

# An Analysis of Codebook Optimization for Image Compression: Modified Genetic Algorithm and Particle Swarm Optimization Algorithm



Pratibha Chavan, B. Sheela Rani, M. Murugan, Pramod Chavan, and M. Kulkarni

**Abstract** Billions of images are uploaded daily, and it requires a large storage space. Utilization of better storage capacity and to improve uploading/downloading time, researchers have designed an image compression model. Many researchers have implemented various approaches to improve the image compression ratio of an image. This paper presents an analysis of various optimization algorithms based on vector quantization (VQ). The first algorithm is a modified genetic algorithm. It is based on Darwin's principle which is natural characteristics. Those who are fit can survive and use it to optimize the codebook. A second algorithm for optimization of the codebook is particle swarm optimization (PSO). PSO algorithm is superior to finding the codeword vectors of codebook from the training image samples for image compression. In the PSO algorithm, the selection approach plays an important role to select the particle based on the fitness of the population. Training images from the standard image database are used for the design of the codebook. The input image set is  $4 \times 4$  or  $8 \times 8$  blocks and is represented as vectors. They are referred to as codewords in the codebook, and it is a component of a code. The codebook size is measured by codewords. The block size is decided by the length of the codeword. These codewords generate the codebook by entering the vector value. Compression is done with the help of sending indices to the decoder. Likewise, analysis of quality measures is presented to the modified GA and PSO algorithms based on mean square

---

P. Chavan · B. Sheela Rani (✉)

Sathyabama Institute of Science and Technology (Deemed to be University), Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai, Tamil Nadu 600 119, India  
e-mail: [sheelarani@sathyabama.ac.in](mailto:sheelarani@sathyabama.ac.in)

P. Chavan · M. Kulkarni

Department of E&TC Engineering, Trinity College of Engineering and Research, Pisoli, Pune, Maharashtra 411048, India

M. Murugan

SRM Valliammai Engineering College, Kattankulathur, Chengalpattu District, Tamil Nadu 603203, India

P. Chavan

Department of E&TC Engineering, K J College of Engineering and Management Research, Pisoli, Pune, Maharashtra 411048, India

error, peak signal-to-noise ratio, structural similarity index, and average difference. In this work, we have calculated bits per pixel (BPP), the compression ratio (CR), and the % compression ratio. The experimental results are validated.

**Keywords** Codebook · Modified genetic algorithm · Particle swarm optimization · Vector quantization

## 1 Introduction

Image compression [1–3] is essential because the education system is using the online platform for teaching, and demand increases for digital notes to be shared with students all over the world. As doctors are doing research on placid trail for patients of COVID-19, the same numerous data storage of patient increases. It improves the transmission speed of information to be transmitted or received in hybrid communication channel and furthermore modern applications [4, 5]. Image compression is classified into a lossy image compression [6] and lossless image compression. These techniques are challenging that to achieve a good compression ratio [7, 8]. Due to small information loss, quality of restoration is poor in lossy method [9, 10]. Lossless image compression technique is having negligible loss [11, 12]. Therefore, the data of the encoder and decoder are closely similar, and the restoration quality is good. Depending on application, researchers can select lossy or lossless image compression [13–15].

Research flow is as follows:

1. This research work is based on vector quantization.
2. The input image is divided into 4 by 4 blocks, and that input vectors are represented by 16-bit codeword.
3. Modified GA and PSO are codebook optimization algorithms, and they are used to improve compression ratio.
4. To achieve good compression ratio, error must be minimum.
5. In the end, the analysis of the BPP and CR is compared.

The work is prepared in following sections. Section 2 scrutinizes the literature survey. Section 3 presents vector quantization method, and Sect. 4 exposes modified GA and PSO. Section 5 presents the experiments, and research work is concluded in Sect. 6.

## 2 Literature Review

### 2.1 Contribution by Researchers

In 2020, Shakya, Subarna, and Lalitpur, Nepal Pulchowk introduced a novel bi-velocity particle swarm optimization scheme for multicast routing problem. Dhaya R. and R. Kanthavel introduced comprehensively meld code clone identifier for replicated source code identification in diverse web browsers. Daniela Sánchez, Patricia Melin, and Oscar Castillo have investigated the firefly algorithm (FF) [6]. Blinking characteristics of fireflies are used in this model. One important feature is that regardless of their sex, fireflies are attracted to one another. Brighter the control attractiveness, brighter the firefly attracts the less bright firefly, else move by chance. Proficient codebook is designed using firefly algorithm [16]; the challenge is less availability of brighter fireflies in exploring.

In 2019, Panda and Das [17] have introduced a gray wolf optimizer (GWO). The gray wolves have hunting behavior and leadership hierarchy. This algorithm is character-motivated algorithm and used for optimization [18]. Neighboring region is supported for the primary elucidation of every progression to produce the best outcome. If termination criteria are not fulfilled, then step will be continued [19].

In 2018, Cui, Z., Li, F. and Zhang, W. have introduced VQ-based bat algorithm (BA). BA is used for image compression. BA technique improves the result. The codebook is designed efficiently by varying all possible parameters.

In 2016, Mirjalili and Lewis [20] have presented whale optimization algorithm. WOA follows the group performance. This algorithm is encouraged through the bubble-net track approach. This is a popular optimization algorithm when it is evaluated with conventional methods [21]. Hence, all these features are considered in this research work along with the challenges faced by these optimization algorithms. This paper is enlightening the performance of image compression models by GA and PSO algorithms.

### 2.2 Problem Definition

New era of internet is demanding for compressed image for the same need to design an optimal codebook with better compression rate and minimum error at reconstructing image [1]. Researchers have proposed optimization techniques for codebook generation because the performance of VQ depends on codebook. Image compression model using modified genetic algorithm was presented in [2] that provides minimized error at reconstructing the image. Selection, crossover, and mutation operators are used for codebook design. Outcomes were satisfactory appropriate to its unique chromosome feature use in the codebook design. Limitation of modified GA is that it undergoes more execution time when population size is large. The researcher has proposed another optimization technique called particle swarm optimization (PSO)

[3]. It offers a superior efficiency and better compression ratio. It initializes the swarm size and calculates the fitness value. Those who are best fit swarm, that position is updated by replacing previous swarm. The steps are repeated until it fulfills the criteria. If the position of the swarm is not varied or the termination criteria are satisfied, then stop. The best position of the particles is updated. For the maximum particle velocity, instability occurs in the particles. It is necessary for analyzing these optimization models to get better image compression results.

### 2.3 Motivation

Optimization algorithms, specialization, and problem face for image compression are described in Table 1. At first, the modified genetic algorithm has been implemented [2] which reduces artifact impacts and simplifies the design of the codebook. Modified GA is based on natural characteristics, so there would be no chances of duplications and gives better results. Modified GA has challenge to produce optimal outcomes. Likewise, second algorithm has explored a novel approach by vector quantization for image compression using particle swarm optimization (PSO) algorithm, which is a superior model. PSO is high quality and better than other optimization model. Therefore PSO model could outperform the other image compression standards. Analysis of these optimization models is to be presented in this paper.

**Table 1** Characteristics and problems face in various image compression models

Algorithm	Characteristics	Problem face
Modified genetic algorithm	Reduced artifact impacts	The challenge to produce optimal outcomes
PSO algorithm	Superior compression ratio	For the high particle velocity challenge is stability
Firefly algorithm	Generate an efficient codebook	Shortage of brighter fireflies
Bat algorithm	Provide high PSNR as compared to the LBG	No significance difference in PSO

### 3 Novel Idea of the Proposed Technique Andmodels

#### 3.1 A. *Vector Quantization and Codebook Optimization Method*

VQ process is superior to scalar quantization techniques. VQ can reduce the distortion with the number of reconstruction levels in keeping constant. Scalar quantization quantizes every value individually.

Figure 1 shows scalar quantization. Vector quantization is complex method and quantizes several values jointly. VQ is usual a lossy image compression technique and presents good compression ratio as shown in Fig. 2.

Vector quantization is a conventional methodology for image compression. As a main contribution, codebook is optimized in such a way that it minimizes error in the reconstructed image. A few images from the database are used for the codebook generation. Set of blocks is presented as vectors. The encoder sends their indices to achieve a compression through a channel. Output file reconstructs the image by placing correct vectors on the image with the help of decoder. The codebook is optimized using optimization algorithms, i.e., GA and PSO, and we could get

Fig. 1 Scalar quantization

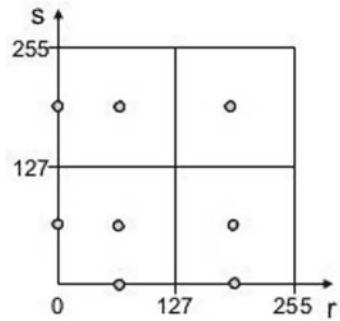
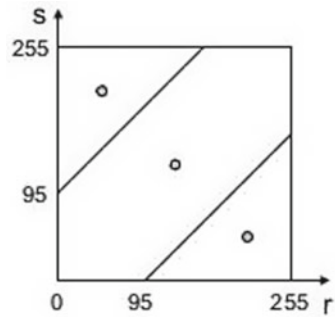


Fig. 2 Vector quantization



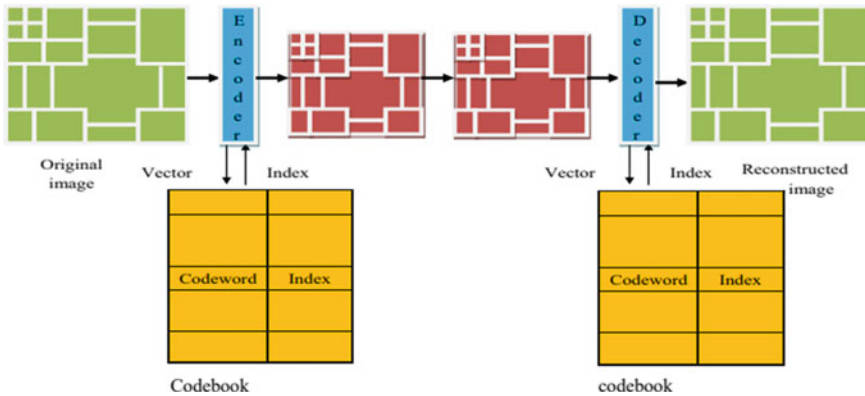


Fig. 3 Architecture of the vector quantization

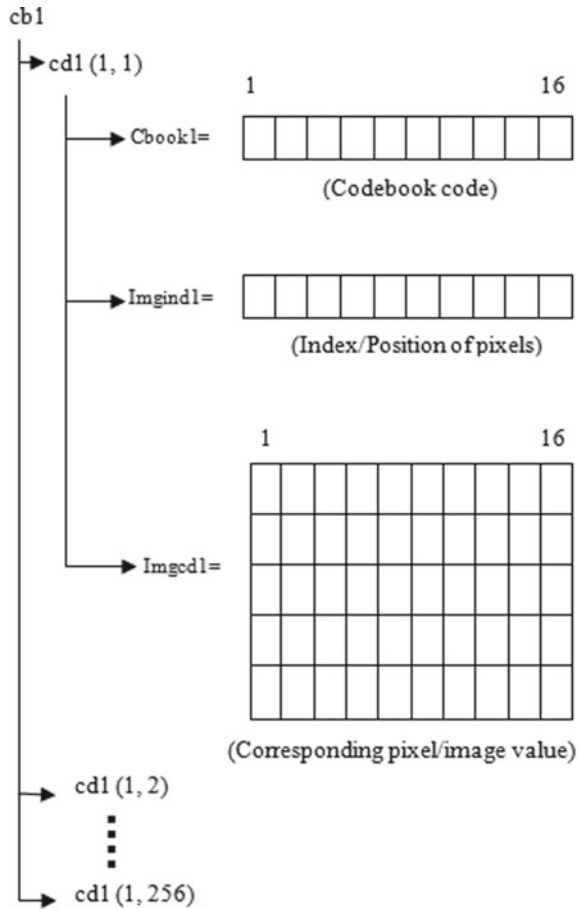
better results. In this paper, we are presenting the analysis using these optimization algorithms. The architecture of the vector quantization is demonstrated in Fig. 3.

### 3.2 Codebook Structure and Fitness Function

The codewords  $cd_1$  are stored in the codebook  $cb_1$ . The size of the codebook can be changed depending upon the size of the input image. Codebook  $cb_1$  consists of 256 codewords, i.e.,  $cd_1(1, 1), cd_1(1, 2) \dots cd_1(1, 256)$ . These codewords are divided further in three parts, viz. the code of codebook ( $cbook_1$ ), indices/positions of codebook ( $imgind_1$ ), and their corresponding image value/pixel ( $imgcd_1$ ). Figure 4 shows the structure of the codebook. A set of chromosomes is stored in  $cbook_1$  of  $cd_1(1, 1)$  by the process of chromosome selection, and then crossover and mutation are performed on these stored chromosomes. The selection of chromosome, crossover, and mutation take place on a random basis. Then again a new set of chromosomes is selected for  $cd_1(1, 2)$ , and the entire process is repeated again up to  $cd_1(1, 256)$ . 0–255 blocks can be designed in the codebook, and up to 256 codewords are presented for distribution [1]. To design the codebook in such a way that it minimizes the image compression effect.

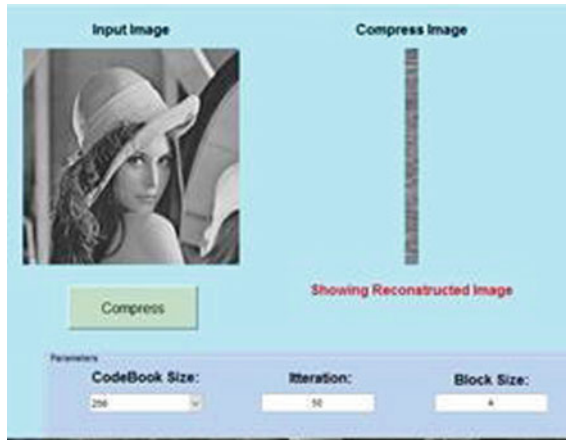
The tradeoff between the number of codeword selected and the extent to which distortion in the image is bearable must be decided [11]. The traditional vector quantization techniques failed to cope up with this requirement to some extent. Thus, for designing an optimal codebook design, the principles of clustering are applied to the traditional vector quantization algorithm. Image is initially divided into  $n \times n$  non-overlapping subdivisions. The similar data vectors are then identified from these subdivisions (blocks) and placed into similar groups. Codebook is generated with the reference input vectors, and then we apply a quantization process. Now, instead of selecting a separate codeword for each of the vectors, the same codeword

**Fig. 4** Structure of codebook



is selected for all the vectors that are placed in the same group and the process of vector quantization is carried out for generation of an optimal codebook. Thus, clustering reduces the number of codewords to be used in the codebook design [3]. The lesser the number of codewords used the higher is the compression ratio achieved and lesser is the image distortion. The identical index of this codeword is transmitted at output file. Figure 5 shows the image division into  $4 \times 4$  non-overlapping blocks and the initial codebook generated using vector quantization.

Efficiency of genetic algorithm depends on fitness, and it is used for the selection of chromosomes. Fitness is calculated with the help of MSE.



**Fig. 5** Image division into  $4 \times 4$  non-overlapping block and codebook generated using vector quantization

### ***3.3 Algorithm for Initial Codebook Generation is Given as Follows***

---

Algorithm for initial codebook generation

---

Step 1: Input image as training set

Step 2: The input image is converted into appropriate structure by dividing it into non-overlapping blocks

Step 3: The blocks of image to be selected for a codeword are picked up on the basis of the code in MATLAB

Step 4: Calculate fitness of all solutions

Step 5: Evaluate the best fitness function

for  $I = 1: n$  hpsnr = 0

for  $j = 1: q$  level if tpsnr > hpsnr

hpsnr = tpsnr;

best\_unit = j;

end

next j

next i

end

---

Number of blocks that belong to the codebook (best-unit) + 1; allocate the block position to the blocks that belong to the best unit in the array of codebook; here,  $n$  is the number of blocks in the input image, and  $q$  level corresponds to the quantization level used.



### 3.4 Quality Measures Are as Follows

#### 3.4.1 Peak Signal-to-Noise Ratios (PSNR)

In image compression, PSNR measures appearance the quality of the image to a detrimental level. PSNR equation is as follows (1),

$$\text{PSNR} = 10 \log_{10} \left( 255 \times \frac{255}{\text{MSE}} \right) \quad (1)$$

$$\begin{aligned} \text{PSNR} &= 10 \cdot \log_{10} \left( \frac{\text{MAX}_I^2}{\text{MSE}} \right) \\ &= 20 \cdot \log_{10} \left( \frac{\text{MAX}_I}{\sqrt{\text{MSE}}} \right) \\ &= 20 \cdot \log_{10}(\text{MAX}_I) - 10 \cdot \log_{10}(\text{MSE}) \end{aligned}$$

Here,  $\text{MAX}_I$  is the maximum possible pixel value of the image. Let us consider pixels are 8 bits per sample, then it is 255. In the absence of noise, images  $I$  and  $K$  are same and thus the MSE is zero. Practically, PSNR is around 30 dB.

#### 3.4.2 Mean Square Error (MSE)

Calculate variations among the actual and reference pixel, and its average is represented by mean squared error. MSE is measured as (2).

$$\text{MSE} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2 \quad (2)$$

#### 3.4.3 Structural Similarity (SSIM) Index

It is a quality measure, and its score is represented by an Eq. (3)

$$\text{SSIM}(x, y) = [l(x, y)]^\alpha \cdot [c(x, y)]^\beta \cdot [s(x, y)]^\gamma \quad (3)$$

SSIM is concerned with quality measurement method to local region using a sliding window approach. Size of  $B \times B$  moves horizontally and vertically from top-left corner to bottom-right corner, all over the image. Accuracy of SSIM is great but acquires more computational cost.

### 3.4.4 Average Difference (AD)

The difference between the input image and output image in image compression is represented as (4)

$$AD = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (x(i, j) - y(i, j)) \quad (4)$$

where  $x(i, j)$  represents the input (reference) image and  $y(I, j)$  represents output image.

## 4 Codebook Optimization Models

### 4.1 Modified Genetic Algorithm

Modified genetic algorithm is used as a natural characteristic for optimization of codebook design in vector quantization for image compression.

**Modified GA steps are as follows:**

---

**Procedure:** Modified Genetic Algorithm

---

Generation of initial chromosomes

While, iterations  $\tau < \tau_{\text{off}}$

for  $\tau = 1$  to 100

**Step 3:** Calculate the fitness of all solutions

**Step 4:** Evaluate the best fitness function

---

**Step 5:** Selection operator: Select the best fit parent chromosome

**Step 6:** Crossover operator: Select the best fit children chromosomes

**Step 7:** Mutation operator: Mutate other children chromosome than the best fitting one

**Step 8:** Find best fitness solutions

**Step 9:** Update the best fitness chromosome

**Step 10:** Termination: if the criterions are satisfied

**Step 11:** If not then Return step 2

**Step 12:**  $\tau = \tau + 1$

end for end while

End

---

Quality of codebook depends on fitness, and fitness is better for minimum error in reconstructed image. Clustering is used to cluster similar data into same block. Genetic operators are selection, crossover, and mutation.

**Selection of the best fit children chromosome:**

(1) Generate random population of  $n$  chromosomes.

- (2) Calculate the fitness of each chromosome in the population.
- (3) Select two parent chromosomes those who are best fit.
- (4) Crossover the parents to generate a children.
- (5) Apply mutation probability.
- (6) Place best fit children in a new population.

## 4.2 Particle Swarm Optimization Algorithm

This method is for optimizing the codebook through the significance of each particle and exploit pbest and best to focus the automatic clustering. Clustering reduces the computational complexity, while the stochastic search method of PSO algorithm helps in obtaining an optimal solution [3, 6]. PSO is a multi-agent approach based on the bird flocking and used for optimization. It is starting with a random particle solution. They move to search for optima by renewing iterations and each particle by two “best” values. The best solution (fitness) it has reached and referred as pbest. Another “best” value in the population is a global best and referred as gbest. In this model,  $x_i$  is particle position and  $v_i$  is velocity. The best position in the vector is pbest. Updated velocity is found with the help of Eqs. (5) and (6).

$$Vid' = w * vid + c1 * r1 * (pbest - xid) + c2 * r2 * (gbest - xid) \quad (5)$$

$$Xid' = xid + Vid' \quad (6)$$

Inertia weight is represented by  $w$ , random numbers are  $r1$  and  $r2$ , coefficient of the self-recognition component is  $c1$ , social component is  $c2$ , and both are positive constant. Based on the best position, particle leads effectively in the search space. The development of PSO [3] is based on a particle’s position toward a target location.

### Steps of PSO:

---

**Procedure: Particle swarm optimization**

---

**Step 1:** Selection of particle

---

**Step 2:** Calculate fitness

---

**Step 3:** Update pbest

---

**Step 4:** Select gbest

---

**Step 5:** Particle velocity calculation

---

**Step 6:** Renew position of the particle

---

**Step 7:** Steps are repeated till termination criteria are achieved

---

Velocity and position of each particle are randomly initialized. Particle maintains its pbest (local best position) and gbest (global best position). To renew the position of given particle, current position of these particles should be better than the previous one.

## 5 Experimental Results

The PSO and GA were tested in MATLAB. In this database included category 1, 2, 3, and 5 images were downloaded from standard database and for category 4 were manually collected.

The analysis of the compression ratio and BPP is demonstrated in Table 2. Results are compared with other algorithms [17–21]. The original image is shown in following Fig. 6, and decompressed by modifying GA is shown in Fig. 7, and Fig. 8 shows decompressed by PSO. CR analysis of optimization algorithms is shown Fig. 9.

We are also demonstrating results on gray-level image. The parameter used in the experimentation is population size: 50 and 100. We continue this uniformity testing. We have set different iterations: 10 and 50. In the experimentation, we divide the image into  $4 \times 4$  and  $8 \times 8$  blocks to encode. We have designed a codebooks using optimization algorithms. In the output file, send indexes to represent every codeword. Initial population is selected with best fitness chromosomes. It has been observed [1, 2] that if the initial chromosomes are fit, then the solution of clustering would be best

**Table 2** Types of images and sets

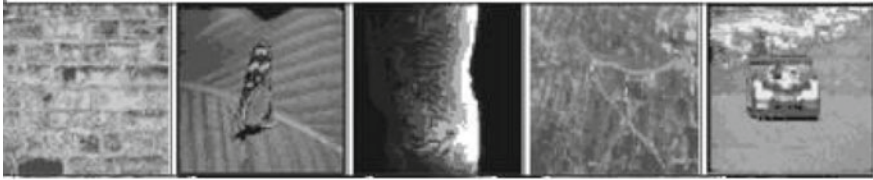
Type of image	Total images in each set
Texture images (category 1)	50 images
Nature image set (category 2)	30 images
Medical image set (category 3)	30 images
Satellite image set (category 4)	10 images
Miscellaneous set (category 5)	50 images



**Fig. 6** Original image: **a** texture image, **b** nature image, **c** medical image, **d** satellite image, and **e** miscellaneous image



**Fig. 7** Decompressed image by modifying GA: **a** texture image, **b** nature image, **c** medical image, **d** satellite image, and **e** miscellaneous image



**Fig. 8** Decompressed image by PSO: **a** texture image, **b** nature image, **c** medical image, **d** satellite image, and **e** miscellaneous image



**Fig. 9** CR analysis of optimization algorithms

as it has much depends on them. The result shows that quality measure parameters are achieved by PSO which is greater than the modified genetic algorithm. Training sets (Fig. 10) are taken with size 512X512, and reconstructed image is shown in Fig. 11. Performance analysis of optimization algorithms are shown in Figs. 12, 13, 14, 15, 16, 17, 18 and 19. Comparison of quality measures is shown in Tables 3 and 4.



**Fig. 10** Training image of Leena (512 × 512)



Fig. 11 Reconstructed image of Leena

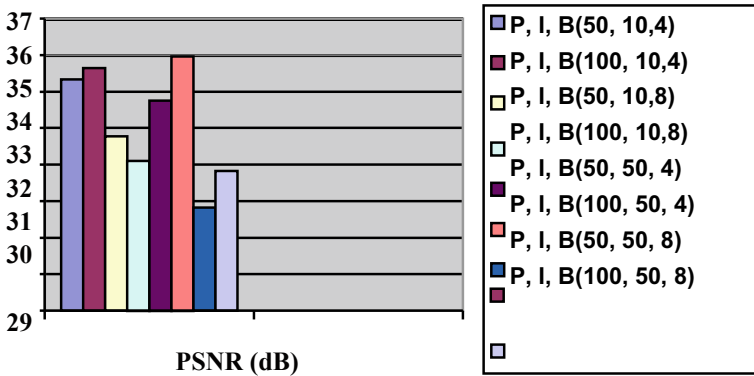


Fig. 12 PSNR analysis of optimization algorithms

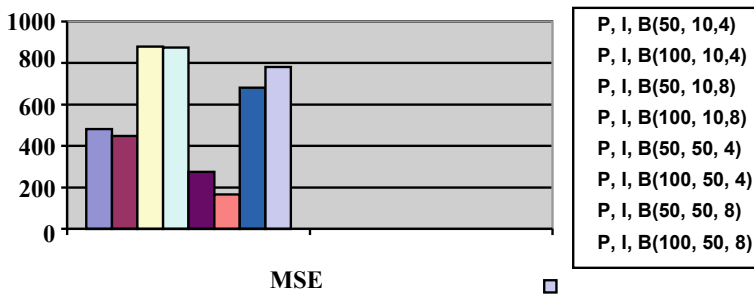


Fig. 13 MSE analysis of optimization algorithms

## 6 Conclusion

Analysis of image compression has been simulated in MATLAB, and the experimentations have been carried out. The performance of the PSO algorithm is to be compared with a modified GA algorithm in terms of analysis on error measures

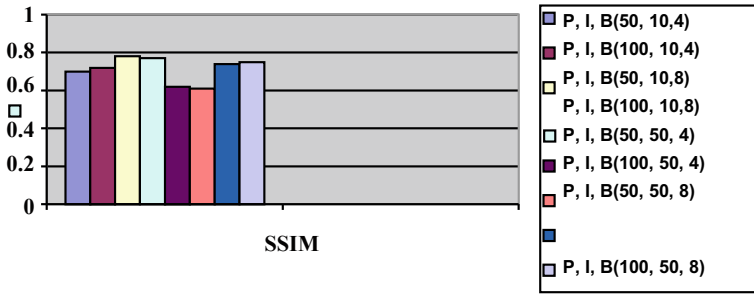


Fig. 14 SSIM analysis of optimization algorithms

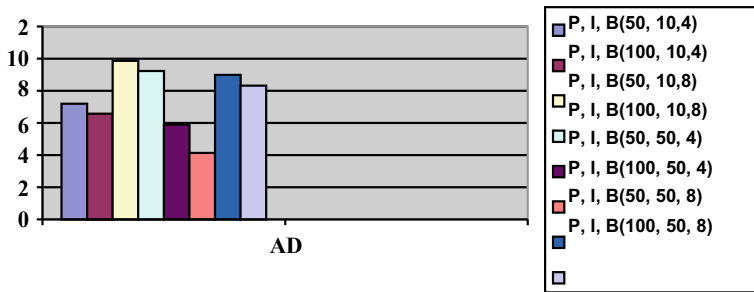


Fig. 15 AD analysis of optimization algorithms

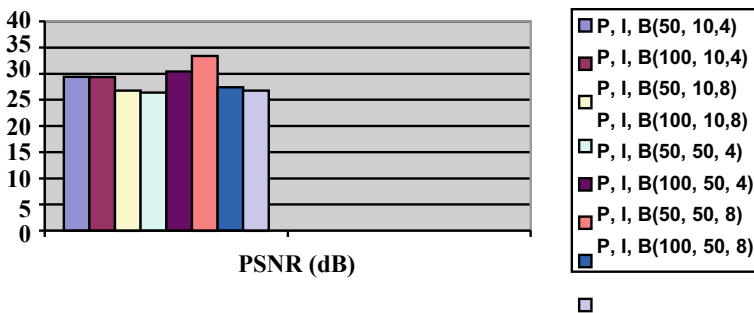


Fig. 16 PSNR analysis of optimization algorithms

like PSNR, MSE, SSIM, and AD. Experimental results show that the PSO is better than the modified genetic algorithm. We have shown that PSO outperforms the image compression standard. A bitrate, BPP, is 8. The compression ratio for PSO is 90.90%, for FF it is 75%, GWO it is 83.33%, and 80% of modified GA. The experimental results indicate clear superiority of PSO over the existing techniques. From result, we have proved that the % of the CR of PSO model is best among other models.

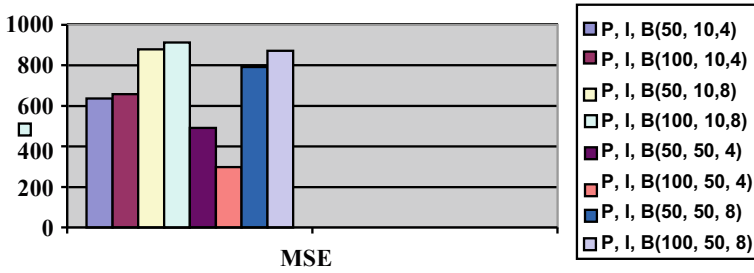


Fig. 17 MSE analysis of optimization algorithms

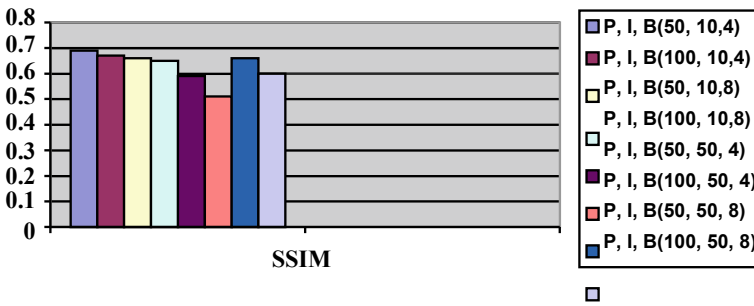


Fig. 18 SSIM analysis of optimization algorithms

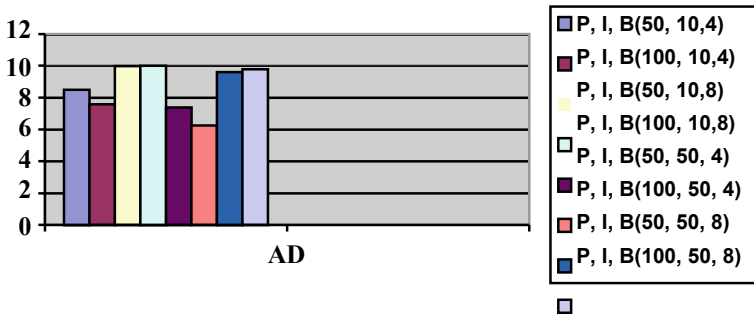


Fig. 19 AD analysis of optimization algorithms



**Table 3** Comparison of quality measures block size of  $4 \times 4$  and  $8 \times 8$ , iteration 10 and 50, population size 50 and 100 using the PSO algorithm on grayscale image  $512 \times 512$  and codebook size = 256

Population size ( <i>P</i> )	Iteration ( <i>I</i> )	Block size ( <i>B</i> )	PSNR (dB)	MSE	SSIM	Average difference
50	10	4	35.34	480.82	0.70	7.19
100	10	4	35.64	448.29	0.72	6.58
50	10	8	33.77	878.73	0.78	9.87
100	10	8	33.10	875.11	0.77	9.23
50	50	4	34.76	275.37	0.62	5.90
100	50	4	35.96	166.13	0.61	4.13
50	50	8	31.83	680.87	0.74	8.99
100	50	8	32.83	780.87	0.75	8.33

**Table 4** Comparison of quality measures with block size of  $4 \times 4$  and  $8 \times 8$ , Iteration 10 and 50, Population size 50 and 100 using modified GA algorithm on grayscale image  $512 \times 512$  and codebook size = 256

Population size ( <i>P</i> )	Iteration ( <i>I</i> )	Block size ( <i>B</i> )	PSNR (dB)	MSE	SSIM	Average difference
50	10	4	29.41	636.49	0.69	8.51
100	10	4	29.36	658.37	0.67	7.68
50	10	8	26.77	874.44	0.66	9.99
100	10	8	26.36	913.17	0.65	10.00
50	50	4	30.44	491.36	0.59	7.38
100	50	4	33.41	298.24	0.51	6.26
50	50	8	27.40	791.83	0.66	9.62
100	50	8	26.79	871.46	0.60	9.79

## References

1. Rajkumar S, Malathi G (2016) A comparative analysis on image quality assessment for real time satellite images. Indian J Sci Technol 9(34). <https://doi.org/10.17485/ijst/2016/v9i34/96766>. ISSN (Print): 0974-6846
2. Sánchez D, Melin P, Castillo O (2020) Modular granular neural network optimization using the firefly algorithm applied to time series prediction. In: Yang X-S (ed) Nature-inspired computation and swarm intelligence. Academic Press, pp 199–216 (Chapter12). <https://www.sciencedirect.com/science/article/pii/B9780128197141000233>
3. Panda M, Das B (2019) Grey wolf optimizer and its applications: a survey. In: Nath V, Mandal J (eds) Proceedings of the third international conference on microelectronics, computing and communication systems. Lecture notes in electrical engineering, vol 556. Springer, Singapore. [https://doi.org/10.1007/978-981-13-7091-5\\_17](https://doi.org/10.1007/978-981-13-7091-5_17)
4. Mirjalili S, Lewis A (2016) The whale optimization algorithm. In: Advances in engineering software, vol 95, pp 51–67. ISSN 0965-9978. <https://doi.org/10.1016/j.advengsoft.2016.01.008>. <https://www.sciencedirect.com/science/article/pii/S0965997816300163>

5. Cui Z, Li F, Zhang W (2019) Bat algorithm with principal component analysis. *Int J Mach Learn Cyber* 10:603–622. <https://doi.org/10.1007/s13042-018-0888-4>
6. Pang C-Y, Zhou R-G, Hu B-Q, Hu WW, El-Rafei A (2019) Signal and image compression using quantum discrete cosine transform. *Inf Sci* 473:121–141
7. Ernawan F, Kabir N, Zamli KZ (2017) An efficient image compression technique using Tchebichefbit allocation. *Opt Int J Light Electron Opt* 148:106–119
8. Roy SK, Kumar S, Chanda B, Chaudhuri BB, Banerjee S (2018) Fractal image compression using upper bound on scaling parameter. *Chaos Solitons Fractals* 106:16–22
9. Brahimi T, Laouir F, Boubchir L, Ali-Chérif A (2017) An improved wavelet-based image coder for embedded greyscale and colour image compression. *AEU-Int J Electron Commun* 73:183–192
10. Xiao B, Lu G, Zhang Y, Li W, Wang G (2016) Lossless image compression based on integer discrete Tchebichef transform. *Neuro Comput* 214:587–593
11. Turcza P, Duplaga M (2017) Near-lossless energy-efficient image compression algorithm for wireless capsule endoscopy. *Biomed Sig Process Control* 38:1–8
12. Zuo Z, Lan X, Deng L, Yao S, Wang X (2015) An improved medical image compression technique with lossless region of interest. *Opt Int J Light Electron Opt* 126(21):2825–2831
13. Chaurasia VS, Chaurasia V (2016) Statistical feature extraction based technique for fast fractal image compression. *J Vis Commun Image Represent* 41:87–95
14. Hussain AJ, Al-Fayadh A, Radi N (2018) Image compression techniques: a survey in lossless and lossy algorithms. *Neuro Comput* 300:44–69
15. Fu C, Yi Y, Luo F (2018) Hyperspectral image compression based on simultaneous sparse representation and general-pixels. *Pattern Recogn Lett* 116:65–71
16. Ji XX, Zhang G (2017) An adaptive SAR image compression method. *Comput Electr Eng* 62:473–484
17. Skorsetz M, Artal P, Bueno JM (2018) Improved multiphoton imaging in biological samples by using variable pulse compression and wavefront assessment. *Opt Commun* 422:44–51
18. Rashid F, Miri A, Woungang I (2016) Secure image deduplication through image compression. *J Inf Secur Appl* 27–28:54–64
19. Huang H, He X, Xiang Y, Wen W, Zhang Y (2018) A compression-diffusion-permutation strategy for securing image. *Sig Process* 150:183–190
20. Shakya S, Pulchowk LN (2020) A novel bi-velocity particle swarm optimization scheme for multicast routing problem. *IRO J Sustain Wireless Syst* 2:50–58
21. Dhaya R, Kanthavel R (2020) Comprehensively meld code clone identifier for replicated source code identification in diverse web browsers. *J Trends Comput Sci Smart Technol (TCSST)* 2(02):109–119