Journal of Computer & Robotics 18 (2), Summer and Autumn 2025, 45-51

Computer & Robotics

Improve Color Image Clustering with Using Combination of GLS, Giza Pyramids Construction and K-Means Algorithm

Masoud Shahrian [a], Madjid Khalilian [a,*]

[a] *Department of Computer Engineering, Karaj Branch, Islamic Azad University, Karaj, Iran*

Received 28 October 2024, Accepted 22 February 2024

Abstract

Color image clustering is recognized as a complex challenge in the field of image processing. To improve the results of image clustering, meta-heuristic optimization algorithms can be employed. These algorithms are typically straightforward and can efficiently tackle problems in a short time frame, which offers distinct advantages. However, as the complexity of the problem increases, the solutions derived from these algorithms often fail to represent the optimal solution, resulting in limitations for their practical use. Thus, improving the performance and accuracy of existing algorithms is essential for broadening their applicability. Many meta-heuristic algorithms struggle to maintain an appropriate balance between exploration and exploitation during their update processes, and this issue has not been sufficiently addressed. In this research, we present a novel approach to image clustering. Our method integrates an enhanced Giza Pyramids Construction (GPC) with the Guided Learning Strategy (GLS) and k-means clustering. The GLS strategy assesses the standard deviation of historical positions of individuals across recent generations to evaluate population dispersion and deduce the type of guidance the algorithm requires at any given time. When the algorithm leans towards exploration, this strategy steers it towards exploitation, and vice versa. By identifying and addressing the algorithm's current needs, this strategy can significantly improve the performance of various optimization algorithms. Furthermore, the Giza Pyramids Construction, inspired by the historical practices of ancient Egypt, mathematically models the behavior of worker groups engaged in constructing large pyramids. We assess the effectiveness of our proposed algorithm in the context of color image clustering and compare the results against several established evaluators that can analyze internal cluster evaluations and inter-cluster distances. Our findings demonstrate that the proposed method achieves superior results compared to other state-of-the-art techniques, based on both objective and subjective evaluation metrics.

Keywords: Image processing, Color Image clustering, Giza Pyramids Construction, GLS, k-means Algorithm

1.Introduction

Image processing and computer vision are two innovative and diverse branches of artificial intelligence. By combining image processing techniques with machine learning algorithms, computers are able to visually perceive and comprehend various attributes of objects. Image clustering is a common practice in various areas of image processing, including image segmentation and object recognition. Clustering involves grouping data into components with similar properties, which may be inherent in the data or derived through computational processes. As the number of clusters decreases, the amount of information and detail used in clustering also decreases, resulting in a more straightforward clustering process [1]. The optimal clustering mode is achieved when the inter-cluster distance is the minimum and the inter-cluster distance is the highest [1]. Clusters provide models of data [2]. Clustering can separate data and information without any training and put clusters of similar data together [2]. Also, each cluster has a distance and difference in data from other clusters [3]. As a result, after clustering, an expert should interpret the clusters which be created in some cases, it is necessary to delete some parameters that are considered in clustering but are irrelevant or not very important. After clustering, it is essential for an expert to

khalilian@kiau.ac.ir

carefully interpret the clusters that have been formed. It may be necessary to remove certain parameters that are deemed irrelevant or unimportant in the clustering process. If this is the case, the clustering process may need to be restarted from the beginning. Additionally, data that has been grouped into logical and justifiable clusters can provide valuable insights or may need to be reassigned to different clusters to improve the overall clustering results. Therefore, clustering plays a crucial role in unsupervised learning. In our proposed method, we aim to combine two algorithms with using the Guided Learning Strategy (GLS) [6] to enhance the clustering process. To begin, we analyze the color photo space of our problem. This data has 3 color spectrums of red, green, and blue, and from these three dimensions, we want to extract the centers of the clusters optimally. The criterion for calculating similarity is the Euclidean distance because images have a large number of data [7]. Usually, addition clustering algorithms analyze images with little speed and accuracy [8]. This issue leads to solving this problem by using optimization algorithms. Optimization algorithms are used to improve the results and help them to converge on global optimum. Then it may be useful that get some help from a group of these algorithms called meta-heuristic algorithms [8]. The goal of meta-heuristic algorithms is to find the best solution by exploring all possible solutions in the shortest possible time. Convergence speed, accuracy, and ability to solve problems in feature spaces with high dimensions are desirable features of a good meta-heuristic algorithm. These two features are suitable for improving our results and can be used. However, meta-heuristic algorithms may produce different answers and results in the same conditions. So reaching close answers can increase the stability of the algorithm. Also, basic metaheuristic algorithms fail to fully utilize the valuable information obtained from individuals in previous iterations because they only use the best previous individuals. In practice, each of the preceding individuals may contain a variety of useful information. If such information is fully exploited and used in the subsequent optimization process, the performance of these metaheuristic algorithms will definitely improve significantly. Of course, creating a balance between exploration and exploitation can also increase the stability of the algorithm and show more efficient results [6]. In this way, in the early times of exploring, the problem space has a higher search rate and produces scattered answers. It is possible to increase the exploitation rate at this moment, on the other hand, when the generated answers become more concentrated, it is possible to achieve more answers by increasing the exploration rate. Balancing exploration and exploitation rates can produce better and more stable results. Of course, the increase in time and complexity can be one of the weak points of this approach. The Giza Pyramids Construction [9] is a meta-heuristic algorithm based on generating a new population, ones such as the Grasshopper Optimization Algorithm[10], the above algorithm works and its results show improvement in the output results. This new algorithm draws inspiration from ancient history because ancient civilizations have exhibited the characteristics of good meta-heuristic algorithms [9]. That's why we use the GPC algorithm to increase accuracy [9]. According to the mentioned point from the combination of GLS-GPC, and K-means, we will achieve better results. We will also review the results of DE [14], ABC [16] and GOA [12]. Using entropy criteria and Rand index [9], We examine the clustering quality of each of these methods. The above results show that the combination of GLS-GPC and K-means in two well-known data sets gives better output results than the combination of these two algorithms without using the GLS approach [6]. In the continuation of the article, firstly, some common and widely used clustering methods have been studied, then the activities performed on image clustering and finally the proposed algorithm has been reviewed, and also some evaluators and criteria have used prominent images to evaluate their qualities by some evaluators and criteria. The quality of our proposed algorithm has been compared with previous image clustering methods [12].

2. Active Performed

Several algorithms have been proposed for image clustering, which can be utilized in various ways. These include partition-based methods such as K-means and Fuzzy C-means [2], as well as collective intelligence-based methods like Differential Evolution (DE) [13] or swarm-based such as GOA [10], and GPC [9]. Additionally, combination of GLS [6] and GPC has been employed. The utilization of combined algorithm has resulted in improved clustering quality and enhanced the execution speed of image clustering algorithm.

2.1. K-means Algorithm

The K-means algorithm is a widely recognized clustering technique that is extensively applied in scientific research, making it one of the most popular centroid-based clustering methods [3]. As its name suggests, each of the K clusters is defined by a weighted average, referred to as the cluster center. K-means is commonly employed for tasks such as automatic image tagging and establishing related mappings. In this algorithm, each data point is assigned a membership rank to different classes. The membership function designates either a zero or a one to each data point, indicating whether it belongs to a specific cluster. The primary goal of the K-means clustering algorithm is to minimize a particular objective function.

$$arg_s min \sum_{i=1}^{k} \sum_{x \epsilon S_i} ||x - \mu_i||^2 \qquad (1)$$

In the equation discussed, Si denotes the $i - th$ cluster, μ represents the centroid of the points within that cluster, x signifies the samples that belong to the cluster, k indicates the total number of clusters, and the fuzziness factor is generally set to 2. It is important to highlight that the initial values of the cluster centroids are critical in influencing the quality of the resulting clusters and can greatly affect the final outcomes [3]. Nevertheless, there is no single definitive approach to determining the optimal initial centroid values, which requires multiple iterations. During each iteration, random initial values are generated, and the outcomes are assessed using validation criteria to identify the best solution. The objective function used in K-means clustering seeks to minimize clustering errors by computing the total differences between the cluster centroid and the other data points in that cluster [1,3]. Throughout the K-means clustering process, after allocating each data point to a cluster, the algorithm iteratively updates the cluster centroids by reallocating the data points to their respective clusters. This iterative cycle continues for several rounds until a stopping condition is met, leading to the appropriate updates of the centroids. As a result, the K-means algorithm is classified as a center-based clustering method [3]. Nevertheless, the K-means algorithm has some limitations and weaknesses:

- The clustering results are highly sensitive to the initial values of the cluster centroids.
- It may converge to local optima instead of the global optimum, leading to suboptimal clustering outcomes.
- Selecting the optimal cluster centroid values is a challenging task due to the absence of a precise method.
- The algorithm is prone to the influence of noise, which can adversely affect the clustering results.

2.2. Guided Learning Strategy (GLC)

The Guided Learning Strategy (GLS) is a-state-of-the-art approach which identifies the needs of the optimization algorithms and help them to establish a balance between exploration and exploitation in accordance with the specific requirements of their problems [6]. When the solutions converge, the method increases the exploration variable to enhance the search around potential solutions. Conversely, when the solutions are widely dispersed, it amplifies the exploitation variable to concentrate on the identified points. This approach draws inspiration from Learning Theory. The learning theory is based on the principle that the teacher teaches, and the student not only learns from the teacher but also completes their learning through the environment, prior knowledge, and other classmates. Furthermore, it incorporates periodic evaluations, informed by Ebbinghaus's forgetting curve, to ensure high adaptability in addressing complex problems [4]. Ebbinghaus's Forgetting Curve integrates different memory processes to illustrate the typical decrease in memory retention over time. Memory is generally divided into two categories: short-term memory and long-term memory. A key element in the process of converting short-term memory into long-term memory is the interval between training sessions. Research indicates that ongoing high-intensity training alone does not lead to the formation of long-term memories. Instead, incorporating spaced training is essential for effectively transforming short-term memory into long-term memory. Thus, it is vital to establish suitable feedback intervals [5].

The concepts of exploration and exploitation are similar to two important concepts in learning theory: individual knowledge and mental structures that contribute to knowledge enhancement and problem-solving. In the article,

V_0 represents the current needs of the algorithm, while α is the parameter for guidance. In this algorithm, learning experiences are stored in st, allowing for their reuse to generate new solutions. The algorithm is divided into two stages: the feedback stage and the guidance stage [6].

In the guidance stage, the algorithm decides whether to reduce the number of guidance instances over a longer time period or to increase the number of guidance instances over a shorter time period based on its parameters [6]. In the feedback stage, the algorithm determines which of the stored points in st should be used to improve the problem solutions based on the parameter V_0. This parameter indicates that when the points in St are dispersed, the algorithm is exploring, and when the points are concentrated, the algorithm is exploiting and is close to a solution.

$$V_0 = Std(st) + B \tag{2}$$

Here, Std is the function that calculates the standard deviation of the points in st, and B is used to normalize V_0, keeping the problem within bounds and preventing the solutions from exceeding the limits.

Decision-making regarding the rates of exploration and exploitation is calculated in this section, which is crucial for guiding the algorithm. In formula 3-a, the algorithm exploits the positions of the problem. In this case, the value of $(V_0 > \alpha)$ in situations where$(V_0 \le \alpha)$, the algorithm generates new positions and randomly produces new points in the problem space.

$$X_{new} = \begin{cases} X_{new} + tan(R * \pi) + (ub - lb)/V_0 & , V_0 > \alpha \\ R * (ub - lb) & , V_0 \le \alpha \end{cases} \tag{3}$$

In this equation, X_{best} is the best individual of each iteration, R is a random number between 0 and 1, X_{new} represents the newly generated position ub, lb are upper and lower bounds respectively. Whenever the problem requires exploitation, it uses section 3-a, and whenever it requires exploration, it uses section 3-b to generate new random positions. Consequently, the complexity of the problem increases only slightly due to the simplicity of the calculations with this strategy.

2.3. Giza Pyramids Constructions

Before delving into the optimizer known as GPC, it is essential to briefly discuss the concept of optimization. Optimization involves searching for values among the parameters of a function that either minimize or maximize that function.

All potential values are referred to as possible solutions, while the best of these solutions is termed the optimal solution. Optimization algorithms can address both maximization and minimization problems and have numerous applications, including resource allocation, scheduling, and decision-making. There are various optimization methods available. In many challenging optimization problems, it can be extremely difficult, if not impossible, to identify the best solution through exhaustive searching [15].

A critical aspect of optimization is the time required to obtain a solution, as comprehensive searches tend to be time-consuming and costly. Innovative methods aim to provide satisfactory solutions within a reasonable time frame, although they do not guarantee optimal results. As problems grow in size and complexity, the use of innovative methods has significantly increased [9]. The Giza Pyramids Construction (GPC) is a novel optimization technique based on probabilistic principles, first introduced in 2021 [9]. In the GPC, potential solutions are likened to workers involved in the construction of the Giza Pyramids, who occupy positions in the problem space and gather the best information about themselves and their surroundings to seek a better location within the search space. The next position of these workers is determined based on their current position, the target position, and the positions of all other workers "(4)"

$$d = \frac{v_0^2}{2g(\sin\theta + \mu_k \cos\theta)} \qquad (4)$$

Where d is the value of displacement. g is the gravity of the Earth as mentioned earlier. The value of g is 9.8 and θ is the angle that the ramp makes with the horizon.V_0 is the initial velocity of the stone block and in the algorithm is determined by a uniformly distributed random number in each iteration. With this arrangement, if a worker applies force to a stone block, the stone block starts moving at an initial velocity. As mentioned earlier, the "(5)" determines the amount of stone block displacement relative to its previous position. This equation is used with little change to determine the new position of the worker. For the worker, friction is not considered. Thus, the new position of the worker pushing the stone block is obtained from the following equation,

$$x = \frac{v_0^2}{2g\, sin\, \theta} \qquad (5)$$

In this equation, x is the amount of worker movement. The worker moves upwards with the stone block and simultaneously applies force to the stone block. The goal here is for workers to have better control over the rock block with their small motions. But the worker also has the initial velocity. So for the worker, the friction is not considered. As shown in "(6)", f_k is the kinetic friction force, and since the stone block is at the threshold of displacement, f_k can be obtained from the following equation by the general equation of maximum static friction force,

$$f_k = \mu_k mgcos\theta \qquad (6)$$

Where m is the mass of the stone block, g is the gravity of the earth, θ is the angle that the ramp makes with the horizon, and μ_k is the kinetic friction coefficient. The equations reveal that

1. The first expression of "(4)" generally considers the position of other workers and implements workers' interactions in the project.
2. The second phrase simulates their desire to move on the ramp
3. GPC updates the position of a search agent based on its current position, overall best, and the position of all other search agents
4. In the GPC algorithm, all search agents play a role in determining the next position for each agent, which is viewed as a worker. To maintain a balance between exploration and exploitation, the parameter fk must be random, as indicated in equation "(6)". This approach enhances the number of interactions during operation. The coefficient fk can sometimes decrease the comfort zone in relation to the number of interactions, and vice versa. Ultimately, the best solution identified thus far is regarded by the group as a target that the workers should strive to pursue and improve. One of the advantages of using GPC is its high speed, ease of execution, and a low likelihood of getting trapped in local optima, as it conducts a global search of the problem space. Additionally, it exhibits low time complexity and tends to yield better results compared to other metaheuristic algorithms, especially in complex problems like image clustering. The implementation process is illustrated in "Fig 1."

Algorithm: pseudo-code of the Implementation
Input: Color image from dataset
Output: A clustered image with GLS-GPC, K-means

Step1: Import image and predict the number of clusters by the Gaussian Mixture Model
step2: Generate initial population by GPC
Calculate the values by the similarity evaluation function (Euclidian distance)
Choose the best location and best workers
Initialize new worker with using GLS
Choose the global best and update positions
Step3: Check the final criteria if not, go to Step 2
Output the optimized centers for K-means
Step4: Using these centers in the K-means algorithm for image clustering
Show result

Fig1. Process of Implementation

This algorithm employs random path selection and random answers, which reduces the likelihood of being trapped in local optima compared to other algorithms. As a result,

during the initial stages of the search, when the workers push the huge stones on the ramp friction has a randomness value to increase the probability to create a new position for search. The workers are in various directions, allowing them to explore the entire problem space [9]. Once they identify a promising solution region and gravity become more focused, causing the workers to converge around the solution. This increase in gravity leads to finding optimum locations. Friction inertia force worker to decrease their velocity, which prevents the group of workers through the stones out-of-range of solutions during the convergence.
Below are examples of clustered images that were downloaded from the dataset on the University of Michigan repository.

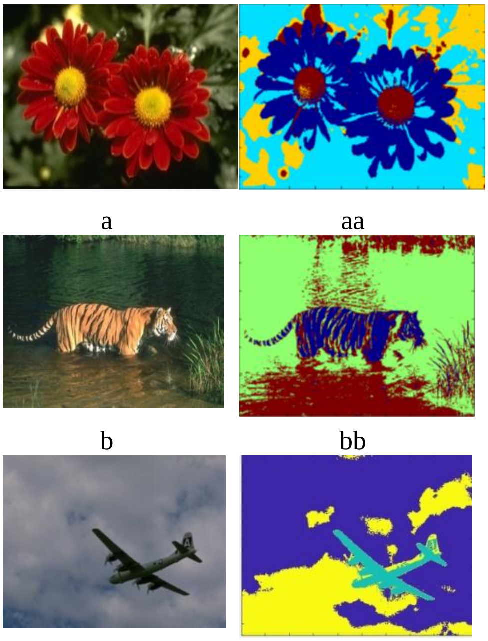

Fig 2: a, b, c are input images, and aa, bb, cc are clustered images with the GLS- GPC and K-means Algorithm.

3. Objective Functions

It is important to select an appropriate objective function for each population-based meta-heuristic algorithm, as these algorithms update candidate solutions based on their quality as the objective function. In our work, we examine three objective functions: The first objective function [3] combines three factors expressing error, intra-cluster distance, and inter-cluster distance in a weighted form:

$$Cf1(xi,Z) = w1dmax\,(Z,xi) + w2(\,zmax\, - dmin\,(Z,xi))\, + \mathrm{w3Je} \tag{7}$$

where Z is the image data set: xi is the i th worker and defined by $xi\, = (m_{i,1}, m_{i,2}, \dots, m_{i,k}, \dots, m_{i,k})$, $m_{i,k}$ is the center of kth cluster of the ith worker. $zmax$ the maximum value of data set (for a s-bit image , $z_{max} = 2^s - 1$), and w_1 , w_2 , and w_3 are user-defined constants. $d_{max}(\mathrm{Z},x_i)$, the intra-cluster distance, is calculated as

$$d_{max} = (Z, x_i) = max_{j=1,\dots,k} \sum_{\forall p_i \in C_j} \frac{d(Z_p, m_{i,k})}{n_{i,k}} \tag{8}$$

where $n_{i,k}$ is the number of pixels in cluster, $C_{i,k}$ and $d(Z_p, m_{i,k})$ is the Euclidean distance between Z_p and $m_{i,k}$. the second component of $Cf1$ is the inter-cluster separation which is known as the minimum average Euclidean distance (d_{min}) between any pairs of clusters. The d_{min} criterion, shown as follows, should be maximized to obtain well-separated clusters:

$$d_{min}(Z, x_i) = min(d(m_i, m_j)) \quad j \neq k \tag{9}$$

And the last component of 1, known as the quantization error (J_e) and defined by Eq .10 is used to express the general quality of a clustering algorithm:

$$J_e = \sum_{k=1}^{K} \sum_{\forall Z_p \in C_k} d(Z_p, m_k)/n_k \tag{10}$$

The second applied objective function: The parameters of $Cf2$ need to be selected empirically for every image, which is not flexible on account of high time consumption. To overcome this problem, $Cf2$ [14] was improved:

$$Cf2(x_i, Z) = \frac{d_{max}(Z,x_i) + J_{e,i}}{d_{min}(Z,x_i)} \tag{11}$$

The proposed objective function: Although $Cf2$ requires no user-defined parameters, the performance of $Cf2$ is almost similar to 1. To tackle the drawbacks of the above-mentioned objective functions, a new fitness function $Cf3$ is improved. Contrary to the others, $Cf3$ considers the mean square error (MSE) and the quantization error in an efficient way. Moreover, $Cf3$ aims to keep the error rate at minor rates while maximizing the d_{max} and minimizing the d_{min} criteria at the same time. The equation of $Cf3$ is as follows:

$$Cf3(x_i, Z) = J_e * \frac{d_{max}(Z, x_i)}{d_{min}(Z, x_i)} * (d_{max}(Z, x_i) + Z_{max} - d_{min}(Z, x_i) + MSE) \tag{12}$$

Where MSE is defined by Eq.12:

$$\mathrm{MSE} = \frac{1}{\mathrm{N}} \sum_{\mathrm{k}=1}^{\mathrm{k}} \sum_{\forall \mathrm{z_p} \in \mathrm{C_k}} \mathrm{d}(\mathrm{z_p}, \mathrm{m_k})^2 \tag{13}$$

Where N is the total number of patterns in a dataset(pixels).

4. Experimental results

To evaluate the proposed image clustering algorithm, we performed a wide range of experiments. The 3 images we used for this purpose are shown in "Fig 2", and are commonly used to evaluate image clustering. We compared our proposed algorithm with several population-based clustering algorithms previously used for image clustering, including Genetic Algorithm [17] combined with K-means clustering, Differential Evolution (DE) combined with K-means clustering, Artificial Bee Colony Algorithm [16] combined with K-means clustering, and Grasshopper Optimization Algorithm (GOA) combined with K-means clustering, as well as conventional clustering (k-mean) algorithm. The population size was the same with other algorithms and the number of iterations for all algorithms were set to 50, respectively, while we used the default values for different parameters. The number of clusters was set to 3. Each algorithm was executed 50 times and in all instances the average of the results of these more than 50 runs[9].

Table 1.
Objective Function Results

image	Obj. func	GA-K	DE-K	ABC-K	GOA-K	GPC-K	**GLS-GOA-K**
Tiger	CF1	89.05	83.87	85.79	80.69	51.762	**50.654**
	CF2	2.32	2.276	2.59	1.927	0.5178	**0.5012**
	CF3	2.06	1.64	2.30	1.25	0.124	**01210**
Red flower	CF1	85.18	87.72	81.73	78.21	53.75	53.751
	CF2	1.36	1.47	1.24	1.11	0.565	0.56511
	CF3	1.04	1.21	0.887	0.72	0.152	0.1516
Plane	CF1	57.77	57.11	57.34	56.87	32.57	**32.327**
	CF2	0.85	0.827	0.82	0.80	0.126	**0.1245**
	CF3	0.199	0.182	0.189	0.18	0.00625	**0.00585**

As mentioned, we consider three different objective functions. Table 1 presents the results for the three target functions for all images and algorithms. As we can see from Table 1, our proposed method leads to better performance values in a number of outputs compared to other algorithms and confirms its higher efficiency.

4.1. Clustering Validity

Clustering validity indices are metrics that indicate clustering quality. In general, they assess two aspects of clustering: 1. Compactness: to the extent possible, samples in a cluster should be similar. 2. Separation: clusters should be separated from each other. In our experiments, we used the Davies–Bouldin (DB) index [3] which is one of the most commonly employed clustering validity indices, The result is shown in Table 2. The DB-index measures the proportion of within-cluster scatter to between-cluster separation. The scatter within the ith cluster is calculated as

$$S_i = \frac{1}{N_i} \sum_{x_j \in c_i} d(x_j, m_i) \tag{14}$$

where N_i is the number of samples of the ith cluster c_i and d(x_j,m_i) is the Euclidean distance between sample x_j and its cluster center m_i. The between-cluster separation is calculated as

$$R_{ij} = \frac{S_i + S_j}{d(m_i, m_j)}, i \neq j \tag{15}$$

Finally, the DB-index is defined as

$$DB = \frac{1}{k} \sum_{k=1}^{K} R_k \tag{16}$$

Where $\mathrm{R}_k = max_{j=1,2,...,\mathrm{K}} R_{ij}$ and $i = 1,2, ..., K$. Table 2. report the average DB-Index results over the 50 runs for all algorithms.

Table 2.
DB-Index results

Image	GA-K	DE-K	ABC-K	GOA-K	GPC-K	**GLS-GPC-k**
Tiger	0.692	0.842	0.926	0.901	0.250	**0.2201**
Red flower	0.9175	0.899	0.858	0.805	0.4098	0.40973
Plane	0.4057	0.398	0.401	0.3924	0.081	**0.07865**

5. Conclusion

The experimental results revealed that the combination of GLS-GPC Algorithm and k-means grows with increasing workers and also the accuracy of the results increases gradually since it does not experience local optimization. Furthermore, our proposed method showed superior performance considering other metaheuristic approaches such as Genetic algorithm, Differential Evolution, Bee Colony, and Grasshopper Optimization Algorithm. These algorithms showed relatively steady performance in terms of the amount of output; Generally, in Cf1, Cf2, Cf3, and DB-index test, our proposed algorithm showed better results in Table 1,2, the same as performance with regard to [12], and [14].

References

[1] Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern recognition letters*, *31*(8), 651-666.

[2] Xu, R., & Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on neural networks*, *16*(3), 645-678.

[3] Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O. P., Tiwari, A., ... & Lin, C. T. (2017). A review of clustering techniques and developments. *Neurocomputing*, *267*, 664-681.

[4] Sikström, S. (2002). Forgetting curves: implications for connectionist models. *Cognitive Psychology*, *45*(1), 95-152.

[5] Hounsell, D., McCune, V., Hounsell, J., & Litjens, J. (2008). The quality of guidance and feedback to students. *Higher Education Research & Development*, *27*(1), 55-67.

[6] Jia, H., & Lu, C. (2024). Guided learning strategy: A novel update mechanism for metaheuristic algorithms design and improvement. *Knowledge-Based Systems*, *286*, 111402.

[7] Abd El Aziz, M., Ewees, A. A., & Hassanien, A. E. (2017). Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. *Expert Systems with Applications*, *83*, 242-256.

[8] Alba, E., & Dorronsoro, B. (2005). The exploration/exploitation tradeoff in dynamic cellular genetic algorithms. *IEEE transactions on evolutionary computation*, *9*(2), 126-142.

[9] Harifi, S., Mohammadzadeh, J., Khalilian, M., & Ebrahimnejad, S. (2021). Giza Pyramids Construction: an ancient-inspired metaheuristic algorithm for optimization. *Evolutionary Intelligence*, *14*(4), 1743-1761.

[10] Meraihi, Y., Gabis, A. B., Mirjalili, S., & Ramdane-Cherif, A. (2021). Grasshopper optimization algorithm: theory, variants, and applications. *Ieee Access*, *9*, 50001-50024.

[11] Mahdavi, M., Fesanghary, M., & Damangir, E. (2007). An improved harmony search algorithm for solving optimization problems. *Applied mathematics and computation*, *188*(2), 1567-1579.

[12] Shahrian, M., & Momtaz, A. K. (2020, December). Multilevel image segmentation using hybrid grasshopper optimization and k-means algorithm. In *2020 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)* (pp. 1-6). IEEE.

[13] Qin, A. K., Huang, V. L., & Suganthan, P. N. (2008). Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE transactions on Evolutionary Computation*, *13*(2), 398-417.

[14] M. Shahrian, and A. K. Momtaz, 1399, Color Image Clustering Using Combination of Grasshopper Optimization Algorithm and k-means Algorithm, The first scientific-research conference on mechanics, electricity, computer and engineering sciences, https://civilica.com /doc/1044248

[15] Meraihi, Y., Gabis, A. B., Mirjalili, S., & Ramdane-Cherif, A. (2021). Grasshopper optimization algorithm: theory, variants, and applications. *Ieee Access*, *9*, 50001-50024.

[16] Karaboga, D., & Ozturk, C. (2011). A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Applied soft computing*, *11*(1), 652-657.

[17] Zeebaree, D. Q., Haron, H., Abdulazeez, A. M., & Zeebaree, S. R. (2017). Combination of K-means clustering with Genetic Algorithm: A review. *International Journal of Applied Engineering Research*, *12*(24), 14238-14245.