International Journal of Engineering & Technology, 7 (2.14) (2018) 105-109

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Silhouette index for determining optimal k-means clustering on images in different color models

Abd. Rasid Mamat*, Fatma Susilawati Mohamed, Mohamad Afendee Mohamed, Norkhairani Mohd Rawi, Mohd Isa Awang

Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Besut Campus, 22200 Terenganu, Malaysia
**Corresponding author E-mail: arm@uinsza.edu.my*

Abstract

Clustering process is an essential part of the image processing. Its aim to group the data according to having the same attributes or similarities of the images. Consequently, determining the number of the optimum clusters or the best (well-clustered) for the image in different color models is very crucial. This is because the cluster validation is fundamental in the process of clustering and it reflects the split between clusters. In this study, the k-means algorithm was used on three colors model: CIE Lab, RGB and HSV and the clustering process made up to *k* clusters. Next, the Silhouette Index (*SI*) is used to the cluster validation process, and this value is range between 0 to 1 and the greater value of *SI* illustrates the best of cluster separation. The results from several experiments show that the best cluster separation occurs when *k*=2 and the value of average *SI* is inversely proportional to the number of *k* cluster for all color model. The result shows in HSV color model the average *SI* decreased 14.11% from *k* = 2 to *k* = 8, 11.1% in HSV color model and 16.7% in CIE Lab color model. Comparisons are also made for the three color models and generally the best cluster separation is found within HSV, followed by the RGB and CIE Lab color models.

Keywords: *Cluster validation; Color model; Image filtering; K-means algorithm; Silhouette index.*

1. Introduction

Clustering is a common technique used to group data based on common patterns or similarities. Many clustering algorithms have emerged, and each algorithm is slightly different from each other in terms of input to the algorithm. For example, clusters can be modeled using distance, density or statistical distributions.

There are many approaches to clustering largely attributed to fields such as image analysis, bioinformatics, psychology, computer security and one of them is K-Means [1]. K-means is a typical clustering algorithm and of unsupervised method [2]. It is commonly used to determine the natural grouping of pixels in an image [3-4]. This algorithm is interesting in that its implementation is quite straightforward and generally get executed very fast [3]. It is also one of the simplest and effective ways used in solving clustering problems [5].

Despite all that, the process of clustering an image using a specific algorithm raises a question of whether the number of clusters obtained is optimal or most appropriate or well-clustered to the image [1]. In fact, this has triggered an invitation to another two fundamental issues. First, different algorithms or configurations of the same algorithm produce different cluster or partition, and none of them appears to be the best in all situations [6]. Another issue is many clustering algorithms cannot determine the best number of clusters for a given data and be used for future process [1]. Consequently, in [6], at the initial stage of the clustering process, it must be supplied with the information, frequently known as the *k* parameters. Further on, all the clusters are evaluated to find and select the clusters that best fit the image.

Many studies for image clustering based on K-means were conducted by researchers for many years. For example, clustering was used to enhance the accuracy of skin detection [7]. Therein, K-means clustering algorithm was used to cluster the image into 3 clusters after using explicit rules. One of the three clusters contains skin regions, and the other two contain the background and the edges of the skin regions. However, the authors did not explain how their algorithm can select the cluster that represents the skin area which is a major issue of this algorithm.

Another algorithm exploited K-means clustering for skin detection was proposed by [8]. In their algorithm, the image was first converted into CIE Lab color space and then the image pixels are segmented into three clusters based on *a* and *b* channels. Supposing one of the clusters will contain most of the skin pixels, the centroids of the three clusters are used to train an (Artificial Neural Network) ANN. The objective of the ANN is to detect which cluster has the probability to contain skin. Similar to [7], this study does not inform how the numbers of clusters selected are the most optimal or fit to the data.

The research in [9], the authors proposed a new clustering method for the white blood cells from microscopic images. The method is based on the K-means clustering algorithm. The RGB test images are converted to the CIE Lab color space and then the two color components (*a* and *b* channels) are used as features to the K-means clustering algorithm. The proposed method was tested and evaluated using blood cell images from publicly available dataset. Likewise to [7, 10], the number of best clusters that is fit for data was not disclosed.

The process of estimating how well the clusters distribution fit the structure underlying the image is known as cluster validation [6]. In other words, the cluster validity problem involves determining the

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

optimal number of clusters which can be approximated by several different techniques [11-12]. There are many techniques that can be used for cluster validation such as Silhouette index, Dun index, Calinski-Harabasz, Gamma Index, C-Index, Davies-Bouldin, Graph theory based on Dunn and Davies-Bouldin, Generalized Dunn Index, CS Index, Score Function, Symmetry Index, Point Symmetry Distance-based index, COP index, SV-Index and OS-Index [n]. All these techniques rely upon cohesion (within or intra-variance of cluster) and separation (between or inter-intra-variance of cluster) and form the basis to clusters optimum [1, 6].

2. Image database

Image database used for this research was taken from the group of Professor Wang researchers from Pennsylvania State University. The database is a subset of the Corel database and is downloadable from the Internet site (http://www.wang.ist.psu/edu). The database contains 1000 color images that are categorized into 10 groups. Each group consists of 100 images and it size is either 384 x 256 or 256 x 384 pixels. The semantic name for the group's image is African People and Village, Beach, Building, Buses, Dinosaurs, Elephants, Flowers, Horses, Mountain and Glacier and Food. Example images are presented in Figure 1.



Fig. 1: An example of an images database

3. Filtering image

A filtering process which is also referred to as smoothing is used for reducing the noise to improve the quality of the image. For this purpose, the median filter is used because this filtering is performing better than the average filtering in the sense of removing impulse noise [13-15]. Algorithm 1 performs the median filtering process [14, 16]:

Algorithm 1: Image filtering using median filter

1 Begin
2 Read the image and display it.
3 Add noise to it (For example image A)
4 For every pixel of image, *n* x *n* (eg. 3 x 3, 5 x 5) neighborhoods with the pixel (*i*,*j*) and consider as a center.
5 Sort the intensity values of the pixels in the *n* x *n* neighborhoods into ascending order
6 The value of the pixel (*i*,*j*) is replaced by the Median of the pixel values in the n x n neighborhood.
7 Repeat the above process until all pixels of the Image calculated
8 Display and save the results.
9 End

4. Image clustering by K-means

K-Means clustering, partitions the input image into *k* clusters [9, 17]. Each cluster is represented by an adaptively changing center which is also called cluster center, starting from some initial values named seed-points. This cluster computes the distances between the input image and centers of image and assigns the input data to the nearest center. Clustering algorithm assumes that a vector space is formed from the data features and tries to identify natural clustering in them. The object are clustered around the centroids μ_i, $\forall_i$ = *1* …*k* which are computed by minimizing the following objective function:

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2 \quad (1)$$

where *k* is the number of clusters, i.e S_i, *i*=1, 2, 3…, *k* and μ_i is the centroid of all points $x_j \in S_j$. In this research, the algorithm K-means requires a color image as input. The algorithm of K-means clustering is given as follows [3, 17].

Algorithm 2: K-means clustering

1 Begin
2 Read an image
3 Compute the distribution of the intensity values (of the image).
4 Using *k* random intensities initialize the centroids.
5 Repeat the step 5 until the labels of the cluster does not change anymore.
6 Cluster the image points based on the distance of their intensity values from the centroid intensity values, $c^{(i)}$.

$$c^{(i)} := \arg\min \left\| x^{(i)} - \mu_j \right\|^2 \quad (2)$$

7 Compute new centroid for each cluster, μ_i.

$$\mu_i := \frac{\sum_{i=1}^{m} 1\{c_{(i)}=j\} x^i}{\sum_{i=1}^{m} 1\{c_{(i)}=j\}} \quad (3)$$

where *k* is the number of clusters, *i* iterates over all the intensity values, *j* iterates over all the centroids (for each cluster) and μ_i is the centroid intensities.
8 End

Example results of K-Means clustering are presented in Figure 3.

5. Cluster validation

The aim of clustering process is to determine the number of clusters and the measurement to evaluate the quality of the clusters has been the main purpose of cluster validation [18]. Clustering validation evaluates the goodness of clustering results [19] and it is one of the major concerns and essentially important to the success of clustering applications [20].
Two measurement criteria have been proposed for evaluating and selecting an optimal clustering algorithm [21]. The criteria are compactness and separation. The compactness ensures the member of the cluster to be as close to each other as possible and the variance is the common value used for compactness. Meanwhile, the separation requires the clusters to be separated as distant as possible among themselves. There are three common approaches to measuring the distance between two different clusters. Firstly, compute the distance between the closest members of the clusters, followed by the distance between the most distant members and finally, the distance between centers of the clusters. This measurement has been widely used due to its computational efficiency and effectiveness for hyper sphere shaped clusters [22].

6. Silhouette index

In this paper, the Silhouette Index *(SI)* is used for cluster validation because this technique is one of the well-known techniques [23-24]. Based on [1, 4], this index was found to be one of the best performing measurement in their research. It is capable of pointing out which objects were placed well within their cluster and which ones are merely somewhere in between clusters.
To calculate SI, it is based on the partition (resulting from the cluster process) and the collection of all proximities between objects [25]. Figure 2 showed an example how to calculate SI.

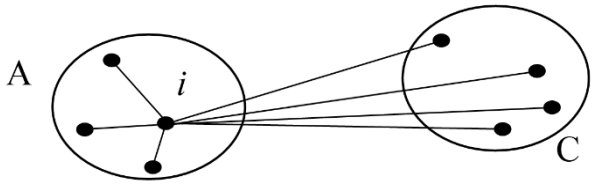


Fig. 2: Diagram to computer the SI

For example, in Figure 2, there are 2 clusters and represented as cluster *A* and cluster *C*. These contain their own objects / pixels. In cluster *A*, take one object and label as *i* and calculate *a(i);*
a(i)= average dissimilarity *i* to all other objects of *A*.
This is an average length of all lines within *A*. Consider any cluster, for example cluster *C* and this cluster is different from *A* and then compute *d(i,C),* which indicates the average length of all lines going from *i* (in cluster *A)* to *C*.
d(i,C)= average dissimilarity of *i* to all objects/pixels of *C*.
Calculate all values *d(i,C)*. Cluster *A* and cluster *C* are two different clusters (C≠ *A*) and select the smallest of those number and denote it by:
b(i) = minimum *d(i,C)*, C≠ *A*.
The value of *SI(i)* is obtained by combining *a(i)* and *b(i)* as in (4):

$$SI(i) = \begin{cases} 1 - \frac{a(i)}{b(i)} & if\ a(i) < b(i) \\ 0 & if\ a(i) = b(i) \\ \frac{b(i)}{a(i)} - 1 & if\ a(i) > b(i) \end{cases} \quad (4)$$

and summarized as in (5):

$$SI(i) = \frac{b(i) - a(i)}{\max\{a(i) - b(i)\}} \quad (5)$$

Further observation, the value of mean *SI* (i) can be interpreted as follows [24]; Excellent Split is in the range of 0.71-1.00, Reasonable Split is in the range of 0.51-0.70, Weak Split is in the range of 0.26-0.5 and below this value in category Bad Split.

7. Results and discussion

The K-means clustering algorithm and cluster validation are implemented as per the discussion above. The experiments, firstly performed in HSV, followed by RGB and CIE Lab color models. Furthermore, the discussion is based on the number of clusters and SI values for each cluster in the color models. Figure 3 shows an example results of K-means clustering and its respective centroid based on the value of *k*. In this example, the value of *k* = 5. Figure 4 shows an example graph of *SI*, for *k* = 4 in HSV color model.

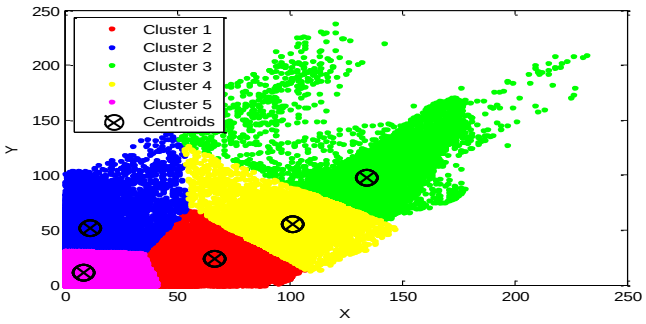


Fig. 3: An examples image for k = 5 clusters and their centroids

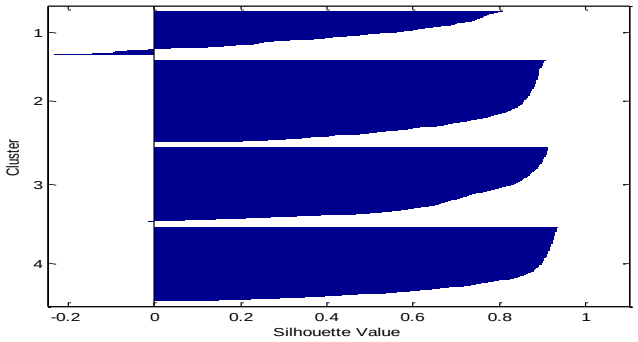


Fig. 4: The *SI* graph for image id = 301.jpg in HSV color model

7.1. K-means clustering image in RGB color model

The first section, describes the results for the clustering images in RGB color model and is shown in Table 1. The result shows the *k* = 2 contributes to the highest of average *SI* compare to others cluster. It indicates the excellent split of the clusters occur where *k* = 2.

7.2. K-means clustering image in HSV color model

This section, discuss image clustering in HSV color model and the result potray in Table 2. The result shown the *k* = 2 is the highest of average *SI* compare to others cluster. According the result, it show the excellent split of the clusters when *k* = 2 compare the others.

7.3. K-means clustering image in CIE LAB color model

Clustering image in CIE Lab color model is discussed in this secction and the result is represented in Table 3. The *k* = 2 shows the highest value of average SI. It can concluded that the split of cluster is excellent when *k* = 2.

7.4. Comparison the average *SI* of clustering in RGB, CIE LAB and HSV color model

Figure 5 illustrates a comparison of the average SI for each cluster in a different color space. It is noticed that the average *SI* of the cluster *k* = 2 is better in CIE Lab compare to others color space, in the range of 11.61%. Starting from cluster *k* = 3 to 8, the highest average *SI* i found in HSV, followed by RGB and CIE Lab. For example in cluster number 3, the highest average *SI* is in HSV, followed by RGB and CIE Lab and the range between the highest and the lowest is between 4.5%.



Fig. 5: Comparison average SI for each cluster in difference color model

Table 1: The *SI* of each cluster in RGB color model

Number	ID Image	SI						
		$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$	$k = 8$
1	1.jpg	0.7154	0.6929	0.6788	0.6414	0.5969	0.5833	0.5612
2	101.jpg	0.8685	0.7018	0.7043	0.587	0.6735	0.6564	0.6779
3	201.jpg	0.8900	0.8459	0.8261	0.7637	0.7637	0.7466	0.7441
4	301.jpg	0.8881	0.7938	0.7772	0.7549	0.7262	0.6988	0.6780
5	401.jpg	0.9227	0.9214	0.9208	0.9143	0.7295	0.8775	0.7833
6	501.jpg	0.6809	0.6531	0.6397	0.6212	0.5770	0.5939	0.5976
7	601.jpg	0.7095	0.6848	0.6189	0.6105	0.6082	0.6207	0.6019
8	701.jpg	0.6618	0.7216	0.6364	0.6277	0.5746	0.5248	0.5488
9	801.jpg	0.8213	0.7670	0.7528	0.7154	0.683	0.6517	0.6405
10	901.jpg	0.7044	0.7048	0.7062	0.7053	0.6491	0.6379	0.6173
Average *SI*		0.7863	0.7487	0.7261	0.6941	0.6582	0.6592	0.6451

Table 2: The *SI* of each cluster in HSV color model

Number	ID Image	SI						
		$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$	$k = 8$
1	1.jpg	0.5737	0.5507	0.5137	0.5634	0.5582	0.5622	0.5691
2	101.jpg	0.7662	0.6895	0.6534	0.6899	0.7183	0.7248	0.7079
3	201.jpg	0.7413	0.7979	0.6717	0.6092	0.6495	0.6204	0.5856
4	301.jpg	0.6762	0.7558	0.7248	0.6704	0.6734	0.6600	0.6285
5	401.jpg	0.9074	0.8557	0.8788	0.4200	0.2951	0.3558	0.3201
6	501.jpg	0.6264	0.6551	0.5427	0.5419	0.5848	0.5557	0.5619
7	601.jpg	0.6576	0.6858	0.6220	0.5699	0.5798	0.5809	0.5571
8	701.jpg	0.502	0.4772	0.566	0.5452	0.5174	0.5059	0.5204
9	801.jpg	0.6576	0.6266	0.7253	0.6742	0.6240	0.6456	0.6212
10	901.jpg	0.6431	0.5957	0.5893	0.5957	0.5805	0.5787	0.5752
Average *SI*		0.6751	0.6690	0.6488	0.5879	0.5781	0.5790	0.5647

Table 3: The SI of each cluster in CIE Lab color model

Number	ID Image	SI						
		$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$	$k = 8$
1	1.jpg	0.7094	0.5306	0.5679	0.6307	0.6338	0.6392	0.5487
2	101.jpg	0.8713	0.8633	0.8171	0.7933	0.5858	0.7534	0.7681
3	201.jpg	0.8162	0.7499	0.7811	0.66	0.6432	0.5221	0.5248
4	301.jpg	0.8964	0.7811	0.6652	0.7199	0.6602	0.6277	0.6275
5	401.jpg	0.8913	0.8201	0.8405	0.8306	0.8211	0.8113	0.7921
6	501.jpg	0.6890	0.6855	0.6588	0.631	0.5979	0.6167	0.6399
7	601.jpg	0.9015	0.7075	0.6225	0.6053	0.5899	0.5986	0.5924
8	701.jpg	0.6736	0.6174	0.6192	0.5942	0.539	0.5209	0.5432
9	801.jpg	0.7467	0.7064	0.6835	0.6523	0.649	0.6299	0.5992
10	901.jpg	0.7228	0.7680	0.686	0.6659	0.7027	0.6983	0.663
Average *SI*		0.7918	0.7229	0.6941	0.6783	0.6422	0.6418	0.6298

8. Conclusion and future work

One way to compare the optimum number of clustering (k) that is obtained by clustering algorithm is using cluster validation. In this research, the *SI* is used to measure the validation of cluster resulting by K-means clustering method on different color model. The goal is to provide the best (well-clustered) or optimum clusters wherein to show the objects/pixels in the appropriate clusters. In addition, the comparison between the color models can also be done to determine which color model that generates optimal cluster. The results showed that the optimal cluster occurs when $k = 2$ and for the next $k = 3$ to 8, it generally becomes less optimal for each color model. The average value of *SI* is reduced from $k = 2$ to $k = 8$ by 14.1% for RGB color model, 11.1% for model HSV color model and 16.2% for CIE Lab color model. In other words, the best cluster occurs when the images are clustered in two clusters only. Comparison of the average value of *SI* shows at $k = 2$, the best color model is the CIE lab from the range of k. Generally, based on *SI*, the

conclusion can be made that the HSV color model is the best (well-clustered), followed by the RGB color model and the latter is the CIE Lab.

For the future works, the time to process clusters, adding some color model such CIE LUV, YCbCr, CMY and validation methods also need to be considered.

Acknowledgement

The authors would like to thanks Research Management, Innovation and Commercialization Centre (RMIC), Universiti Sultan Zainal Abidin (UniSZA) for the support this project.

References

[1] Arbelaitz O, Gurrutxaga I, Muguerza J, PéRez JM & Perona I (2013), An extensive comparative study of cluster validity indices. *Pattern Recognition* 46, 243–256.

[2] MacQueen J (1967), Some methods for classification and analysis of multivariate observations. *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, 281–297.

[3] Dubey SR, Dixit P, Singh N & Gupta JP (2013), Infected fruit part detection using K-means clustering segmentation technique. *International Journal of Artificial Intelligence and Interactive Multimedia* 2, 65–72.

[4] Vendramin L, Campello RJ & Hruschka ER (2010), Relative clustering validity criteria: A comparative overview. *Statistical Analysis and Data Mining: The ASA Data Science Journal* 3, 209–235.

[5] Küçükkülahli E, Erdoğmuş P & Polat K (2016), Brain MRI segmentation based on different clustering algorithms. *International Journal of Computer Applications* 155, 37–40.

[6] Pal NR & Biswas J (1997), Cluster validation using graph theoretic concepts. *Pattern Recognition* 30, 847–857.

[7] Sree PK & Babu IR (2013), Face detection from still and video images using unsupervised cellular automata with K means clustering algorithm. *ICGST-GVIP Journal* 8, 1–7.

[8] Bevilacqua V, Filograno G & Mastronardi G (2008), Face detection by means of skin detection. *Proceedings of the International Conference on Intelligent Computing*, pp. 1210–1220.

[9] Dhanachandra N, Manglem K & Chanu YJ (2015), Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science* 54, 764–771.

[10] Salem NM (2014), Segmentation of white blood cells from microscopic images using K-means clustering. *Proceedings of the* IEEE *31st National Radio Science Conference*, pp. 371–376.

[11] Liu Y, Li Z, Xiong H, Gao X & Wu J (2010), Understanding of internal clustering validation measures. *Proceedings of the IEEE 10th International Conference on Data Mining*, pp. 911–916.

[12] Qiao H & Edwards B (2009), A data clustering tool with cluster validity indices. IEEE *International Conference on Computing, Engineering and Information*, pp. 303–309.

[13] Manoharan S & Sathappan S (2013), A novel approach for content based image retrieval using hybrid filter techniques. *Proceedings of the IEEE 8th International Conference on Computer Science and Education*, pp. 518–524.

[14] Kannan A, Mohan V & Anbazhagan N (2010), An effective method of image retrieval using image mining techniques. *International journal of Multimedia and Its Applications* 2, 17–26.

[15] Szeliski R (2010), *Computer vision: Algorithms and applications*, Springer Science and Business Media.

[16] Zhao H, Kim P & Park J (2009), Feature analysis based on Edge Extraction and Median Filtering for CBIR. *Proceedings of the IEEE 11th International Conference on Computer Modelling and Simulation*, pp. 245–249.

[17] Arumugadevi S & Seenivasagam V (2016), Color image segmentation using feedforward neural networks with FCM. *International Journal of Automation and Computing* 13, 491–500.

[18] Arumugadevi S & Seenivasagam V (2016), Color image segmentation using feedforward neural networks with FCM. *International Journal of Automation and Computing* 13, 491–500.

[19] Maulik U & Bandyopadhyay S (2002), Performance evaluation of some clustering algorithms and validity indices. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 1650–1654.

[20] Dubes RC & Jain AK (1988), *Algorithms for clustering data*, Prentice-Hall.

[21] Berry MJ & Linoff G (1997), *Data mining techniques: For marketing, sales, and customer support*. John Wiley and Sons.

[22] Rendón E, Abundez I, Arizmendi A & Quiroz EM (2011), Internal versus external cluster validation indexes. *International Journal of Computers and Communications* 5, 27–34.

[23] Chaimontree S, Atkinson K & Coenen F (2010), Best clustering configuration metrics: Towards multiagent based clustering. *Proceedings of the International Conference on Advanced Data Mining and Applications*, pp. 48–59.

[24] Burney SA & Tariq H (2014), K-means cluster analysis for image segmentation. *International Journal of Computer Applications* 96, 1–8.

[25] Rousseeuw PJ (1987), Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* 20, 53–65.