ISSN: 0976-9102 (ONLINE)
DOI: 10.21917/ijivp.2018.0256

IMAGE SEGMENTATION ANALYSIS BASED ON K-MEANS PSO BY USING THREE DISTANCE MEASURES

Elindra Ambar Pambudi[1], Pulung Nurtantio Andono[2] and Ricardus Anggi Pramunendar[3]

[1]Informatic Engineering, Universitas Muhammadiyah Purwokerto and Universitas Dian Nuswantoro, Indonesia
[2,3]Informatic Engineering, Universitas Dian Nuswantoro, Indonesia

Abstract

The image segmentation is a technique of image processing which divides image into segments. The many proposed image segmentation techniques, k-Means clustering has been one of the basic image segmentation techniques. The advantages of k-Means are easy calculation, the number of small iteration, and one of the most commonly used clustering algorithm. but, The main problem in this algorithm is sensitive to selection initial cluster center. In this research, we present two approaches method which are used to execute image. It is PSO and k-Means. k-Means integrated with Particle Swarm Optimization (PSO) to improve the accuracy. The purpose of this research to find the effect of PSO towards k-Means in order to get the best selection initial cluster center. This research has been implemented using matlab and taking image dataset from weizzmann institute. The Result of our experiment, we have different result RMSE of k-Means PSO. Euclidean has less RMSE value than Manhattan. The difference RMSE between Euclidean PSO and Manhattan PSO only four point. but if we compare by processing time we have significant difference.

Keywords:

Vision Computing, Image Processing, Segmentation, Swarm Intelligence, Computer Science

1. INTRODUCTION

Image processing is the process and the analysis image using many visual perceptions. One of the process of image processing for recognition of objects is image segmentation. Image segmentation is an important part of analysis digital image. For analysis image, many cases can be appointed as research materials, example feature extraction, segmentation, image classification, object recognition. Image segmentation is the step of image processing to divide objects from the background. The image is parted to be regions and we know the discontinuities as the boundaries between the regions. Image segmentation is indispensable to doing next step into digital image processing like feature extraction, classification, and detection. The background usage color image segmentation was strengthened by paper Nikita Sharma, et al [1], She states the improvement of computer processing capabilities and the increased application of color image, the color image segmentation is more and more concerned by the researchers.

In this case, fundamental technique of image segmentation is clustering. We have priority choosing *k*-Means as clustering method of Image segmentation. The reason of choosing *k*-Means as method in this research based paper Komarasamy et al. [2] is that *k*-Means is the famous non-hierarchical clustering algorithm and it has good convergence. Pritesh Vora et al. [3] states that *k*-Means clustering is simple, flexible, can be understood and implemented easily. After reading the description of the previous paper Komarasamy and Vora that the advantage of *k*-Means is can be understood, easy to implement, and flexible. Therefore, we choose *k*-Means as an image segmentation algorithm.

The first drawback of *k*-Means is the cluster result is sensitive for selection initial cluster center. It is further strengthened by paper Panchal et al. [4] where the election initial cluster center and search local solution can influence on the performance methods and the partition. The second disadvantage, it is sensitive for choice method distance measure. In several paper, the distance measure algorithm is more frequent using Euclidean distance. Meanwhile, in paper Kouser et al. [5], He researched about comparison distance measure algorithm and he concluded the result Euclidean distance is not always good when used in *k*-Means algorithm.

In order to improve performance of image segmentation especially clustering technique we use Particle Swarm Optimization for optimizing centroid cluster of *k*-Means. The usage of Particle Swarm Optimization by many researchers can solve the problem of clustering. The literature [6] proposed the usage PSO and *k*-Means clustering algorithm to retrieve images accurately from large image database similar. The literature [7] an approach where we combine a clustering technique and a stochastic technique to select effective features from the high dimensional breast cancer data set in quick time.

Therefore, Based on the analysis of the above algorithm. In this paper we present and clustering optimization of *k*-means algorithm implemented in image segmentation base on the improved Particle Swarm Optimization algorithm. We propose using particle Swarm Optimization to reduce the first disadvantage of *k*-Means to selection of the initial cluster centroids. PSO is used to generate good initial cluster centroid. The result of our research indicate that the PSO + *k*-Means algorithm can generate the best results minimum RMSE in comparison with the *k*-Means algorithm. We also attempting to compare distance measure algorithm Manhattan, Euclidean, and Canberra.

The Most common from exist paper is only used one distance measure Euclidean distance or city block/Manhattan. Researches about image segmentations by using *k*-Means PSO are commonly used, but if we use three distance measure for *k*-Means PSO, it is going to be different thing. Many papers *k*-Means PSO rarely discussing about implementation three distance measure. In this paper, we use three distance measure which is Euclidean, Manhattan and Canberra distance. Three distance measures implemented in *k*-Means PSO is novelty of this paper.

2. RELATED WORK

There are several researchers who have been carried out to enhance the accuracy of *k*-Means Algorithm with Particle Swarm Algorithm. PSO gives better initial seed for *k*-Means clustering to result better cluster.

Cui and Potok [8] proposed Method PSO+ *k*-Means, the author presented a hybrid document clustering by using the number document 204 until 800, and result of the paper is PSO method capable to enhance performance *k*-means in global searching.

Doreswamy and Salma [6] proposed an approach where they combine a clustering and stochastic technique to select effective features from the high dimensional breast cancer data set in quick. In their paper, they used PSO to optimize Fast *k*-Means algorithm.

Wahi [2] proposed Method *k*-Means and PSO, the author presented about clustering dataset text from UCI machine learning repository. In his research contains comparison clustering *k*-Means stand alone, PSO stand alone and combining *k*-Means PSO.

The all researches already mentioned above is only used Euclidean or Manhattan as distance measure. The researchers did not use comparison three distance measure.

3. K-MEANS

Many clustering algorithms are used to solve image segmentation problem, one of them is *k*-Means. *k*-Means clustering is the one of nonhierarchical data clustering method to divide the data in cluster, thus the data have same characteristic grouped into same cluster and the data having different characteristic is grouped into other cluster. The process *k*-Means is done by minimizing sum squares of distance between the corresponding cluster centroid and data.

- Initializing the number K to be formed
- Determine random cluster centroid $z_1, z_2, ..., z_k$ from the n data vector ($x_1, x_2, x_3, ..., x_n$)
- Calculate the distance of each object to the centroids with Canberra, Euclidean and Manhattan.
- Grouping objects based on minimum distance (find the closest centroid)
- Determine new cluster center $m_1, m_2, ..., m_k$ as follows:

$$*x_p = \frac{1}{n_j} \sum_{n_j = C_i} z_p \tag{1}$$

n_j is the number elements belong to c_i

- The terminate condition is reached, If new cluster no change convergence has occured or the number iterations is less than preassigned maximum value. Otherwise return to step 2.

3.1 EUCLIDEAN

The euclidean distance is commonly used to calculate vector similarity. the euclidean formula is root square from difference 2 vector. It given as follows:

$$dist\left(z_i, x_p\right) = \sqrt{\sum_{i=1}^{k} \left(z_i - x_j\right)^2} \tag{2}$$

3.2 CANBERRA

The Canberra distance is a metric used for data scattered around the origin. The canberra distance d between 2 vectors x and y is given as follows:

$$dist\left(z_i, x_p\right) = \sum_{i=1}^{k} \frac{\left|z_i - x_j\right|}{\left|z_i\right| + \left|x_j\right|} \tag{3}$$

3.3 MANHATTAN

Manhattan distance also known as city block distance, The Manhattan distance is the distance between two points measured along axes at right angles. The formula of manhattan distance are defined as follows:

$$dist\left(z_i, x_p\right) = \sum_{i=1}^{k} \left|z_i - x_j\right| \tag{4}$$

4. PARTICLE SWARM OPTIMIZATION

PSO is based on Swam Intelligence. This method was originally developed first by Russel C. Eberhart and James Kennedy in 1995 [9]. As described Eberhart and Kennedy [9], PSO is theory a population based stochastic search process, modeled after the social bahaviour of "bird flock". In this context, which swarm refers to a population and Individual refers to a particle. Each particle in PSO relates with velocity.

First step of PSO is Initialize each particle with random position and velocity. Second step, for each iteration Evaluate fitness value for each particle. Third Step, calculate (Pbest and Gbest), There are two steps that influencing renewal process of velocity, the first step is updating best position for each particle (Pbest) and the second step is updating best position for all population (Gbest). PSO has two primary operators: Velocity and Position update. The velocity and position of each particle was updated by Pbest and Gbest of each generation. In the method of PSO, due process initialization velocity (Vel_{id}) = 0. For next position we calculate new velocity and best previous position. The stop Condition if the criteria is reached. If the criteria aren't reached therefore back to evaluate fitness value.

The Formula of PSO according to Eberhart and Kennedy [9]:

$$vel_{id} = w * vel_{id} + c_1 * rand_1 *(p_{id} - x_{id}) + c_2 * rand_2 \ (p_{gd}-x_{id}) \tag{5}$$

$$x_{id} = x_{id} + vel_{id} \tag{6}$$

where w = inertia, vel_{id} is velocity particle-id, x_{id} is position particle-*id*, c_1/c_2 is learning rate, p_{id} is best position for each particle (pbest) and p_{gd} for all particle (gbest).

5. PROPOSED METHOD

- **Input Image and Color mapping**: The input image is digital color image RGB 24 bit with extension file *.bmp, *.jpg. Reading color pixel needed for knowing value of RGB pixel image, and then mapping the image pixels to the RGB color space.

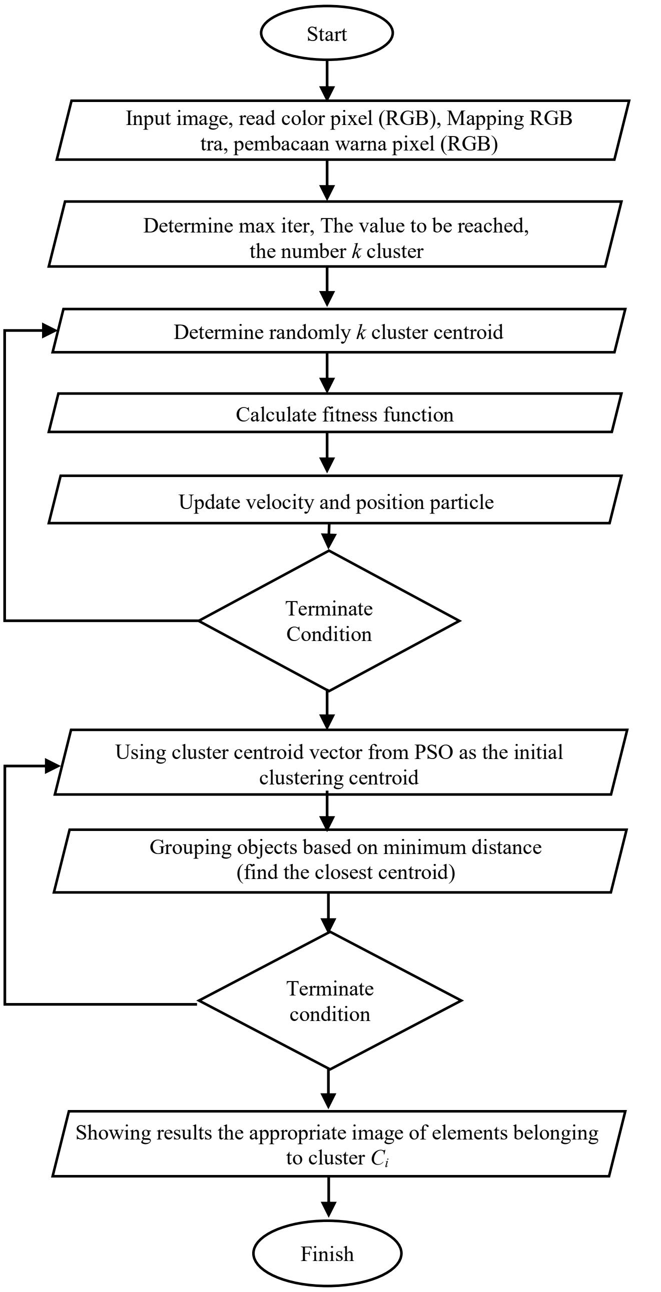

Start
Input image, read color pixel (RGB), Mapping RGB tra, pembacaan warna pixel (RGB)
Determine max iter, The value to be reached, the number *k* cluster
Determine randomly *k* cluster centroid
Calculate fitness function
Update velocity and position particle
Terminate Condition
Using cluster centroid vector from PSO as the initial clustering centroid
Grouping objects based on minimum distance (find the closest centroid)
Terminate condition
Showing results the appropriate image of elements belonging to cluster C_i
Finish

Fig.1. Flowchart *k*-Means + PSO

- **Determine Max iteration**: The value to be reached, and the number *k* cluster. The number *k* cluster are taken from questioner. The number *k* cluster is essential, we use different number *k* cluster for each image according questioner. Because, each of the image has different color.
- **Determine randomly *k* cluster centroid**: In this step, determine the position start point *k* cluster center randomly.
- **Calculate Fitness Function**: PSO clustering in this research, cluster centroid represents particle. Clustering the data based closer distance to cluster center. Calculate the distance measure with Euclidean Eq.(2), Manhattan Eq.(4), and Canberra Eq.(3). For each particle calculate fitness value according to Fitness value. Fitness function in this research is minimum value SSE (Sum Square Error). To get p_{id} and p_{gd} following Eq.(7).

$$p_i(t+1)=\begin{cases} p_i(t) & f\left(x_i(t+1)\le f\left(x_i(t)\right)\right) \\ x_i(t+1) & f\left(x_i(t+1)>f\left(x_i(t)\right)\right)\end{cases} \tag{7}$$

where, the symbol *f* denotes the fitness function, $P_i(t)$ stands for the best fitness values and the coordination and finally the value was calculated. *t* denotes the generation step.

P_1(Pbest 1) = (49.75,59.5,7.75), (9, 17, 2), (125,121,111);

P_2(Pbest 2) = (17,26.5,1), (151.3, 144, 136.3), (57, 66, 11);

Fitness 1st particle: 2,798142781

Fitness 2nd particle: 6, 701695375

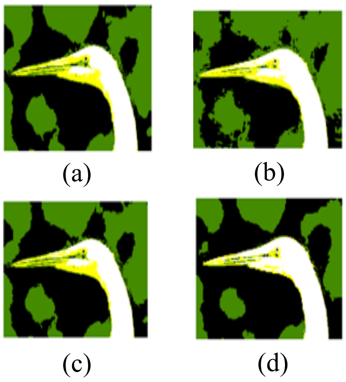
where, p_{gd} (Gbest) is best solution value for all particles. Gbest result is (49.75,59.5,7.75), (9, 17, 2), (125,121,111).

- **Update Velocity Eq.(5) and Position Particle Eq.(6)**: Initialize PSO parameter, Generate velocity (Vel_0) of each particle. Compare each particle fitness value Eq.(7) with particle's local best solution, the better candidate solution encountered by particle *i*, if current value is better than Pbest, set Pbest equal to the current particle. Choose the best value p_{id} (Pbest) from all particles as p_{gd} (Gbest) before entry into process calculate velocity and position.

  After found Pbest and Gbest value, the next step is updating velocity with formula Eq.(5) and then update position cluster center with formula Eq.(6). Initialize Vel_{id} value is zero, the value of C_1 and C_2 is 1.
- Using cluster centroid vector from PSO as the initial clustering centroid. *k*-Means will inherit cluster centroid vector from PSO as the initial clustering centroids. In this step, we just determine max iteration and terminate condition. Assign data vector to the closest cluster centroids. Recalculate cluster centroid with Eq.(1).
- Grouping objects based on minimum distance (find the closest centroid)

6. IMPLEMENTATION RESULT

For our research, we use five datasets image. The proposed technique in this research is implemented on different images and results different RMSE and Processing time. In this section we present the result image segmentation with *k*-Means and *k*-Means PSO on different Distance Measure.



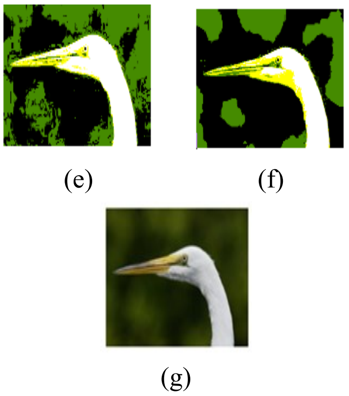


Fig.2. Image Egret_face.png *k*-Means → (a) Euclidean, (b) Canberra, (c) Manhattan, *k*-Means + PSO → (d) Euclidean, (e) Canberra, (f) Manhattan, (g) Original image

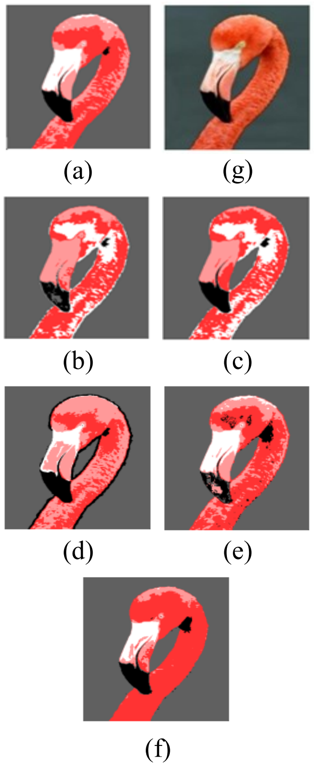
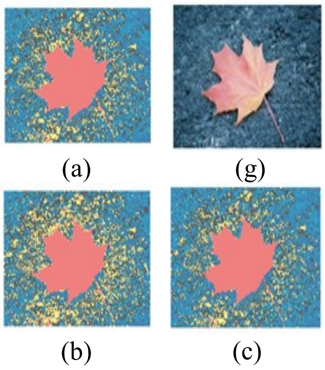


Fig.3. Image Img_7359_copia.png *k*-Means → (a) Euclidean, (b) Canberra, (c) Manhattan, *k*-Means PSO → (d) Euclidean, (e) Canberra, (f) Manhattan, (g) Original Image



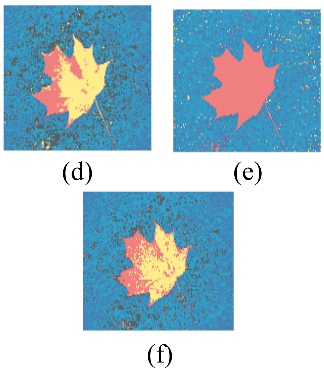


Fig.4. Image Leafpav.png *k*-Means → (a) Euclidean, (b) Canberra, (c) Manhattan, *k*-Means PSO → (d) Euclidean, (e) Canberra, (f) Manhattan, (g) Original image

From the results of testing that has been carried out with *k*-Means and *k*-Means PSO with different distance measure Euclidean, Canberra, Manhattan. So, we can evaluate performance comparison between *k*-Means and *k*-Means PSO based RMSE Value using Eq.(3) and Processing Time.

Table.1. Performance Comparative RMSE of *k*-Means and Kmeans+PSO on five different image datasets

Image Test	**Distance Method**	**RMSE Value**					
		***k*-Means**			***k*-Means PSO**		
		1st Trial	**2nd trial**	**3rd trial**	**1st Trial**	**2nd trial**	**3rd trial**
egret_face	Euclidean	25,22	25,20	25,22	22,47	27,13	24,48
	Canberra	26,71	26,75	27,01	25,48	25,07	23,43
	Manhattan	25,26	25,36	25,26	25,43	21,80	23,31
img_7359_copia	Euclidean	24,10	21,93	21,93	23,37	21,30	21,06
	Canberra	24,66	24,68	24,85	23,57	23,23	26,29
	Manhattan	28,08	31,49	22,75	24,46	22,49	22,08
leafpav	Euclidean	22,97	14,84	14,84	14,09	18,30	19,02
	Canberra	41,91	49,29	40,93	31,25	48,12	25,37
	Manhattan	62,45	21,82	31,75	15,61	34,68	28,81
redberry_rb03	Euclidean	21,44	21,51	21,44	21,07	19,34	20,60
	Canberra	21,29	20,18	20,18	20,68	20,02	19,32
	Manhattan	22,63	22,24	22,25	20,23	20,32	19,91
hot_air_balloons_05	Euclidean	68,82	43,45	48,27	42,85	44,28	44,45
	Canberra	42,90	42,80	46,53	42,74	40,13	44,36
	Manhattan	50,84	44,57	42,42	39,61	39,73	38,56

The Table.1 shown Result of RMSE value, it seems RMSE value of *k*-Means PSO algorithm is smaller than *k*-Means for each image testing with difference distance measure. In other words, result of the proposed algorithm is more accurate than *k*-Means.

Table.2. A Performance Comparative processing time of K-means and *k*-means + PSO on five different image datasets

Image Test	Distance Method	Processing Time					
		***k*-Means**			***k*-Means PSO**		
		1st Trial	**2nd Trial**	**3rd Trial**	**1st Trial**	**2nd Trial**	**3rd Trial**
egret_face	Euclidean	3,287	4,308	5,133	20,929	20,990	20,907
	Canberra	2,896	4,468	6,020	17,749	18,241	20,513
	Manhattan	2,780	4,451	8,316	15,669	16,606	16,732
img_7359_copia	Euclidean	5,234	8,389	11,555	36,331	36,490	39,756
	Canberra	5,788	9,354	12,256	41,721	38,913	39,741
	Manhattan	5,206	8,262	11,412	34,783	34,230	34,320
Leafpav	Euclidean	3,873	5,327	6,690	14,307	14,235	14,501
	Canberra	3,840	5,538	7,581	17,741	16,247	16,877
	Manhattan	3,701	4,454	7,657	15,695	14,960	15,113
redberry_rb03	Euclidean	5,207	7,876	11,058	32,830	33,169	34,333
	Canberra	5,180	8,968	11,625	36,510	36,783	37,797
	Manhattan	4,943	7,844	10,503	31,934	32,673	32,625
hotair_balloons_05	Euclidean	5,389	8,269	11,206	34,778	34,785	34,337
	Canberra	5,406	8,879	12,323	44,664	36,684	36,242
	Manhattan	5,199	8,057	10,872	33,768	34,143	35,259

The Table.2 shown Result processing time of *k*-Means PSO algorithm is longer than *k*-Means algorithm for each image testing. This is because early of cluster center *k*-Means must be first processed PSO.

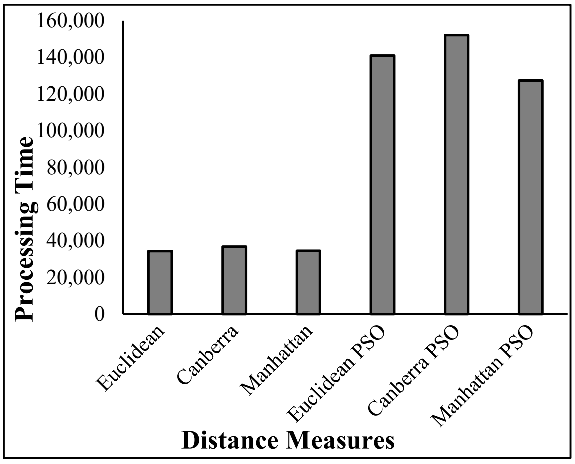


Fig.7. Block Diagram Processing Time

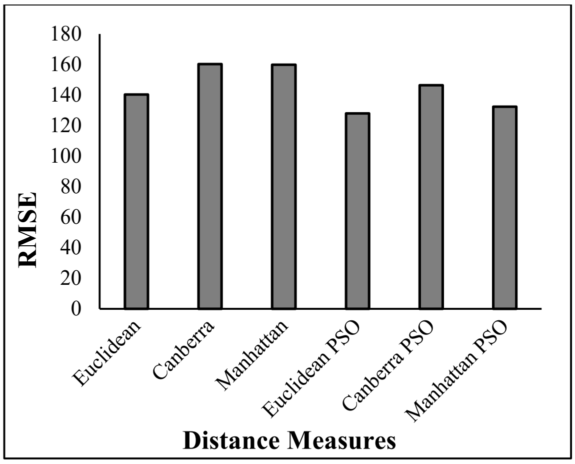


Fig.8. Block Diagram Root Mean Square Error (RMSE)

So from the Results of experiment, we have different results RMSE of Euclidean distance of *k*-Means PSO. Euclidean has less RMSE value than Manhattan. The difference RMSE between Euclidean PSO and Manhattan PSO only four point. but if we compare by processing time significant difference.

7. CONCLUSION

In this research, we have observed results of image segmentation obtained from particle swarm optimization based *k*-Means clustering. The results performance RMSE of *k*-Means PSO are proven that it increase the accuracy for datasets image, but *k*-Means PSO need the time proccess much longer than *k*-Means standard. For the distance measure method of *k*-Means PSO getting results RMSE value Euclidean 128, Manhattan 132, Canberra 146 and *k*-Means standard getting results RMSE value euclidean 140, Manhattan 160, Canberra 160. If we consider fast proccess, we can choose manhattan distance otherwise we can choose Euclidean distance for accuracy.

REFERENCES

[1] J. Nikita Sharma, Mahendra Mishra and Manish Shrivastava, “Colour Image Segmentation Techniques and Issues: An Approach”, *International Journal of Scientific and Technology*, Vol. 1, No. 4, pp. 9-12, 2012.

[2] G. Komarasamy and Amitabh Wahi, “Improving the Cluster Performance by Combining PSO and K-Means Algorithm”, *ICTACT Journal Soft Computing*, Vol. 1, No. 4, pp. 206-208, 2011.

[3] Pritesh Vora and Bhavesh Oza, “A Suvey on K-Mean Clustering and Particle Swarm Optimization”, *International Journal of Science and Modern Engineering*, Vol. 1, No. 3, pp. 24-26, 2013.

[4] V.K. Panchal, H. Kundra and J. Kaur, “Comparative Study of Particle Swarm Optimization based Unsupervised Clustering Techniques”, *International Journal of Computer Science and Network Security*, Vol. 9, No. 10, pp. 132-140, 2009.

[5] Kahkashan Kouser and Sunita, “A Comparative Study of K Means Algorithm by Different Distance Measures”, *International Journal of Innovative Research in Computer*

and Communication Engineering, Vol. 1, No. 9, pp. 2443-2447, 2013.

[6] Doreswamy and M. Umme Salma, “PSO Based Fast K-means Algorithm for Feature Selection from High Dimensional Medical Data Set”, *Proceedings of 10th International Conference on Intelligent Systems and Control*, pp. 1-6, 2016.

[7] Zeyad Safaa Younus, Dzulkifli Mohamad, Tanzila Saba, Mohammed Hasim Alkawaz, Amjad Rehman, Mznah Al-Rodhaan and Abdullah Al-Dhelaan, “Content-based Image Retrieval using PSO and K-Means Clustering Algorithm”, *Arabian Journal of Geosciences*, Vol. 8, No. 8, pp. 6211-6224, 2014.

[8] Xiaohui Cui and Thomas E. Potok, “Document Clustering Analysis Based on Hybrid PSO+K-Means Algorithm”, Available at: https://pdfs.semanticscholar.org/7c2a/8d6e5fe53248bccc49b5fc605ddd7aa23021.pdf.

[9] J. Kennedy and R. Ebenhart, “A New Optimizer using Particle Swarm Theory”, *Proceedings of IEEE 6th International Symposium on Micro Machine and Human Science*, pp. 1-4, 2001.