elhadi S. Data compression techniques in Wireless Sensor Networks. Futur Gener Comput Syst 2016;64:151–62. doi:10.1016/j.future.2016.01.015.
7. Narasimha M, Peterson A. On the Computation of the Discrete Cosine Transform. IEEE Trans Commun 1978;26:934–936.
8. Bonabeau E, Dorigo M, Theraulaz G. Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press; 1999.
9. Deb K. Optimisation for Engineering Design. Prentice-Hall, New Delhi; 1995.
10. Kennedy J, Eberhart R, Shi Y. Swarm intelligence. London: Academic Press; 2001.
11. Shilane D, Martikainen J, Dudoit S, Ovaska SJ. A general framework for statistical performance comparison of evolutionary computation algorithms. Inf Sci (Ny) 2008;178:) 2870–2879.
12. Kennedy J, Eberhart RC. Particle swarm optimization. Proc. IEEE Int. Conf. Neural Networks, Piscataway, NJ, 1995, p. 1942–1948.
13. Wang Y, Feng XY, Huang YX, Pu DB, Zhou WG, Liang YC. A novel quantum swarm evolutionary algorithm and its applications. Neurocomputing 2007;70:633–640.
14. Goldberg DE. Genetic Algorithms in Search, Optimization, and Machine Learning. ADDISON-WESLEY PUBLISHING COMPANY, INC.; 1989.
15. Storn R, Price K. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. J Glob Optim 1997;11:341–59. doi:10.1023/A:1008202821328.
16. Abbasss HA. Marriage in honey-bee optimization (HBO): A haplometrosis Computation, polygynous swarming approach. Congr. Evol., 2001, p. 207–14.
17. Yang X-S. Flower Pollination Algorithm for Global Optimization. Int. Conf. Unconv. Comput. Nat. Comput. UCNC 2012 Unconv. Comput. Nat. Comput., 2012, p. 240–9.
18. Yang XS, Deb S. Engineering optimisation by cuckoo search. Int J Math Model Numer Optim 2010;1:330–43.
19. Muruganandham A, Wahida Banu RSD. Adaptive Fractal Image Compression using PSO. Procedia Comput Sci 2010;2:338–44. doi:10.1016/j.procs.2010.11.044.
20. Horng MH, Jiang TW. Image vector quantization algorithm via honey bee mating optimization. Expert Syst Appl 2011;38:1382–92. doi:10.1016/j.eswa.2010.07.037.
21. Linde Y, Buzo A, Gray RM. An algorithm for vector quantizer design. IEEE Trans Commun 1980;28:84–95.
22. Horng MH. Vector quantization using the firefly algorithm for image compression. Expert Syst Appl 2012;39:1078–91. doi:10.1016/j.eswa.2011.07.108.
23. Ukrit, Mferni. Suresh G. Effective lossless compression for medical image sequences using composite algorithm. Int. Conf. Circuits, Power Comput. Technol., 2013, p. 1122–6.
24. Paul S, Bandyopadhyay B. A Novel Approach for Image Compression Based on Multi-level Image Thresholding using Shannon Entropy and Differential Evolution. Proceeding 2014 IEEE Students' Technol. Symp. A, 2014, p. 56–61.
25. Wu MS. Genetic algorithm based on discrete wavelet transformation for fractal image compression. J Vis Commun Image Represent 2014;25:1835–41. doi:10.1016/j.jvcir.2014.09.001.
26. Fouad MM. A Lossless Image Compression Using Integer Wavelet Transform With a Simplified Median-edge Detector Algorithm. Int J Eng Technol 2015;15:68–73.
27. Omari M, Yaichi S. Image Compression Based on Genetic Algorithm Optimization. 015 2nd World Symp. Web Appl. Netw., Sousse: 2015, p. 1–5.
28. Kaur H, Kaur R, Kumar N. Lossless compression of DICOM images using genetic algorithm. 2015 1st Int. Conf. Next Gener. Comput. Technol., 2015, p. 985–9. doi:10.1109/NGCT.2015.7375268.
29. Ismail BM, Eswara Reddy B, Bhaskara Reddy T. Cuckoo inspired fast search algorithm for fractal image encoding. J King Saud Univ - Comput Inf Sci 2016. doi:10.1016/j.jksuci.2016.11.003.
30. Jindal P, Raj bhupinder Kaur. Lossless Image Compression for storage reduction using Pollination Based Optimization. Commun. Electron. Syst. (ICCES), Int. Conf., 2016, p. 1–6.
31. Watson AB (Nasa ARC. Image Compression Using the Discrete Cosine Transform. Math J 1994;4:81–8. doi:10.1006/jvci.1997.0323.
32. Yang X-S. Firefly Algorithms for Multimodal Optimization. Proc. 5th Int. Conf. Stoch. Algorithms Found. Appl., 2009, p. 169–78. doi:10.1007/978-3-642-04944-6_14.
33. Capon J. A probabilistic model for run-length coding of pictures. IRE Trans Inf Theory 1959;100:157–63.
34. Huffman DA. A Method for the Construction of Minimum-Redundancu Codes. A Method Constr Minimum-Redundancu Codes 1952:1098–102.
35. Rao RV, Savsani VJ, Vakharia DP. Teaching–learning-based optimization: A Comput.-, novel method for constrained mechanical design optimization problems. Aided Des 2011;43:303–315.
36. LIVE Image Quality Assessment Database n.d. http://live.ece.utexas.edu/research/quality/subjective.htm.
37. Sayood K. Introduction to Data Compression. 2006. doi:10.1159/000207355.
38. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. EEE Trans Image Process 2004;13:600–12. doi:10.1109/TIP.2003.819861.

AUTHORS PROFILE

Ms. Preethi is a research scholar and is pursuing his Ph.D. in the Department of Computer Science and Engineering, Pondicherry Engineering College (PEC), Pondicherry, India. She has completed his M.Sc degree in Computer Science from Pondicherry University, India. Her area of interest includes image processing, image compression.

Dr. D. Loganathan is a Professor in the Department of Computer Science and Engineering, Pondicherry Engineering College (PEC), Pondicherry, India. He received M.E from BITS Pilani and PhD from Anna University. He has experience over twenty nine years. His research interest include Image processing and Information Security. He holds more than thirty journal publications and proceedings.