International Conference on Information and Communication Technology and Its Applications (ICTA 2016)
Federal University of Technology, Minna, Nigeria
November 28 – 30, 2016

Automatic Gray Image Contrast Enhancement using Particle Swarm and Cuckoo Search Optimization Algorithms

[1]A. J. Onumanyi, [2]F. Idris, [2]M. B. Abdullahi, [1]H. Bello-Salau, [1]S. O. Aliyu, & [1]M. Okwori
[1]Dept. of Telecommunication Engineering, Federal University of Technology, Minna
[2]Dept. of Computer Science, Federal University of Technology, Minna
adeiza1@futminna.edu.ng, fatiidris2012@gmail.com, el.bashir02@futminna.edu.ng,
bellosalau@gmail.com, salihu.aliyu@futminna.edu.ng, michael_okwori@yahoo.com

***Abstract*—In this paper, we report on the investigation of two different metaheuristic based algorithms for Gray Image (GI) enhancement. First, we investigated the Particle Swarm Optimization (PSO) algorithm under certain parameter settings for the GI enhancement task, and followed with the Cuckoo Search (CS) algorithm for the same task. Then, we proposed an algorithmic procedure for computing a new set of objective measures for quantifying the performance of any image enhancement algorithm. Comparative analyses were conducted alongside classical approaches such as the Linear Contrast Stretching (LCS) and the Histogram Equalization (HS) techniques. Our findings revealed that the CS and the PSO algorithms provide better performance than the popularly used LCS and HE techniques. However, between the PSO and the CS algorithm, the CS performed better on more images than the PSO. These results obtained using the proposed metrics were seen to be clearly consistent with the enhanced images and thus, we concluded that autonomous GI enhancement methods based on metaheuristic optimization algorithms produce efficient results, and can effectively replace our dependence on subjective human judgment.**

***Keywords* — Cuckoo Search, Contrast Enhancement, Gray Image, Metaheuristic, Particle Swarm Optimization,**

I. INTRODUCTION

Nowadays, digital images have become a typical way of acquiring, storing and communicating information among people, corporations, businesses and security outfits [1]. Thus, it has become pertinent to ensure the integrity of digital images, particularly those used for sensitive purposes in pattern recognition, forensics, and a host of other applications.

In this regard, an important area of focus is gray image enhancement [2]. Several works have tried to improve the contrast of Gray Images (GI), however, these techniques have been either fully manual, that is, humans are required to identify areas for improvement, or partially automated, where humans need to assess the enhancement performance to make conclusions. For most automated techniques, it has been observed that they often depend only on the global information of the image, without consideration for local details [3]. Furthermore, because of their dependence on the specific image being processed, most automated techniques lack the capacity for generalization. Despite these limitations, full automation is evidently required for most new applications in areas such as pattern recognition, forensics and robotics, and thus the need for better techniques.

In this paper, we report on the investigation of two metaheuristic algorithms for autonomous GI enhancement. To achieve this, we adopted the transformation and evaluation functions in [3] and applied them for GI enhancement. First, we investigated the Particle Swarm Optimization (PSO) technique based on certain parameter settings. Secondly, we explored the Cuckoo Search (CS) algorithm for the same task. These algorithms were chosen owing to their respective high performance output, as noted in the literature [4]. Each algorithm was modified and details of the modifications are presented in appropriate sections. The results of the different algorithms were analyzed using a set of newly proposed metrics and findings are presented herein to justify the effectiveness of the metaheuristic algorithms. An algorithm for computing these metrics is also presented and readers are provided with output images to enable them cross evaluate between the proposed metrics and the reader's perception of the enhanced images.

The rest of the paper is organized as follows: Section II provides a brief review of the relevant literature. In Section III, we present details of the methodology used, while results and analysis are provided in Section IV. Conclusion is drawn in Section V.

II. REVIEW OF RELEVANT LITERATURE

There are several reported works on Gray Image (GI) enhancement. These methods can be broadly divided into point operations, spatial operations, transform operations, and pseudocolouring methods [3]. Techniques under point operation (also termed indirect method) include contrast stretching, window slicing and histogram modeling [3]. These are the simplest and most popular methods for GI enhancement, thus, they are widely deployed in the literature. However, they have more global effect than local effect; thus, they suffer from over stretching of the image contrast. Indirect methods typically adjust the image histogram to improve the entropy. On the other hand, spatial operations (or direct methods) establish criterions of contrast measurement and enhance the image by improving the measure [5]. In [5], Fuzzy logic was used as an adaptive direct enhancement method based on fuzzy entropy principle and fuzzy set theory. Authors claimed that the proposed technique performed better than the Adaptive Contrast Enhancement technique; however, evaluation was done subjectively. In 2004, Munteanu and Rosa [3] proposed a transformation function for contrast enhancement and used Evolutionary Algorithm (EA) as a global search strategy for the best enhancement. Authors became one of the first to use heuristic algorithm for GI enhancement, and they used both subjective and objective methods for evaluation, and showed the superiority of their method over Linear Stretching (LS) and Histogram Equalization (HE). In 2005, Russo [6] proposed an objective evaluation technique based on the histograms of the edge gradients. Though shown to outperform both linear and nonlinear unsharp masking technique, Rosso's technique produces overshoots along the object contours. Kwok et al., [7] in 2006 proposed an intensity-preserving technique for contrast enhancement using Particle Swarm Optimization (PSO). The PSO technique was used to obtain proper gamma-factor values for the enhancement process. The use of mean-intensity as the objective measure of evaluation is insufficient to make conclusions, as such, broad measures are required. In [8], authors used PSO to maximize the information content of an enhanced image using Munteanu's functions in [3]. Gorai et al., [8] showed that PSO performed better than GA, LS and HE. Particularly, Ghosh et al., in [4] explored the use of Cuckoo Search (CS) algorithm for image enhancement. It was concluded that CS provides better performance compared to PSO, Genetic Algorithm (GA), Linear Contrast Stretching (LCS) and Histogram Equalization. Similarly, other works in [9][10][11][12][13] have made efforts to enhance GIs, and the trending conclusion is that metaheuristic algorithms tend to provide better image enhancement based on the use of transformation functions. However, these methods often lack standard objective measures for measuring their effectiveness. Thus, in addition to investigating both PSO and CS in our work, we proposed new objective measures for quantifying an algorithm's performance.

III. Methodology

In this section, we provide details on the two metaheuristic based algorithms used here for GI enhancement, namely Particle Swarm Optimization (PSO) and Cuckoo Search (CS) algorithms. In addition, we provide an algorithm for computing a new set of metrics for evaluating any GI enhancement algorithm. Details of these algorithms are provided in their respective subsections.

A. Input Parameters

Let the image to be enhanced be denoted as I, with dimensions $(R \times C)$. Now, based on the user's requirement, the image, I, can be resized to a smaller dimension to improve processing speed, however, in this work, we used the original dimension given as $(R \times C)$. The image, I, is converted to gray scale, G, with same dimension, $(R \times C)$. By using a local window size, $LW = 3$, we computed the local mean, using the $LW \times LW$ window size, the global mean of the entire image, G, and the local standard deviation, σ. These parameters served as the basic requirements for running the GI process.

B. The Image Transformation and Evaluation Process

To begin, it is worth noting that an image transformation function typically changes the intensity value of a gray image pixel, $G_{i,j}$ for $i = 1, 2, ..., R$ and $j = 1, 2, ..., C$, to a different value, $F_{i,j}$ for $i = 1, 2, ..., R$ and $j = 1, 2, ..., C$. Thus, the transformation function we used in this work is given as [3]

$$\begin{aligned} F_{i,j} &= T(G_{i,j}) \qquad \forall i \in R; j \in C \qquad (1) \\ &= \left(\kappa \frac{\mu_G}{\sigma_{i,j+b}}\right) \times \left(G_{i,j} - c \times \mu_{i,j}\right) + \left(\mu_{i,j}\right)^a \end{aligned}$$

where a, b, c, and κ are the parameters of the enhancement kernel to be optimized, having the following typical values: $0.5 < \kappa < 1.5$; $0 \leq a \leq 2$; $\mu_G < b < 0.5$, and $0 \leq c \leq 1$ [3].The transformed or enhanced image is obtained by computing $F_{i,j}$ for all i, j using (1). Subsequently, the number of edges, N_e in $F_{i,j}$ is computed using a Sobel detector. The Sobel detector produces an edge image, $E_{i,j}$ containing ones at pixels describing the image's edge pixels in $F_{i,j}$, and zeroes at other non-edge pixels. Because $E_{i,j}$ contains only binary representation of pixels corresponding to edges in $F_{i,j}$, the total number of edges in $E_{i,j}$ can be computed as:

$$N_e = \sum_{i=1}^{R} \sum_{j=1}^{C} E_{i,j} \qquad (2)$$

The intensity, $\xi_{i,j}$ of each pixel is obtained as

$$\xi_{i,j} = E_{i,j} \bullet F_{i,j}; \qquad \forall i \in R; j \in C \qquad (3)$$

where $(\bullet)$ denotes element wise multiplication. Thus, the total intensity of $F_{i,j}$ is given as

$$\Phi = \sum_{i=1}^{R} \sum_{j=1}^{C} \xi_{i,j} \qquad (4)$$

To evaluate the goodness of the enhanced image, we used the fitness function given in [3] as follows

$$Z = \log(\log(\Phi)) \times \left(\frac{N_e}{R \times C}\right) \times \exp(H(F_{i,j})) \qquad (5)$$

where $H(F_{i,j})$ is the entropy of the enhanced image, $F_{i,j}$. Thus, having established the requisite functions for transformation in (1) and for the evaluation in (5), the two metaheuristic algorithms, PSO and CS were then used to obtain the optimum values of a, b, c and κ, in accordance with their constraints, using (5) as the fitness function.

C. Use of Particle Swarm Optimization (PSO) Algorithm

We present here the use of Particle Swarm Optimization (PSO) algorithm [14] as modified for the enhancement process. First, we state the necessary functions used in PSO as

$$\begin{aligned} v_{t+1}^n &= w^n v_t^n + c_1^n r_1(gbest - x_t^n) + c_2^n r_2(pbest(n) - x_t^n) \\ x_{t+1}^n &= x_t^n + v_{t+1}^n \end{aligned} \quad (6)$$

where x_t^n is the position of the n^{th} particle, v_{t+1}^n denotes the next velocity of an n^{th} particle, w^n is the inertial weighting, r_1 and r_2 are randomly generated numbers within the range 0 and 1, and c_1^n and c_2^n are the social and cognitive components of the kernel, while $gbest$ and $pbest$ are the global and personal best values of the entire particle population, and individual particle, respectively. Next, a population size, P, is set for the possible number of solutions (or particles) to be used by the algorithm. The dimensions of each particle d, is given as the number of parameters to be optimized ($d = 4$, in this case). Next, the initial random values were generated for a, b, c, and κ, and these values were used in (1) to transform the image $G_{i,j}$ into $F_{i,j}$, while evaluating the fitness of $F_{i,j}$ using (5) based on the values of a, b, c, and κ. At an initial time t, the fitness of each particle, n is stored as $pbest$ (for $n = 1, 2, \ldots, P$), while the global best value in the population of all particles is stored as $gbest$. The subsequent steps taken by the algorithm are as follows:

1) *For each particle, n = 1 to P, do*
2) *Compute the fitness value of each particle, n using (5), after transformation using (1)*
3) *Compare the pbest(t) and pbest(t+1), and do*
   a. *If pbest(t+1) > pbest(t)*
   b. *Then, pbest(t+1) is made the current best value of the particle.*
4) *Return to 1, and do for all P*
5) *Obtain the gbest at t+1*
6) *If gbest(t+1) > gbest(t)*
7) *Then, gbest(t+1) is made the current global at t+1.*
8) *Thus, compute the next value of the velocity and the particles using (6)*
9) *Return to 1, until P.*

D. The use of Cuckoo Search (CS) Algorithm

The Cuckoo Search (CS) Algorithm using Levy flight [15] is applied here for image enhancement and the kernel function used for finding new solutions is given as:

$$x_{t=1}^n = x_t^n + \alpha \oplus Levy(\lambda) \quad (7)$$

where $\alpha > 0$ is the step size related to the scale of the problem of interest, in most cases, $\alpha = 1$ is normally used, and $1 < \lambda < 3$ is the Levy distribution parameter, while x_t^n and $x_{t=1}^n$ are the current and next solutions with dimension, d. In our work, the CS algorithm was used as follows:

1) *Let the number of nests (or different solutions, similar to particles in PSO) be n, and the dimension of each particle be d (where d = 4). Let the probability of discovering an alien egg (or solution) in a nest be pa. Thus, for a number of iteration, $Niter$, the rest of the process ensues as follows:*
2) *Set the lower and upper bounds for the parameter constraints based on dimension, d,*
3) *Obtain the random initial solutions (or nests),*
4) *For each iteration, until Niter, do*
5) *Get a cuckoo randomly by Levy Flights*
6) *Evaluate each solution (or nest) using (5), after transformation using (1),*
7) *Obtain the global best value among all nest as Maxfit*
8) *If Maxfit(t+1) > Maxfit(t)*
9) *Update the new global best*
10) *End if*
11) *Empty a fraction, pa, of the worst nests,*
12) *Update the new nests using (7)*
13) *Keep the best solutions*
14) *Return to 1, until Niter is completed.*

Summarily, the optimum values of a, b, c, and κ, computed by the CS algorithm will typically produce the best enhanced image at the end of the iteration.

E. Proposed Objective Evaluation Measures

We propose here an algorithmic procedure for measuring the performance of a GI enhancement algorithm. The procedure is as follows: Let the original and enhanced gray image be $G_{i,j}$ and $F_{i,j}$, respectively, for $i \in R$ and $j \in C$. Then, the algorithm uses a (3×3) window size to compute the local variance, σ_G and σ_F, of both $G_{i,j}$ and $F_{i,j}$, respectively. Next, it uses Otsu's algorithm to compute an optimum threshold value, T_G from σ_G. Finally, to compute the measurement metrics, let the count of the Detailed and Background Variance of both the original and enhanced image be denoted as D_O and B_O, and D_E and B_E, respectively. The algorithm computes these metrics as follows:

1) *For i = 1 to R, do*
2) *For j = 1 to C, do*
3) *If $\sigma_G(i, j) \geq T_G$*
4) *Increment D_O set by 1*
5) *Else*
6) *Increment B_O set by 1*
7) *End*
8) *If $\sigma_F(i, j) \geq T_G$*
9) *Increment D_E set by 1*
10) *Else*
11) *Increment B_E set by 1*
12) *End*
13) *End*
14) *End*

At the end of Line 14, the algorithm computes the overall Detailed and Background Variance of both the original and enhanced image by adding all the counts in D_O and B_O, and D_E and B_E, respectively. In addition, by using a Sobel detector, the number of edges denoted as N_O and N_E respectively for the original and enhanced image are also considered for evaluating the algorithm's performance.

IV. Results and Discussion

For evaluation purpose, four different images (see Figs. 1 – 4) were used for running the PSO and CS algorithm alongside classical techniques such as the Linear Contrast Stretching (LCS) and Histogram Equalization (HE) techniques. The images used for evaluation have various properties relevant for evaluating these algorithms, such as a variety of both small and large number of pixels (see Table 1a), different shades, darkness, and representing different applications, e.g Fingerprint image for finger print analysis. The parameters used for the metaheuristic algorithms are provided in Table 1b. Before proceeding, it should be noted that the term Background (BV) and Detailed Variance (DV) are only similar terminology-wise to the metrics used in [3], but different in their technical interpretation. Here, BV describes the number of pixels that belong to the background image (or noisy component) of the image, while DV describes the number of pixels that belong to the foreground image (or true signal component). Both metrics form an effective measure for evaluating any GI enhancement algorithm. We provide the enhanced images (see Figs. 1 – 4) outputted by each algorithm so that readers can make their subjective evaluation and then proceed to corroborate their judgment using the corresponding output metrics in Tables 2 – 5. Thus, by visually analyzing each algorithm's output (see Figs. 1 – 4) and comparing them with their corresponding objective measures (see Tables 2 – 5), it can be clearly seen that the objective measures closely reflect the true outcome of the enhancement process. Consequently, it can be seen that a well enhanced image should have lower BV and higher DV along with more number of edges than its original version.

Table 1a: Images used and their respective dimensions

Figure	Image Name	Size (Pixels)
Fig.1	Coins	246×300
Fig.2	Cameraman	256×256
Fig.3	Pout	291×240
Fig.4	Fingerprint	480×640

Table 1b: Parameter Settings for the Metaheuristic Algorithms

Method	Generations	Pop. Size	Parameters
PSO	50	25	$C_1, C_2 = 0.6; w = 1$
CS	50	25	$P_a = 0.25$

Table2:PerformanceEvaluationfor"Coins"Image

	PSO	CS	LCS	HE
B_O	70453	70453	70453	70453
B_E	52610	49506	69548	69671
D_O	3347	3347	3347	3347
D_E	21190	24294	4252	4129
N_O	2103	2103	2103	2103
N_E	9808	10529	2309	2095

Table3:PerformanceEvaluationfor"Cameraman"Image

	PSO	CS	LCS	HE
B_O	61198	61198	61198	61198
B_E	52713	50246	60174	60548
D_O	4338	4338	4338	4338
D_E	12823	15290	5362	4988
N_O	2503	2503	2503	2503
N_E	6281	7005	2808	2749

Table4:PerformanceEvaluationfor"Pout"Image

	PSO	CS	LCS	HE
B_O	68852	68852	68852	68852
B_E	63628	50514	66696	66461
D_O	988	988	988	988
D_E	6212	19326	3144	3379
N_O	1519	1519	1519	1519
N_E	7437	15879	4271	5585

Table5:PerformanceEvaluationfor"Fingerprint"Image

	PSO	CS	LCS	HE
B_O	291820	291820	291820	291820
B_E	278541	279972	283368	284964
D_O	15380	15380	15380	15380
D_E	28659	27228	23832	22236
N_O	8775	8775	8775	8775
N_E	12409	12369	942	8923

Furthermore, it should be noted that the PSO and CS algorithms are statistical in nature, thus, they often provide different results on different runs. Consequently, the values provided here for their evaluations were averaged over 5 different runs. Over the different images used in this work, it can be seen that the two metaheuristic algorithms clearly outperform the classical LCS and HE techniques (see Tables 2 – 5). However, between the PSO and CS algorithm, the CS technique provided 12.78% performance gain over the PSO in the DV for "Coins" image, 16.14% gain over the PSO in the DV for "Cameraman" image, while the PSO provided a 4.99% gain over the CS in the "fingerprint" image. Interestingly, the CS achieved 67.86% gain over the PSO in the "Pout" image. Upon closer examination of the "Pout" image, it can be seen that it has the smallest Signal to Noise Ratio (SNR) based on the DV and BV of the original image, thus, it contains more noise. Consequently, the CS algorithm performed better on the image with high noisy content than other techniques. Though the CS provides better performance than the PSO, it should be noted that this performance was averaged over several runs. Thus, users could obtain variations for a single run wherein the PSO algorithm provides a better result than the CS (hence, justifying the need for averaging). However, against the classical methods (that is, LCS and HE), both metaheuristic algorithms consistently provide better performance whether on single or over several runs.

V. Conclusion

This paper has presented an investigation of two metaheuristic algorithms, namely Particle Swarm Optimization (PSO) and Cuckoo Search (CS) algorithm for the Gray Image (GI) enhancement task. As a contribution, the paper has provided consistent and objective measures that can be used to evaluate any GI enhancement algorithm. These measures are clearly consistent over the evaluation of four different images. It has been shown that the metaheuristic algorithms outperform two popular classical methods namely Linear Contrast Stretching (LCS) and Histogram Equalization (HE). However, between the CS and PSO algorithms, the CS algorithm performed better on more images than the PSO.

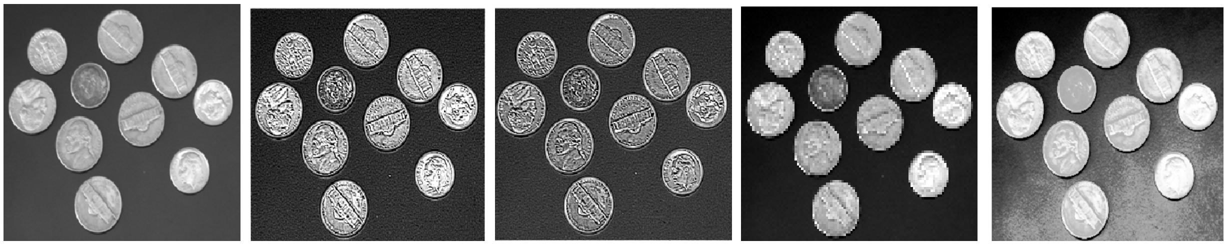

Fig. 1a: Original Fig. 1b: PSO Fig. 1c: CS Fig. 1d: LCS Fig. 1e: HE

Fig. 2a: Original Fig. 2b: PSO Fig. 2c: CS Fig. 2d: LCS Fig. 2e: HE

Fig. 3a: Original Fig. 3b: PSO Fig. 3c: CS Fig. 3d: LCS Fig. 3e: HE

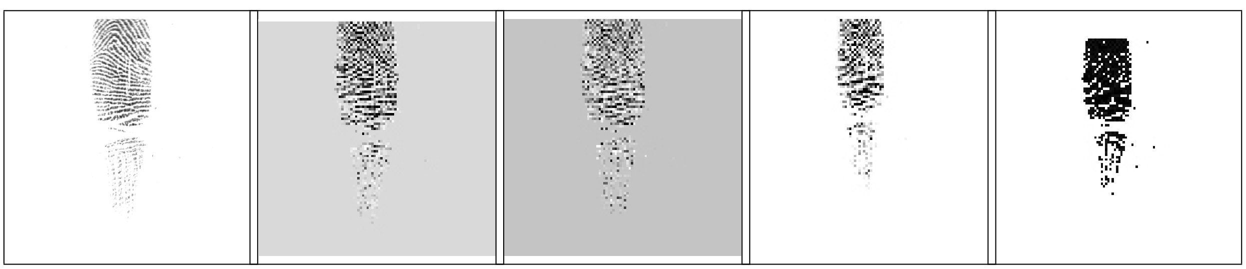

Fig. 4a: Original Fig. 4b: PSO Fig. 4c: CS Fig. 4d: LCS Fig. 4e: HE

While noting that these metaheuristic algorithms are highly statistical in nature, and often converge to solutions close to the optimal, it might be difficult to conclude which is better on single runs of the algorithm. However, on the average, the CS provides better performance. Summarily, this work adds to the body of evidence supporting the claim that metaheuristic algorithms possess the potential to replace subjective and manual methods based on human judgement in GI enhancement. Future works will provide a thorough, indepth and objective evaluation of different metatheuristic algorithms for the GI enhancement problem.

REFERENCES

[1] C. A. Schneider, W. S. Rasband, and K. W. Eliceiri, "Nih image to imagej: 25 years of image analysis," *Nat methods*, vol. 9, no. 7, pp. 671–675, 2012.

[2] R. Maini and H. Aggarwal, "A comprehensive review of image enhancement techniques," *arXiv preprint arXiv:1003.4053*, 2010.

[3] C. Munteanu and A. Rosa, "Gray-scale image enhancement as an automatic process driven by evolution," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 2, pp. 1292–1298, 2004.

[4] S. Ghosh, S. Roy, U. Kumar, and A. Mallick, *Gray Level Image Enhancement Using Cuckoo Search Algorithm*. Springer, 2014, pp. 275–286.

[5] H. Cheng, R. Min, and M. Zhang, "Automatic wavelet base selection and its application to contrast enhancement," *Signal Processing*, vol. 90, no. 4, pp. 1279–1289, 2010.

[6] F. Russo, "Automatic enhancement of noisy images using objective evaluation of image quality," *IEEE transactions on instrumentation and measurement*, vol. 54, no. 4, pp. 1600–1606, 2005.

[7] N. M. Kwok, Q. P. Ha, D. Liu, and G. Fang, "Intensity-preserving contrast enhancement for gray-level images using multi-objective particle swarm optimization," in *2006 IEEE International Conference on Automation Science and Engineering*. IEEE, 2006, pp. 21–26.

[8] A. Gorai and A. Ghosh, "Gray-level image enhancement by particle swarm optimization," in *Nature & Biologically Inspired Computing,*

2009. NaBIC 2009. World Congress on. IEEE, 2009, pp. 72–77.
[9] N. M. Kwok, Q. P. Ha, D. Liu, and G. Fang, "Contrast enhancement and intensity preservation for gray-level images using multiobjective particle swarm optimization," *IEEE Transactions on Automation Science and Engineering*, vol. 6, no. 1, pp. 145–155, 2009.
[10] H.-D. Cheng and H. Xu, "A novel fuzzy logic approach to contrast enhancement," *Pattern Recognition*, vol. 33, no. 5, pp. 809–819, 2000.
[11] S. Hashemi, S. Kiani, N. Noroozi, and M. E. Moghaddam, "An image contrast enhancement method based on genetic algorithm," *Pattern Recognition Letters*, vol. 31, no. 13, pp. 1816–1824, 2010.
[12] T. Celik and T. Tjahjadi, "Automatic image equalization and contrast enhancement using gaussian mixture modeling," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 145–156, 2012.
[13] G. Maragatham and S. M. M. Roomi, "A review of image contrast enhancement methods and techniques," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 9, no. 5, pp. 309–326, 2015.
[14] J. Kennedy, *Particle swarm optimization*. Springer, 2011, pp. 760–766.
[15] X.-S. Yang and S. Deb, "Cuckoo search via lvy flights," in *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*. IEEE, 2009, pp. 210–214.