

## Segmentation of Covid-19 Affected X-Ray Image using K-means and DPSO Algorithm

**Roopa Kumari**

Department of Computer Science,  
Gurukula Kangri Vishwvidyalya, Haridwar, Uttarakhand, India.  
E-mail: rooparawal@gmail.com

**Neena Gupta**

Department of Computer Science,  
Gurukula Kangri Vishwvidyalya, Haridwar, Uttarakhand, India.  
*Corresponding author:* ngupta@gkv.ac.in

**Narender Kumar**

Department of Computer Science,  
Doon University, Dehradun, Uttarakhand, India.  
E-mail: narenrawal@doonuniversity.ac.in

(Received on May 14, 2021; Accepted on July 31, 2021)

### Abstract

Covid-19, a disease that originated in the Chinese city of Wuhan, has spread across almost the entire globe. Pneumonia, which infects the lungs, is one of the symptoms of this disease. In the past X-ray images were used to segment various diseases such as pneumonia, tuberculosis, or lung cancer. Recent studies showed that Covid-19 affects the lungs. As a result, an X-ray imaging could help to detect and diagnose Covid-19 infection. This study presents a novel hybrid algorithm (CHDPSOK) for segmenting a Covid-19 infected X-ray image. To find Covid-19 contamination in the lungs, we use a segmentation-based approach using K-means and Dynamic PSO algorithm. In the present paper, segmentation of infected regions in the X-ray image uses a cumulative histogram to initialize the population of the PSO algorithm. In a dynamic PSO algorithm, the velocity of the particle changes dynamically which is useful to avoid the local minima. K-means is used to change the position of the particle dynamically for better convergence. To validate the segmentation performance of our algorithm, we used the Kaggle dataset in our experiments. The performance of the proposed method is analyzed both qualitatively and quantitatively. The results explicitly demonstrate the outperformance of the proposed algorithm.

**Keywords-** Covid-19, Chest X-ray, Image segmentation, Particle swarm optimization, K-means.

### 1. Introduction

The COVID-19 has caused millions of deaths and billions of infected cases around the world. Coronavirus 2 (Severe Acute Respiratory Syndrome) causes COVID-19, a highly contagious and harmful viral infection. The virus first originated in the Chinese city of Wuhan in December 2019 (Giannis et al., 2020). After that, it spread all over the world, becoming the world's fifth pandemic disease, following the flu pandemic. Fever, exhaustion, loss of smell, taste, and respiratory symptoms (cough, sore throat, and breathless) are the most common symptoms of COVID-19 disease. Pneumonia and organ failure-like symptoms can occur in certain people, and in the worst-case scenario, they can lead to death. Covid-19 infection cases show mild symptoms in 80% of cases and serve infection in 20%. As of April 8, 2021, there had been more than 132,730,691 cases recorded across 188 countries and territories, with more than 606,000 confirmed deaths.

Since the virus spreads quickly, it's important to catch it early and isolate infected people to prevent infection in the general population (Ai et al., 2020). COVID-19 is detected with a variety of methods. COVID-19 has recently been detected in clinics using Reverse Transcription Polymerase Chain Reaction (RT-PCR) and respiratory specimens. Although RT-PCR is a popular method for detecting COVID-19, it is not without faults. However, RT-PCR takes a long time because it's a tedious, time-taking, and sophisticated process. As a result, infected people transmit disease to other healthy people at a rapid rate. RT-PCR is an expensive procedure that can cost anything from \$15,000 to \$90,000. The poor cannot afford such a high cost, and hospital fees are often very high. Fang et al. (2020) used RT-PCR testing and an initial chest CT to diagnose 81 patients with COVID-19. 30 patients had no initial testing, 51 patients had both initial CT and RT-PCR, and 15 patients had a negative RT-PCR at initial presentation but a positive report after 7 days, while only one patient had a negative chest CT report that became positive after seven days. Many patients may not be diagnosed at an early stage due to the low sensitivity of RT-PCR, and so may not receive adequate care. According to Xie et al. (2020), 3% patients of COVID-19 exhibited a negative RT-PCR. Pneumonia is a serious lung infection that can be fatal in the elderly and those with breathing problems. The diagnosis of pneumonia using chest CT and X-ray is a simple and fast process (Apostolopoulos and Mpesiana, 2020). As a result, diagnosing COVID-19 at an early stage is extremely beneficial. Since it is an automated disease detection system, there is no need for a manual method, and detection is fast. White spots or a hazy appearance can be seen in CT and X-ray images of the lungs. These symptoms indicate an abnormality in the lungs known as "Ground Glass Opacity". GGO is a partial filling of the lungs airspace. Since X-rays are less expensive than CT scans and take less time to acquire, they are being used more often these days. Asif and Wenhui (2020), presented a new method for finding COVID-19 pneumonia patients by chest X-ray images using deep convolutional neural networks, which had a 96 percent accuracy rate. Das et al. (2020), suggested a truncated Inception Net deep learning model find COVID-19 infected people with the help of chest X-rays. The author employed six different datasets for validation and experiment testing: COVID-19 infection, pneumonia infection, tuberculosis infection, and stable chest x-rays.

The suggested approach has a precision of 99.90 percent. For the development of COVID-CxNet, Haghanifar et al. (2020), presented a CheXNet based on the transfer learning model. Based on specific and meaningful characteristics, this model detects COVID pneumonia. Mangal et al. (2020), presented a model COVIDAID (COVID-19 AI detector) based on deep ANN to find the COVID-19 infected people by images of X-ray. The model predicts COVID-19 infection with 90.5 percent accuracy and 100 percent sensitivity. Ahishali et al. (2020), Presents a comparative review on early COVID-19 detection from chest X-ray images. This research employs a compact classifier and a deep learning model.

To analyze and recognize an image, it is necessary to segment it. Image segmentation is the technique of partitioning an image into non-overlapping regions with the aim of each region having the same features (Dhanachandra and Chanu, 2020). It aids in the reduction of an image's complexity so that the image's meaningful features can be extracted. There are several segmentation techniques like conventional techniques (thresholding based, region-based, edge detection based, and clustering-based techniques) and soft computing techniques (genetic algorithm, fuzzy based technique, neural network-based, and particle swarm optimization technique). The recent trend is to use a soft computing approach for image segmentation. This is because it resulted in an optimal solution to the segmentation problem. The soft computing approach gives solutions to complex problems, time-consuming problems, and NP-hard problems

(that has not exact solutions to problems) The PSO was used to optimize the problem related to image reconstruction from a small number of projections (Kumar and Srivastava, 2011).

The clustering technique is the most common and popularly used image segmentation method. Clustering is the method of grouping or partitioning data objects based on their similarity into clusters (Li et al., 2015). Various clustering strategies exist, including density-based clustering, partitioning clustering, hierarchal clustering, fuzzy clustering, and model-based clustering. It consider the number of classes or clusters to be identified using the clustering method, and then use an objective function to find the best partitions or groups. K-means approach (partitioning algorithm) can be considered as the most fundamental and simple methods for clustering. K-means is a very popular method however it has the following disadvantages:

- (i) Choose K manually, the total clusters must be determined in advance.
- (ii) Sensitive towards initial data and maybe falling into local optima.
- (iii) K-means didn't work properly with the global cluster.

Different algorithms have been developed to solve the drawbacks of this clustering technique. Soft membership is a fuzzy c-means (FCM) algorithm that enables one piece of data maybe belongs to more than one clusters. Dunn (1973) invented or suggested this approach, and Bezdek (1981) improved or generalized it. Mashor (2000) introduced a new version of K-means that is moving K-means clustering (MKM). MKM was confident in its ability to reduce the initialization effect, but it was unable to adequately resolve Intracluster variance. Isa et al. (2009) developed three modified variants of K-means: Fuzzy moving K-means, Adaptive moving K-means, and Adaptive Fuzzy moving K-means. These approaches are less susceptible to noise and the clustering value initialization procedure, but it still has issues with dead centers and local minima. Sulaiman and Isa (2010) presented a new method Adaptive Fuzzy K-means. It applies an adaptive and iterative fuzzy to attain the optimal result or better segmentation result. Siddiqui and Isa (2011) introduced an algorithm that is Enhanced moving K-means, which overcome the drawback of the MKM algorithm that is avoiding dead center and local minima problem.

All of the above algorithms partly, but not entirely, resolve the disadvantage of K-means. As a result, further research is needed to overcome the disadvantages of K-mean, especially the local optima problem, since K-means easily collapse into local optima and thus miss the global optima. The PSO algorithm is a population-based search algorithm that was inspired by the social interactions of birds and fish schooling. It's a meta-heuristic algorithm for solving difficult mathematical problems. Omran et al. (2004) introduced an improved PSO by modifying its fitness function. PSO is easy to implement and can perform parallel computations in a short amount of time. PSO has a high probability of finding good global optima, which is an advantage of PSO over K-means. This is the main reason why researchers are interested in combining the two algorithms PSO and K-means to achieve a better or optimal result. Van der Merwe and Engelbrecht (2003) introduced a hybrid approach, in this the individuals of the swarm are seeded by the result of the K-means algorithm. Researchers have recently used three modes to combine or merge PSO and K-means. 1. PSO + K-means 2. K-means + PSO+ K-means 3. K-means + PSO. The PSO + K-means model outperforms the other two modes in terms of experimental evaluation. Since we know that PSO effectively searches all of the space during the initialization of a global search, and then moves to K-means since it converges faster than PSO (Omran et al. 2004). Ahmadyfard and Modares (2008) proposed an algorithm by combining PSO and K-means that is PSOKM to enhance data clustering. Prabha and Visalakshi (2014) developed an efficient

algorithm by combining the merits of PSO and normalization with K-means. Saini and Kaur (2014) proposed a new algorithm that applied a meta-heuristic technique to overcome the drawback of K-means using PSO. Although the PSO algorithm has some flaws, it is still useful. Finding a suitable constant for PSO as an inertia weight is difficult. Because the inertia weight inherits current velocity, the algorithm will become stuck in local optima if the inertia weight is not chosen properly. Finding suitable constants to use as PSO learning variables, on the other hand, can be difficult. Unsuitable factors can cause a particle to waver too often or too often in the local extreme, allowing the algorithm to converge densely or quickly to local minimums. Furthermore, combining PSO and K-means exacerbates all errors or limitations, resulting in a less efficient algorithm. As a result, in recent years, combining PSO and K-means to generate a more effective or optimal algorithm has become popular. Li et al. (2015) introduced DPSOK, a novel image segmentation technique based on Dynamic PSO and K-means clustering. DPSOK calculates the velocity of the particle using dynamic inertia weight and learning parameters. The method may provide good visual effects by equilibrium optimization using dynamic inertia weight and learning factor. DPSOK, a combination of DPSO and K-means, was later applied by Xiaoqiong and Zhang (2020). To improve its optimization capability, the algorithm was subjected to dynamic inertia coefficient and dynamic learning factor. Dhanachandra and Chanu (2020), presented a new segmentation algorithm based on Dynamic PSO and the Fuzzy C-means (FCM) algorithm, as well as a noise reduction mechanism.

CHDPSOK is a new and efficient hybrid algorithm for image segmentation proposed in this paper. The problem is introduced in Section 1, and the fundamentals of particle swarm optimization and K-means are covered in Section 2. The proposed approach is defined in section 3 and the experimental results of the proposed approach are analyzed and discussed in section 4. In the last section, the conclusions of the paper and its future scope are described.

## 2. Related Work

### 2.1 K-means Clustering Algorithm

The Clustering algorithm divides a collection of data or objects into groups called clusters. The K-means technique is the most extensively used clustering algorithm. The K-means clustering algorithm was invented by MacQueen (1967). Clustering problems are solved using the K-means technique, which is an unsupervised learning process. It divides a group of unlabeled data into a K number of clusters in K-means. The clusters were developed based on their similarity. K is predefined, which implies that if  $K=2$ , there will be two clusters, and if  $K=3$ , there will be three clusters, and so on. K-means clustering accomplishes two goals: It determines the K centroid first, and then allocates each data point to the cluster with the closest centroid. Heuristics techniques such as PSO, GA, and Differential evolution can tackle the NP-hard problem of K-means. This problem is also resolved using other techniques. There are various similarity measures used to identify the distance of the closest centroid, such as Manhattan distance and Cosine distance, but Euclidian distance is one of the most commonly used approaches to determine the distance of the closest centroid. After the cluster is established and the centroid is determined, the distance between data points is measured using Euclidian distance, and the data points with the shortest Euclidian distance are assigned to the cluster. Since K-means is an iterative method, it calculates centroid and assigns data points to it until the number of distances is minimized from the centroid of their cluster and each data point (Dhanachandra et al., 2015).

Let us consider an image  $X$  and the image has to be clustered into  $K$  numbers of groups then the working of the K-means (Nearest Centroid Classification) algorithm is:

Step 1: Initialize the value of  $K$  and the centroid randomly.

Step 2: Calculate Euclidian distance for each pixel in an image from pixel to the centroid using the relation given below

$$D_K(i, j) = \|X_K(i, j) - C_K\| \quad (1)$$

where,  $C_K$  is the centroid.

Step 3: Arrange the pixels with in the cluster based on their minimum distances ( $D_1, D_2 \dots D_K$ ).

Step 4: Recalculate the new position of the centroid using the following relation

$$C_K = \frac{1}{|S_K|} \sum_{(i,j) \in S_K} X_K(i, j) \quad (2)$$

where  $X_K(i, j)$  is the value of the pixel labeled with  $k^{th}$  segment and  $S_K$  is the set of indices of the pixel belonging to class  $K \in Y$ . Where  $Y$  is the set of labels of segments.

Step 5: Repeat the step 2 to 4 until any reassignment occurs.

Step 6: Reshape the cluster pixels into the image.

## 2.2 PSO Algorithm

Kennedy and Eberhart (1995) discovered PSO, which is an evolutionary algorithm. It's a population-based meta-heuristic technique focused on simulating the social and cooperative behavior of organisms looking for food. They traveled as a group and worked together to find their food. This method can be used for a variety of optimization problems. A crowd of individuals known as particles flies around the solution space in PSO. Each particle has a spatial location that corresponds to its fitness value. The particle's direction is moving at a constant rate.

Each particle tries to find the best solution for itself, learning from its previous experiences as well as the experiences of other particles. Throughout the movement, each particle maintains its best location based on its fitness value. Personal best (Pbest) position refers to a particle's best position. Global best (Gbest) position is the best of all personal best positions of all particles. With particle acceleration, the velocity and direction of the particle are modified (each individual in a particle). The factors which change velocity/position is current velocity, current position, Pbest position, Gbest position (Kumari et al., 2020).

Let  $n$  is the total number of the particles in a swarm. The positions of each individual in particles are represented by  $X_i^P$  for iteration  $I$  and particle  $P$ . Fitness of particle  $P$  is

$$F_i^P = \text{fitness function} (X_i^P) \quad (3)$$

The personal best value of each individual in a particle  $P$  for iteration  $i$  is calculated based on the fitness value such that

$$F_{i^{Pbest}}^P = \text{minimum} (F_1^P, F_2^P, \dots \dots F_i^P) \quad (4)$$

Here  $i^{Pbest}$  is the iteration number for the best fitness value of the particle. Hence, the personal best position of the particle = best position of the particle  $P = X_{i^{Pbest}}^P$  and the global best position of the particle is

$$Gbest F_{i^{Pbest}}^{P^{Gbest}} = \text{minimum} (F_{i^{Pbest}}^1, F_{i^{Pbest}}^2, \dots \dots F_{i^{Pbest}}^n) \quad (5)$$

where  $P^{Gbest}$  is denoted the global best position of the particles  $P$ . Updated velocity equation is designed as:

$$V_{i+1}^P = w * V_i^P + c1 * r1 * |X_i^P - X_{iPbest}^P| + c2 * r2 * |X_i^P - X_i^{Gbest}| \quad (6)$$

Updated position calculated as:

$$X_{i+1}^P = X_i^P + V_{i+1}^P \quad (7)$$

## 2.3 Cumulative Histogram

### 2.3.1 Histogram

A histogram is a graphical representation of a variable's frequencies values interleaved into ranges, i.e., dividing the whole set of values into intervals and counting how many values fall into each interval. The histogram was invented by Pearson (1895). The continuous ranges were expressed by the histogram's frequency value. These ranges divide a vector into successive and non-overlapping intervals.

### 2.3.2 Image Histogram

In an image, an image histogram is a type of the histogram in which the x-axis represents the gray value of a pixel and the y-axis represents the number of pixels with a given gray level.

Let  $n$  is the number of pixels in an image, which is distributed in  $K$  number of gray levels.

$$n = \sum_{i=0}^{K-1} Xi \quad (8)$$

where  $Xi$  is  $i^{th}$  gray level of the pixel in an image.

### 2.3.3 Normalized Histogram

A histogram can be a normalized by adding the probability density function in normalized histogram the sum of all probability density values will be 1. A normalized histogram

$$Y_i = \frac{Xi}{n} \quad (9)$$

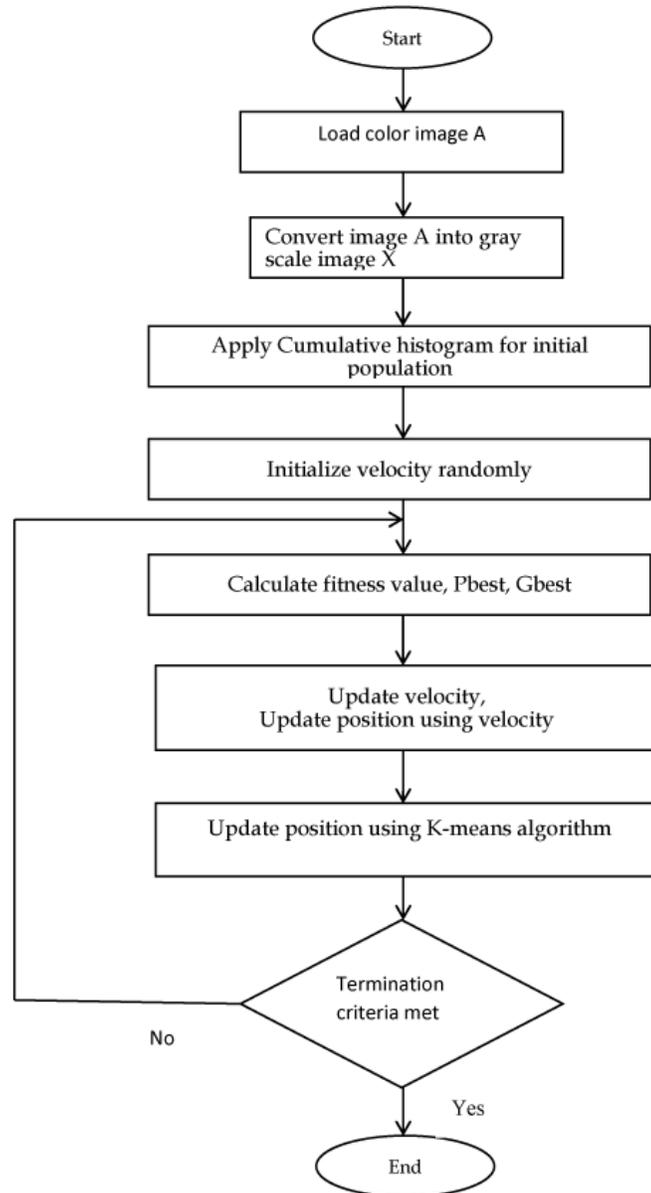
### 2.3.4 Cumulative Histogram

A cumulative histogram is like a graph between x-axis and y-axis in which  $f(x)$  value is taken as the cumulative sum of  $f(x)$  value. It can be defined as

$$Y_i = \sum_{j=0}^{j=i} \frac{Xj}{n} \quad (10)$$

## 3. Proposed Hybrid PSO Algorithm for Image Segmentation

The initial step in the proposed approach is to convert the input image to a gray scale. Each pixel value in a given image varies from 0 to 255 after gray level conversion. The cumulative histogram is used to initialize the particle population, as explained in section 2.3. Figure 1 shows the flowchart of the proposed algorithm (CHDPSOK).



**Figure 1.** Flow chart of proposed algorithm (CHDPSOK).

First, we construct the histogram of the image, then normalize it by using equation (9) and, with the help of equation (10), find its cumulative histogram. Then partition the curve of the cumulative histogram into an "S" number of equal parts and draw a perpendicular from the x-axis to the boundary of these partitions. Here, "S" is the number of segments in the given image. The point on the x-axis where these perpendiculars meet will be the threshold value of the segment. After that, I randomized these segment values by adding a random number from  $-r$  to  $+r$ . These random values will be near the threshold value of the segmentation. Where "r" is a small integer of up to 5. All these steps are described in Algorithm 1.

Algorithm 1. Function to initialize the population of the particle through cumulative histogram.

**random\_generation\_new(img,col,row,pop\_size,culster\_size):**

- 1: Find the histogram using equation (8).
- 2: Normalize the histogram through equation (9).
- 3: Find the cumulative histogram through equation (10).
- 4: Find the threshold value using a cumulative histogram by dividing the x-axis.
- 5: Randomize the value at the boundaries of the segments.

The optimization function can be used to calculate the fitness function. The fitness function will describe the goodness of an individual particle to find an approximate solution. Using the equation below, find the fitness value of each particle:

$$F (\text{Fitness function}) = \frac{\sum_k^N \sum_i^{N_k} |x_{i,k} - c_k|}{\sum_k^N c_k} \quad (11)$$

where  $C_k$  is the centroid of  $k^{th}$  segment and

$$C_k = \frac{\sum_i^{N_k} x_{i,k}}{N_k} \quad (12)$$

where  $N_k$  will be the  $k^{th}$  segment's total pixels and  $x_{i,k}$  will be the intensity value of  $k^{th}$  segments of  $i^{th}$  pixel.

Algorithm 2. Function to calculate fitness function of each swarm of the particle.

**fitness\_calculation(img,col,row,pop\_size,centroid,particle):**

- 1: Initialize array fitness [pop\_size][2]=0
- 2: Repeat step 2 from k=0 to k=pop\_size
- 3: Repeat step 3 to 4 from i=0 to i=row
- 4: Repeat step 4 from j=0 to j=col
- 5: Set l = particle[i][j][k]
- 6: Set fitness[k][1]=fitness[k][1]+ abs((img[i][j]-centroid[l][k]))
- 7: Set fitness[k][0]=k
- 8: Return fitness

Algorithm 2 is used to calculate the particle's fitness value using equations (11) and (12). It measures the average distance of gray value of each pixel from the centroid of the segment.

Algorithm 3. Function to initialize the velocity

**Init\_velocity (row, col, pop\_size)**

- 1: Initialize the array velocity [row][col][pop\_size]:=0
- 2: Repeat step 3 to 5 from i:=0 to i:=row
- 3: Repeat step 4, 5 from j:=0 to j:=col
- 4: Repeat step 5 from k:=0 to k:=pop\_size
- 5: Set velocity[i][j][k] := random integer from 0 to 1
- 6: Return velocity

Algorithm 3 is used to initialize the velocity of particles of the swarm. This algorithm initialized the velocity of each point of a particle using random values from 0 to 1.

Algorithm 4. Function to update Pbest position of the particle

**update\_pbestposition (row, col, pop\_size, pbestposition, cposition, pbestfit, fitness):**

- 1: Repeat step 2 from k=0 to k=pop\_size
- 2: If(pbestfit[k][1]>fitness[k][1]) then goto step 3 else goto step 6
- 3: Repeat step 4 to 5 from i=0 to i=row
- 4: Repeat step 5 from h=0 to h=col
- 5: Set pbestposition[i][h][k] =cposition[i][h][k]
- 6: Return pbestposition

After initialization velocity, algorithm 4 is utilized to determine the best position of the particles from its past history. This best position is called personal best position. The current fitness value of the particle is compared to the fitness value of the particle's personal best position in this algorithm. If the fitness value of the particle's current position is less than the fitness value of the particle's personal best position, the personal best position is updated with the particle's current position.

Algorithm 5. Function to determine Gbest position of the particles

**update\_gbestfit(pop\_size,pbestfit,gb):**

- 1: Repeat step 2 from k=0 to k=pop\_size
- 2: If(pbestfit[k][1]<pbestfit[gb][1]) then goto step 3 else step 4
- 3: Set gb=k
- 4: Return gb

Algorithm 5 is used to find the global best position among all the personal best positions of the particle. It can be calculated by finding the best position using the fitness value of the personal best position.

Algorithm 6. Function to update the velocity of particle

**velocity\_update(col,row,pop\_size,velocity,cposition,pbestposition,gb,w,c1,c2):**

- 1: Set r1=random integer from 0 to 1, r2=random integer from 0 to 1
- 2: Repeat step 3 to 5 from i=0 to i=row
- 3: Repeat step 4, 5 from j=0 to j=col
- 4: Repeat step 5 from k=0 to k=pop\_size
- 5: Set  $velocity[i][j][k] = \text{round}(w*velocity[i][j][k]+c1*r1*(pbestposition[i][j][k]-cposition[i][j][k])+c2*r2*(pbestposition[i][j][gb]-cposition[i][j][k]))$
- 6: Return velocity

Once the velocity of each particle is initialized and after finding the value of Pbest and Gbest Algorithm 6 is used to update the velocity using the equation (6). In this algorithm, velocity is updated using the personal best position, global best values, and current values of the particles of a swarm. Here r1 and r2 are random variables whose values will be set between 0 and 1. The

previous value of velocity is also added to the current velocity by using the weight variable  $w$ .  
Algorithm 7. Function to update current position of the particle through PSO

**update\_cposition\_PSO(row,col,pop\_size,cposition,velocity,culster\_size):**

- 1: Repeat step 2 from  $k=0$  to  $k=pop\_size$
- 2: Repeat step 3 to 4 from  $i=0$  to  $i=row$
- 3: Repeat step 4 from  $h=0$  to  $h=col$
- 4: Set  $cposition[i][h][k] = cposition[i][h][k] + velocity[i][h][k]$
- 5: Return cposition

By using equation (7), Algorithm 7 is utilized to modify the particle's current position. The position is updated using the prior position and current velocity in this algorithm.

Algorithm 8. The K-means algorithm is used to find the current position of a particle.

**kmean\_position(img,col,row,pop\_size,centroid,cromosome,culster\_size):**

- 1: Repeat step 2 from  $k=0$  to  $k=pop\_size$
- 2: Repeat step 3 to 4 from  $i=0$  to  $i=row$
- 3: Repeat step 4 from  $j=0$  to  $j=col$
- 4: Set  $l = \text{particle}[i][j][k]$
- 5: Set distance  $d = (\text{img}[i][j] - \text{centroid}[l][k]) * (\text{img}[i][j] - \text{centroid}[l][k])$
- 6: Repeat step 8 to 10 from  $m=0$  to  $m= \text{cluster\_size}-1$
- 7: If  $(d > (\text{img}[i][j] - \text{centroid}[m][k]) * (\text{img}[i][j] - \text{centroid}[m][k]))$  then
- 8: Set  $d = (\text{img}[i][j] - \text{centroid}[m][k]) * (\text{img}[i][j] - \text{centroid}[m][k])$
- 9: Set  $l = m$
- 10: Return particle

Algorithm 8 is used to determine the current position of the particle by K-means algorithm equation (1) and (2). In this algorithm, distance of each pixel from all centroid of the image is calculated and the pixel is assigned to that segment that has a minimum distance from its centroid. In the Algorithm 9 (Proposed Algorithm CHDPSOK) all the function is called and update the position using the velocity of each particle by equation (7) and update the position of each particle using K-means iteration using the equation (1) and (2). Repeat the entire step until certain criteria are met. Here certain criteria are the maximum number of iteration. Once a criterion is met particle with the global best position will be our final solution and the pixel value of this particle with segmentation is updated to the final segmented image.

## Algorithm 9. Proposed Algorithm CHDPSOK

```

CHDPSOK(input_image, segmented_image):
1: Read input image into matrix img,
2: Set row=number of row(img) and col= number of column(img)
3: Set pop size=10 ,cluster_size= n, gb=0,c1=0.5,c2=0.5,w=0.5
4: Set cposition= call random_generation_new(img,col,row,pop_size,culster_size)
5: Set centroid= call centriod_calculation(img,col,row,pop_size,culster_size,cposition)
6: Set fitness= call fitness_calculation(img,col,row,pop_size,centroid,cposition)
7: Set pbestposition= cposition
8: Set pbestfit= fitness
9: Set pbestfit= call update_pbestfit(pop_size,pbestfit,fitness)
10 :Set:pbestposition= call
update_pbestposition(row,col,pop_size,pbestposition,cposition,pbestfit,fitness)
11: Set gb=update_gbestfit(pop_size,pbestfit,gb)
12: Set velocity= call velocity_init(col,row,pop_size)
13: Set velocity= call
velocity_update(col,row,pop_size,velocity,cposition,pbestposition,gb,w,c1,c2)
14: Set Repeat step 14 to 20 for g=1 to g=generation
15: Set centroid= call centriod_calculation(img,col,row,pop_size,culster_size,cposition)
16: Set fitness= call fitness_calculation(img,col,row,pop_size,centroid,cposition)
17: Set pbestfit= call pbestfit=update_pbestfit(pop_size,pbestfit,fitness)
18: Set pbestposition= call
update_pbestposition(row,col,pop_size,pbestposition,cposition,pbestfit,fitness)
19: Set gb= call update_gbestfit(pop_size,pbestfit,gb),w=g/generation
20: Set velocity=velocity_update(col,row,pop_size,velocity,cposition,pbestposition,gb,w,c1,c2)
21: Set cposition= call kmean_position(img,col,row,pop_size,centroid,cposition,culster_size)
22: Set repeat step 22 to 24 for i =0 to i=row
23: Set repeat step 23 to 24 for j =0 to j=col
24: Set l=cposition[i][j][gb]
25: Set segmented_image [i][j]=round(centroid[l][gb])
26: Return segmented_image

```

#### 4. Result and Discussions

We have collected different images of chest X-rays of covid-19 patients from Kaggle COVID-19 Radiography Database COVID-19 Chest X-ray Database from (COVID-19 data GitHub, 2020). These data sets are provided by the different hospitals for research purposes. The number of generations in the proposed algorithm is taken as 100. Segmentation of these images is done using the proposed algorithm and the result of segmented images is shown in Figure 2 to Figure 4 for different X-ray images.

Following efficiency, factors are used for the evaluation of the algorithm

$$\text{MAE (Mean Absolute Error)} = \frac{1}{ij} \sum |X(i, j) - \bar{X}(i, j)|.$$

Here  $X(i, j)$  and  $\bar{X}(i, j)$  are intensity values of pixel of images with  $i \times j$  dimensions.

$$\text{RMSE (Root Mean Square Error)} = \sqrt{MSE}.$$

$$\text{PSNR (Peak Signal to Noise Ratio)} = \frac{10 \log 255^2}{MSE}$$

$$\text{(Normalized Absolute Error)} = \frac{\sum |X(i,j) - \bar{A}X(i,j)|}{\sum |X(i,j)|}$$

Here  $X(i, j)$  and  $\bar{A}(X i, j)$  are intensity values of pixel of images with I x, J dimensions.

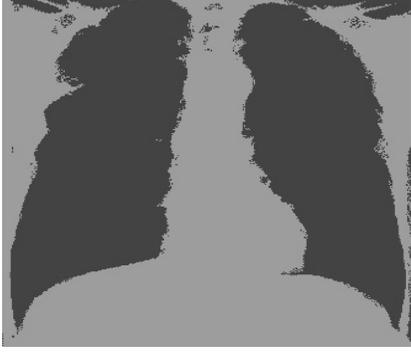
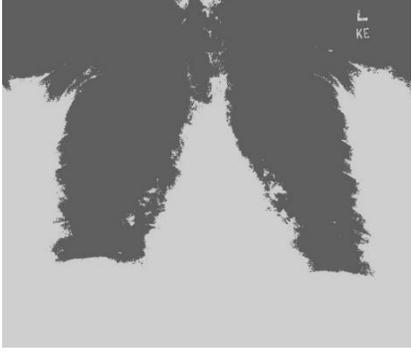
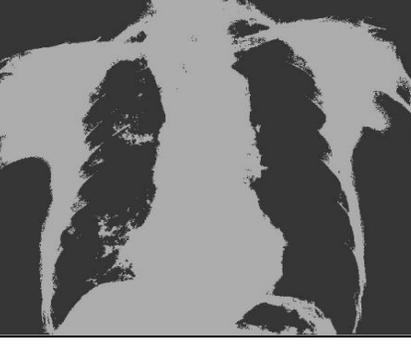
**Table 1.** Comparison of the proposed approach with standard K-means algorithm on segmented images.

Images	Proposed method (CHDPSOK)				K-means				PSO			
	PSNR	MAE	RMSE	NAE	PSNR	MAE	RMSE	NAE	PSNR	MAE	RMSE	NAE
X-ray image (20)	65.327	113.666	9.727	1.027	65.325	113.694	9.728	1.027	65.270	113.946	9.755	1.029
X-ray image (17)	64.521	130.354	10.127	1.154	64.509	130.036	10.133	1.149	63.523	130.410	10.126	1.155
X-ray image (1)	65.333	154.030	9.724	1.029	65.038	154.579	9.869	1.029	65.303	154.708	9.724	1.029
X-ray image (3)	65.525	137.701	9.631	1.114	65.476	138.005	9.655	1.118	63.829	143.321	10.484	1.158

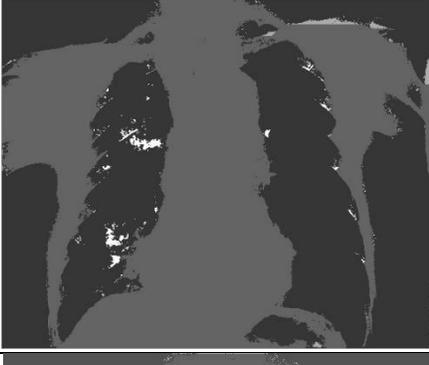
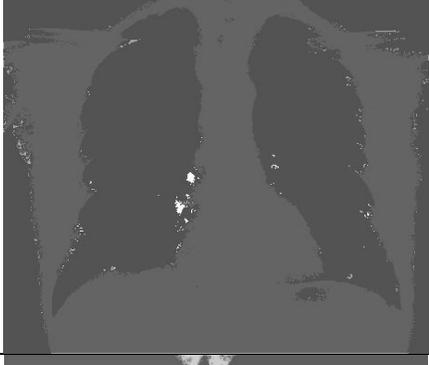
Table 1 to Table 3 is used for comparison of the proposed algorithm (CHDPSOK) with conventional K-means and PSO at different quality measurements (PSNR, MAE, RMSE, NAE). Table 1 shows that values of MAE, RMSE, NAE are less than the K-means algorithm and the PSNR value of the proposed approach is greater than that of standard K-means and PSO technique, which show the optimal result.

**Table 2.** Comparison of the proposed approach with standard K-means algorithm on segmented images with white patches.

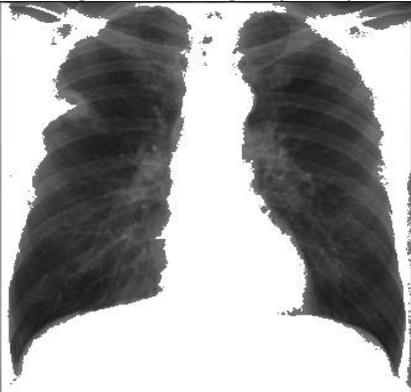
Images	Proposed method (CHDPSOK)				K-means				PSO			
	PSNR	MAE	RMSE	NAE	PSNR	MAE	RMSE	NAE	PSNR	MAE	RMSE	NAE
X-ray image (20)	60.329	103.049	12.493	1.140	60.321	103.036	12.494	1.240	60.087	109.266	12.424	1.199
X-ray image (17)	60.203	116.931	12.560	1.509	60.200	117.698	12.570	1.500	60.201	117.950	12.563	1.508
X-ray image (1)	60.035	122.292	12.674	1.241	59.856	123.305	12.788	1.246	60.030	122.308	12.675	1.241
X-ray image (3)	60.159	120.576	12.095	1.300	60.100	121.141	12.632	1.323	60.009	127.870	12.395	1.307

	<i>Original Images of X-ray</i>	<i>Segmented Images of X-ray</i>
<b>X-ray Image-3</b>		
<b>X-ray Image-20</b>		
<b>X-ray Image-1</b>		
<b>X-ray Image-17</b>		

**Figure 2.** Original and the segmented images of X-ray.

	<i>Original Images of X-ray</i>	<i>Segmented Images of X-ray</i>
<b>X-ray Image-20</b>		
<b>X-ray Image-17</b>		
<b>X-ray Image-3</b>		
<b>X-ray Image-1</b>		

**Figure 3.** Original and segmented images with white patches.

	<i>Original Images of X-ray</i>	<i>Segmented Images of X-ray</i>
<b>X-ray Image -20</b>		
<b>X-ray Image-17</b>		
<b>X-ray Image-3</b>		
<b>X-ray Image-1</b>		

**Figure 4.** Original and the segmented lung images.

**Table 3.** Comparison of the proposed approach with standard k-means algorithm on lung segmented images.

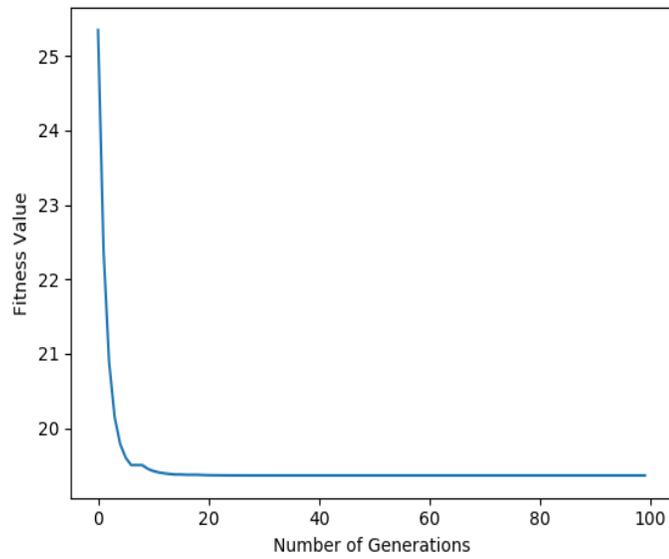
Images	Proposed method (CHDPSOK)				K-means				PSO			
	PSNR	MAE	RMSE	NAE	PSNR	MAE	RMSE	NAE	PSNR	MAE	RMSE	NAE
X-ray image (20)	61.320	103.035	12.494	1.239	60.320	103.055	12.495	1.240	61.085	109.295	12.025	1.250
X-ray image (17)	60.182	118.610	12.501	1.517	60.175	118.678	12.585	1.521	60.102	118.834	12.576	1.528
X-ray image (1)	60.035	123.071	12.674	1.251	59.856	123.181	12.788	1.257	60.030	123.141	12.775	1.251
X-ray image (3)	60.142	120.851	12.606	1.310	60.079	121.477	12.646	1.329	60.209	127.985	12.395	1.316

Table 2 described that the PSNR value of the proposed method (CHDPSOK) is greater than that of the K-means and PSO algorithm and the error rates such that the value of MAE, RMSE, and NAE are also less in comparison to traditional K-means and PSO algorithm.

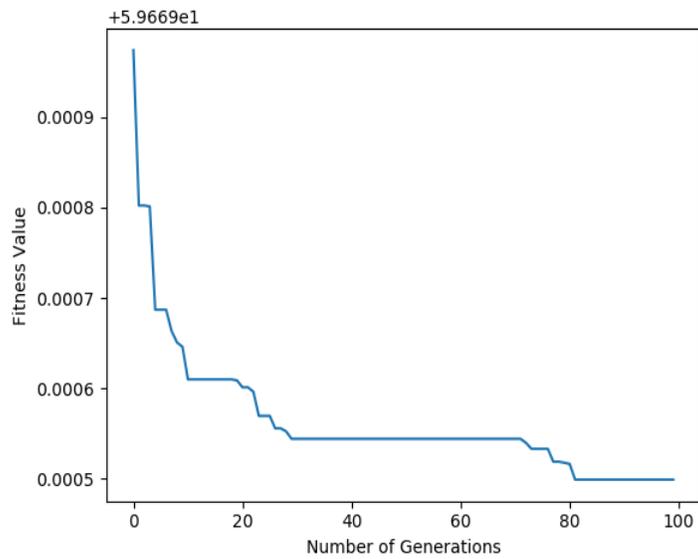
The table shows the quantitative result of Figure 3 images with white patches after the segmentation of the X-ray images. The results clearly show the efficiency of the proposed algorithm. Table 3 also shows the PSNR values of the proposed algorithm are more than conventional K-means and PSO algorithm and as we know PSNR is inversely proportional to MAE so, If the PSNR value is more at the same time MAE value is less and Table 3 results fulfill this condition also. The result shows that the MAE, NAE, and RMSE value is less in the proposed algorithm and the PSNR value is more. These tables show the quantitative result of Figure 4 images that segment the lung portion from the image.

Figures 2 to Figure 4 show the qualitative result of the proposed method for different X-ray images. Figure 2 shows the segmentation of the given X-ray image. White patches are detected from these images using the proposed algorithm in Figure 3. In Figure 4 chest part of the X-ray is segmented using the proposed algorithm.

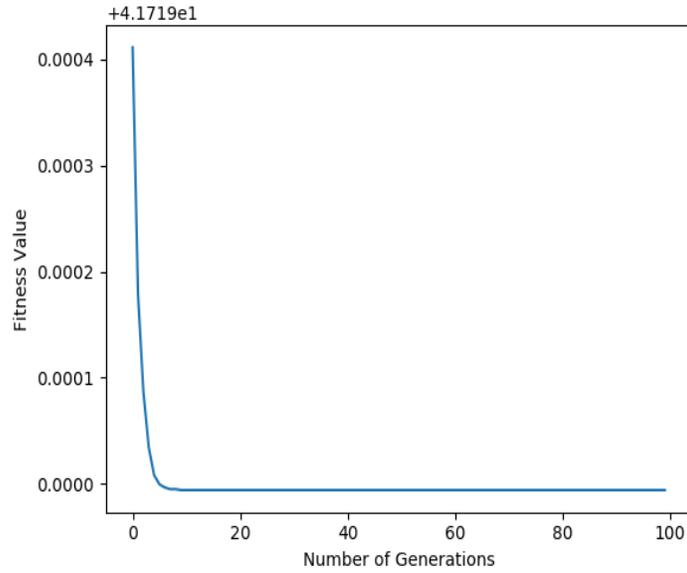
The convergence of the proposed method with various images is depicted in Figures 5 to 8. The x-axis in these graphs reflects the number of generations, while the y-axis represents the suggested algorithm's fitness value. The graphs clearly illustrate that after a certain number of iterations (generations), the fitness value becomes constant, indicating that the algorithm has converged.



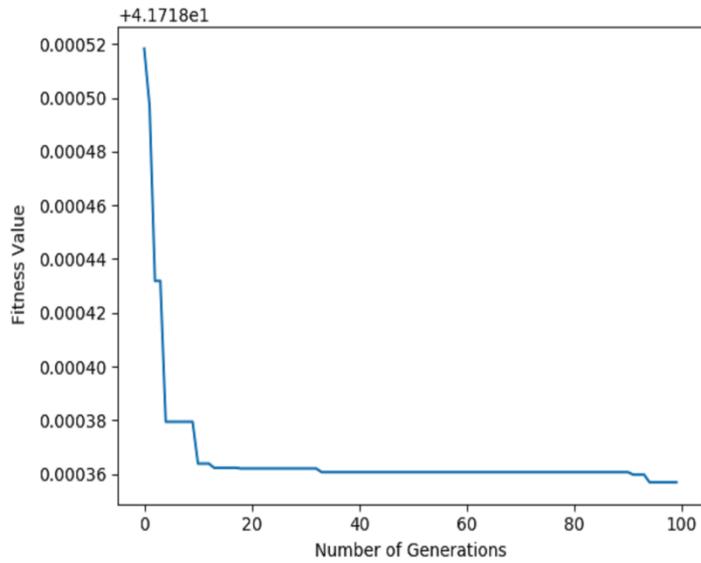
**Figure 5.** Convergence graph for image-20.



**Figure 6.** Convergence graph for image-17.



**Figure 7.** Convergence graph for image-3.



**Figure 8.** Convergence graph for image-1.

## 5. Conclusions

The main aim of this research is to propose a novel CHDPSOK image segmentation technique that is hybrid. In this approach, the cumulated histogram is used to initialize the population of particles for segmentation. The PSO algorithm is then used to segment the data, and K-means is

then utilized to increase convergence by measuring and updating the particle's location. To test the accuracy of the proposed method, different chest X-ray images of covid-19 patients from the Kaggle dataset were employed. After comparing the suggested algorithm's (CHDPSOK) results to those of existing algorithms (PSO and K-means), PSNR (Peak Signal to Noise Ratio), MAE (Mean Absolute Error), RMSE (Root Mean Square Error), and NAE (Normalized Absolute Error) were found to be superior with the proposed algorithm (CHDPSOK). In the future, alternative heuristic methods could be used to implement this approach for colored images.

#### Conflict of Interest

The authors confirm that there is no conflict of interest to declare for this publication.

#### Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors would like to thank the editor and anonymous reviewers for their comments that help improve the quality of this work.

#### References

- Ahishali, M., Degerli, A., Yamac, M., Kiranyaz, S., Chowdhury, M.E., Hameed, K., Hamid, T., & Gabbouj, M. (2020). A comparative study on early detection of covid-19 from chest x-ray images. *arXiv preprint arXiv:2006.05332*.
- Ahmadyfard, A., & Modares, H. (2008). Combining PSO and K-means to enhance data clustering. In *2008 International Symposium on Telecommunications* (pp. 688-691). IEEE. Tehran, Iran.
- Ai, T., Yang, Z., Hou, H., Zhan, C., Chen, C., Lv, W., & Xia, L. (2020). Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. *Radiology*, *296*(2), E32-E40.
- Apostolopoulos, I.D., & Mpesiana, T.A. (2020). Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine*, *43*(2), 635-640.
- Asif, S., & Wenhui, Y. (2020). Automatic detection of COVID-19 using X-ray images with deep convolutional neural networks and machine learning. *medRxiv*.
- Bezdek, J.C. (1981). Objective function clustering. In *Pattern Recognition with Fuzzy Objective Function Algorithms*. Springer, Boston, pp. 43-94.
- COVID-19 data github (2020). Retrieved from: <https://github.com/ieee8023/covid-chestxray-dataset/tree/master/images>.
- Das, D., Santosh, K.C., & Pal, U. (2020). Truncated inception net: COVID-19 outbreak screening using chest X-rays. *Physical and Engineering Sciences in Medicine*, *43*(3), 915-925.
- Dhanachandra, N., & Chanu, Y.J. (2020). An image segmentation approach based on fuzzy c-means and dynamic particle swarm optimization algorithm. *Multimedia Tools and Applications*, *79*, 18839-18858.
- Dhanachandra, N., Mangle, K., & Chanu, Y.J. (2015). Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science*, *54*, 764-771. DOI: 10.1016/j.procs.2015.06.090.
- Dunn, J.C. (1973). A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, *3*(3), 32-57. DOI: 10.1080/01969727308546046.

- Fang, Y., Zhang, H., Xie, J., Lin, M., Ying, L., Pang, P., & Ji, W. (2020). Sensitivity of chest CT for COVID-19: comparison to RT-PCR. *Radiology*, 296(2), E115-E117.
- Giannis, D., Ziogas, I.A., & Gianni, P. (2020). Coagulation disorders in coronavirus infected patients: COVID-19, SARS-CoV-1, MERS-CoV and lessons from the past. *Journal of Clinical Virology*, 127, 104362.
- Haghanifar, A., Majdabadi, M.M., Choi, Y., Deivalakshmi, S., & Ko, S. (2020). Covid-cxnet: Detecting covid-19 in frontal chest x-ray images using deep learning. *arXiv preprint arXiv:2006.13807*.
- Isa, N.A.M., Salamah, S.A., & Ngah, U.K. (2009). Adaptive fuzzy moving K-means clustering algorithm for image segmentation. *IEEE Transactions on Consumer Electronics*, 55(4), 2145-2153.
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95-International Conference on Neural Networks* (Vol. 4, pp. 1942-1948). IEEE. Perth, WA, Australia.
- Kumar, N., & Srivastava, T. (2011). A PSO based approach to image reconstruction from projections. *International Journal of Tomography & Statistics*, 17(S11), 29-38.
- Kumari, R., Gupta, N., & Kumar, N. (2020). Cumulative histogram based dynamic particle swarm optimization algorithm for image segmentation. *Indian Journal of Computer Science and Engineering*, 11(5), 557-567.
- Li, H., He, H., & Wen, Y. (2015). Dynamic particle swarm optimization and K-means clustering algorithm for image segmentation. *Optik*, 126(24), 4817-4822.
- MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability* (Vol. 1, No. 14, pp. 281-297). University of California, Berkeley.
- Mangal, A., Kalia, S., Rajgopal, H., Rangarajan, K., Namboodiri, V., Banerjee, S., & Arora, C. (2020). CovidAID: COVID-19 detection using chest x-ray. *arXiv preprint arXiv:2004.09803*.
- Mashor, M.Y. (2000). Hybrid training algorithm for RBF network. *International Journal of the Computer, the Internet and Management*, 8(2), 50-65.
- Omran, M.G., Engelbrecht, A.P., & Salman, A. (2004). Image classification using particle swarm optimization. In Tan, K.C. (ed) *Recent Advances in Simulated Evolution and Learning*. World Scientific, Singapore, pp. 347-365.
- Pearson, K. (1895). X. Contributions to the mathematical theory of evolution.—II. Skew variation in homogeneous material. *Philosophical Transactions of the Royal Society of London A*, 186, 343-414.
- Prabha, K.A., & Visalakshi, N.K. (2014). Improved particle swarm optimization based k-means clustering. In *2014 International Conference on Intelligent Computing Applications* (pp. 59-63). IEEE. Coimbatore, India.
- Saini, G., & Kaur, H. (2014). A novel approach towards K-mean clustering algorithm with PSO. *International Journal of Computer Science and Information Technologies*, 5, 5978-5986.
- Siddiqui, F.U., & Isa, N.A.M. (2011). Enhanced moving K-means (EMKM) algorithm for image segmentation. *IEEE Transactions on Consumer Electronics*, 57(2), 833-841.
- Sulaiman, S.N., & Isa, N.A.M. (2010). Adaptive fuzzy-K-means clustering algorithm for image segmentation. *IEEE Transactions on Consumer Electronics*, 56(4), 2661-2668.
- Van der Merwe, D.W., & Engelbrecht, A.P. (2003). Data clustering using particle swarm optimization. In *The 2003 Congress on Evolutionary Computation-CEC 2003*, (Vol. 1, pp. 215-220). IEEE. Canberra, ACT, Australia.

Xiaoqiong, W., & Zhang, Y.E. (2020). Image segmentation algorithm based on dynamic particle swarm optimization and K-means clustering. *International Journal of Computers and Applications*, 42(7), 649-654.

Xie, X., Zhong, Z., Zhao, W., Zheng, C., Wang, F., & Liu, J. (2020). Chest CT for typical Coronavirus disease 2019 (COVID-19) pneumonia: relationship to negative RT-PCR testing. *Radiology*, 296(2), E41-E45.



Original content of this work is copyright © International Journal of Mathematical, Engineering and Management Sciences. Uses under the Creative Commons Attribution 4.0 International (CC BY 4.0) license at <https://creativecommons.org/licenses/by/4.0/>