International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.7 (2016), pp.107-116
http://dx.doi.org/10.14257/ijsip.2016.9.7.10

Automated Numbers of Cluster Determination Using the Combination of Entropy and Histogram Peaks from Multiple Images

Swati Chowdhury[1], Sudipta Roy[2], Anirban Mitra[3] and Prasenjit Das[4]

Department of Computer Science and Engineering, Academy of Technology, G T Road (Adisaptagram), Aedconagar, Hooghly-712121, West Bengal, India
[1]swatichowdhury55@gmail.com, [2]sudiptaroy01@yahoo.com, [3]anirban.mitra.cse@gmail.com, [4]p.das@aot.edu.in

Abstract

Image segmentation subdivided an image into its constituent regions or segments. This regions or segments of an image is known as 'cluster' and the method used for this is called 'clustering method'. There are different methods or algorithms are out there to segment an image. There is a problem with those algorithms user has to supply the number of cluster in which it has to be segmented. Here we introduced a method using combination of both entropy of RGB color component and histogram to automatic determination of cluster present in a color image.

Keywords: *Cluster, Segmentation, Histogram, Entropy, Relative Error, Image Processing.*

1. Introduction

Image segmentation segment the input image into different sub division or region based on its color, depth, texture, surface, gray scale, motion and etc which enhance the view to see an ordinary image. This kind of segment or region is called 'cluster' and the method has been used for grouping the objects belong to the same cluster is called 'clustering'. A cluster is the combinations of data set which are similar to each other and have distinguish dissimilarity with the data set of other cluster. Determine the number of cluster before finding the clusters is one of the major issue because the algorithm use for determining the cluster like k-means, isodata, knn, fuzzy c-means and etc need to input the number of cluster from the user side and in isodata there are six parameters that used be provided by the user also. To overcome this problem we have introduce a technique using the combination of entropy of RGB color component and histogram peak to calculate number of clusters automatically. After that we have use the k-means clustering algorithm to segment the input image into detected number of clusters.

K-means algorithm is used to cluster 'n' object based on attributes into 'k' partitions where $k < n$. K-means is a partitioning based algorithm where user have to provide the number of clusters. The grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroids. Here the input of number of clusters automatically given to the k-means algorithm to segment the clusters. This technique has been tested with some images and with some real images and the result is pretty much impressive.

The rest of the paper has organized as followed: Brief review on existing work has been describe in section 2, the details of our proposed methodology has been describe in section 3, the results on different input images has been describe in section 4, finally conclude our paper in section 5.

ISSN: 2005-4254 IJSIP

2. Review Work

Sanguinetti *et al.* suggests [1] automatic determination of the number of clusters using spectral algorithms. Spectral clustering is a method of grouping the input data into a user pre-specified number of cluster. The beginning point is an affinity matrix from the input data to form an N x N matrix according to the in between different points in matrix. Here k-means algorithm is modified along with radial directions and transversal directions. This specified that the points in one cluster have the minimal difference from the same cluster centre. Mercovich *et al.* suggest [2] automatic clustering of multispectral imagery by maximization of the graph modularity. Graphical representation of the input image has been generated by spectral data structure. The clustering of the data has been done by optimal modularity method. Background where image has been visualized as a graph, graph partitioning it can see that created can be divided again or not using modularity and edge creation the edges of the graph considering by distance measure these three methods have been used here.

Luccehse *et al.* suggest[3] unsupervised segmentation of color image based on k-means clustering on the chromaticity plane. Here introduce an ad hoc k-means algorithm based of chromaticity of color image in 2D plane. In the beginning of this method it measure the chromaticity coordinates of input color image then with luminance values are assigned to its proper chromaticity coordinates. During the changes of surface curvature which is caused change in color of the object this method almost gives the same result. Koonsanit *et al.* suggest [4] determination of the initialization number of clusters in k-means clustering application using co-occurrence statistics techniques for multispectral satellite imagery. Here co-occurrence matrix has been introduced to cluster the image in segments and local maximum technique to calculate the number of segments. This technique is only verified with multiple satellite images where it gives satisfactory result with pre-defined numbers of regions in k-means clustering algorithm.

Wagstaff *et al.* suggest [5] constrained k-means clustering with background knowledge. Here levels have been used with results to evaluate the cluster based on rand index. COP k-means has been used here compare with the constrained k-mean on GPS images. If there is some mistake has been made in the early stages then after the whole procedure there produced a cluster without any clustering point but it can back track itself to correct the error and produced a correct cluster. Ray *et al.* suggests[6] determination of number of clusters in k-means clustering and application in color image segmentation. This method is based on the inter-cluster and intra-cluster distance calculation of input data points. Intra-cluster distance which is has to be minimum, has been calculating by the distance between the points and their respective cluster centers. Here the concept of validity measure has been used with the intra and inter cluster distance measure. This process has been successfully examined with synthetic images. Dubey *et al.* suggests [7] infected fruit part detection using k-means clustering segmentation technique which will be very much helpful in agricultural science. Here k-means clustering algorithm has been used along with L*a*b color space technique to detect the infected fruit or fruit part. K-means algorithm combined with defect segmentation can find possible infected parts like stem and calyx of the apple which is the experimental object here but it a manual process and for the future work it should be detected in the automated way.

Burney *et al.* [8] suggests k-means cluster analysis for image segmentation. Here spatial coordinate's information of the input image has been used for calculating region growing method. After this calculation the produced result has been used in k-means algorithm for further process. In this process the k-means clustering examined with RGB and L*a*b color space to get cluster and compared with human vision number of clusters, where L*a*b color space give more reliable results than the RGB color space with real images. K.Srinivas *et al.* suggests[9] a scientific approach for segmentation and clustering technique of improved k-means and neural networks. In this procedure k-means clustering algorithm combined with artificial neural network (ANN) to construct a bunch of effective algorithms for image segmentation. Here lower and upper limit values have been chosen based on histogram. If any point value is valid its inside otherwise outside value. The k-means has been performed on these inside values to form the cluster of given data set. Pakhira *et al.* suggests[10] finding number of clusters before finding clusters. Here dark square blocks and algorithm concept have been used to automatically detect the number of clusters. In this method the algorithms have been examined on images.

3. Method

We introduce a method using histogram and entropy of an image we can determine the number of clusters before using any of those clustering technique. Here we used the k-means method for the clustering purpose.

In determining number of clusters we have take a color image and change it to a grayscale image. The input image has been converted converts RGB image or colormap to grayscale image. RGB values to grayscale values by forming a weighted sum of the R, G, and B components. Mathematical equation of this function:

$$0.2989 * R + 0.5870 * G + 0.1140 * B$$

Also we split up the three color components R (red), G (green), B (blue) of the image using the component splitting.

This component splitting procedure is done here for calculating “entropy” of the image in different color components. Entropy returns, a scalar value representing the entropy of grayscale image I. Entropy is a statistical measure of randomness that can be used to characterize the texture of the input image. Mathematical expression of entropy E of color component is:

$$\text{Entropy} = -\sum_i Pj \, log2 \, Pj$$

After converting the image into gray scale we can found the histogram of that image using histogram function. The displays a histogram for the image I above a grayscale color bar. The number of bins in the histogram is specified by the image type. If I is a grayscale image, histogram uses a default value of 256 bins.

In histogram equalization input pixel intensity x, is transformed to a new intensity value x' by T. The histogram function T is the product of a cumulative histogram and scale function. The scale function needed to fit the new intensity value within the range of intensity value for example 0 to 255. The mathematical equation of histogram is:

$$x' = T(x) = \sum_{i=0}^{n} ni \; \frac{max\ intensity}{N}$$

Where ni is the number of pixels at intensity i, and N is the total number of pixel in the image. Now after getting the histogram of the image we can calculate the average value of that histogram and only show and count the numbers of that histogram with are above the average value.

$$I_{avg} = \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{(m,n)}{m * n}$$

Where Iavg is the average value of the histogram peaks and m, n are image dimension. From this average value we can calculate the numbers of peaks which are more than that average value and also discard those which are less than that average value. Then we can calculate the numbers of sharp peaks which are greater than three peaks before and after them. This is also a method to detect the number of clusters.

if(avgplus(i-3)<avgplus(i-2)<avgplus(i-1)<avgplus(i)>avgplus(i+1)>
avgplus(i+2)> avgplus(i+4))

c1 = c1+1;

After determine the number of clusters we use k-means method for splitting the image into different region. Mathematical equation of k-mean clustering algorithm is:

$$J(v) = \sum_{i=1}^{c} \sum_{j=1}^{ci} (\| xi - vi \|)^2$$

Where ||xi-vj|| is the Euclidean distance between xi and vj, ci is the number of data points in the cluster, c is the number of cluster centers.

4. Results and Discussion

In this method we have taken input RGB image as our original color input image. Before examined on any real images we have examined our procedure on an image. Therefore we will firstly covert our RGB image into a grayscale image using in matlab.

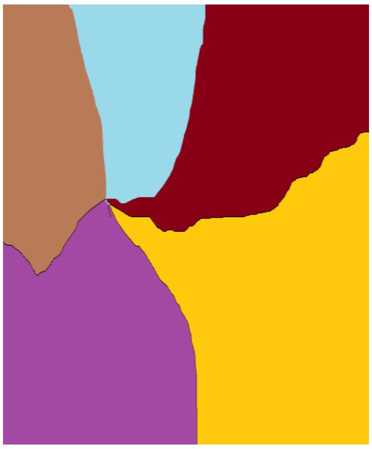

Figure 1. Original Image

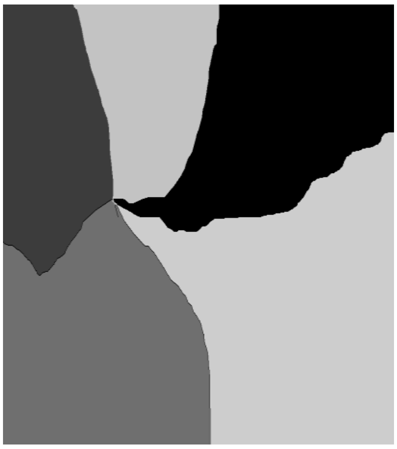

Figure 2. Grayscale Image

Here as we can see in the Figure 1 there is our original RGB image and in Figure 2 there is the converted grayscale image. Now there is a question why should we have to convert the RGB image into grayscale image because we have been performed our histogram function on the grayscale image to find the histogram.

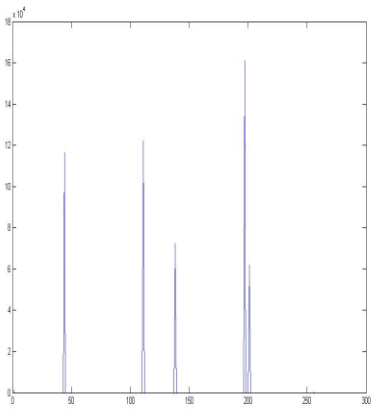

Figure 3. (a) Original Histogram Image

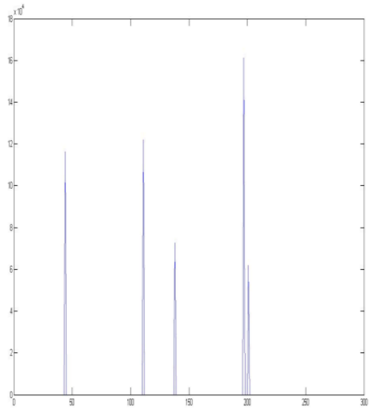

Figure 3. (b) Histogram Above Average Value

The histogram image of grayscale image has been shown in Figure 3(a). This is the original histogram of that image without any subtraction. Now in Figure 3(b) we can image of histogram which values are more than average value of original histogram image. We calculate the average value of original histogram and it is 2.0899e+003. The number of values with are above the histogram average value is 5. Then we can calculate the numbers of sharp peaks which are greater than three peaks before and after them. Here the numbers of sharp peaks are 5.

We also have done the entropy of RGB color component of the input image. At first we have divide the color component image for each color component like red(R), green(G) and blue(B). Then we have compare and concatenate the color component with the original color image. We have store this concatenation value into three different variables for three color component. After that we have used 'entropy' function on that variable and display the value of entropy for each color component. The entropy value for red(R) color component is 1.6620, green(G) color component is 1. 3201 and for blue(B) color component is 1.6620. After finding all three entropy for RGB color component we have to first combine these entropy values with the numbers of sharp peaks that we have already found from histogram peak values averaging method. After the finding the sum of these values we have to divide it with 2 to get the number of cluster in an automated way. Here after all these procedure we have found the 4.82205 as our number of cluster. In k-means clustering algorithm which has been used here to segment the image into different clusters, is not allowed to take the fractional numbers for this reason we have taken the numbers which is near to this result for this image which is 5. The number of cluster here is 5 and it is the automatic input for k-means algorithm.

We have also compared this automated number of clusters with number of clusters visualized by human vision. For case these two way cluster detection method produced same number, 5.

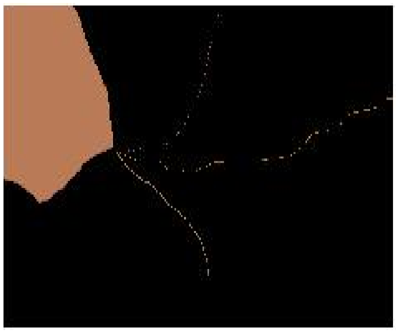

Figure 4.(a) First Cluster **Figure 4.(b) Second Cluster**

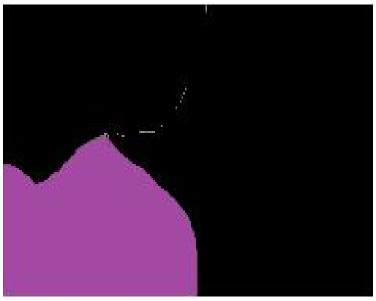

Figure 4.(c) Third Cluster **Figure 4.(d) Fourth Cluster**

Figure 4.(e) Fifth Cluster

After these we have calculate difference between numbers of sharp peaks in histogram image and number of clusters determine by human visual and also find the relative difference by dividing the difference with number of clusters find by human vision. For this case difference and relative difference both are 0.

Table 1. Table of Analyzing Proposed Procedure on Different Images

Sl. No.	Name of the image	Entropy for			No of cluster depend on histogram(i)	No of X= $\frac{r+g+b+i}{2}$	No of clusters depend on visual(j)	Difference k=i-j	Relative difference (d) I= k/j
		Red(r)	Green(g)	Blue(b)					
1.	Image	1.5809	1.5809	1.2139	4	4.18785	4	0	0
2.	Image1	1.6620	1.3201	1.6620	5	4.82205	5	0	0
3.	Image2	1.4460	1.4460	1.4460	4	4.169	5	1	0.2
4.	Image5	1.7196	1.6872	1.7046	6	5.557	8	2	0.25
5.	table	3.3980	3.3755	3.4304	4	7.10195	5	1	0.2
6.	koala	3.5090	3.4843	3.4887	5	7.741	6	1	0.1666
7.	tulips	3.3635	3.4903	1.6259	3	5.73985	4	1	0.25
8.	len	3.3360	3.4496	3.2411	3	6.51335	4	1	0.25
9.	sepals	3.3951	3.3883	3.2608	3	6.5221	4	1	0.2
10.	peas	3.4944	3.4222	3.0959	2	6.00625	2	0	0
11.	flowers	3.1969	3.4643	3.2284	3	6.4448	4	1	0.25
12.	craft	3.3785	3.4422	2.2850	3	6.05285	4	1	0.25

13.	sun	3.0491	3.4403	3.2527	3	6.37105	4	1	0.25
14.	lotus	3.4033	3.3682	3.2882	2	6.02985	2	0	0
15.	desert	3.4788	3.4001	2.9109	7	8.3949	9	2	0.222

For experiment of our proposed method we take 15 different images to examine. At first we found the entropy of color components of those images by splitting them into RGB component which is red(R), green(G) and blue(B) and calculate their entropy value. The entropy values respectively store in ‘r’, ‘g’ and ‘b’ variables. Then we have transformed these images to their respective grayscale images to find the histogram of those images. We also calculate the average histogram value from this histogram and count the no of peaks which are above the average histogram value. Then we can calculate the number of sharp peaks from sorted number of peaks. The number of sharp peak value stored in ‘i’ variable. Now we have to calculate the number of clusters by calculating sum of entropy of RGB color component and sharp peaks and divided it by 2. The result of this calculation is stored in ‘X’ variable. After all these procedure we have analysis these 15 images by our human vision to find out the numbers of clusters which is store in ‘j’ variable. Now we have to find difference between the sharp peaks (i) and the numbers of clusters define by human vision (j). This result value is stored in ‘k’ variable. Now at last we found the relative difference by dividing the difference (k) with the number of clusters detected in human vision (j). This relative difference value is stored in‘d’ variable.

5. Conclusion

In this method we have examined on 15 images with entropy of color component and histogram to automatically detect the numbers cluster and compare that number with the human vision. In these 15 images the percentage of relative difference is not more than 25%. So we can say that our proposed method is almost producing nearer result to the human vision. In future we will be improve our method to produce more and more accurate result and examined with different types of images to detect more specific area of interest.

Appendix

Now here some other examples of our proposed method with original image have been shown below.

Figure 5.(a)

Figure 5.(b)

Figure 5.(c)

Figure 5.(d)

Figure 5.(e)

This is another example of our proposed method where the RGB input image Figure 5(a) has been divided into 4 different regions shown in 5(b), 5(c), 5(d) and 5(e).

Figure 6.(a)

Figure 6.(b)

Figure 6.(c)

Figure 6.(d)


Figure 6.(e)


Figure 6.(f)

This is a real life RGB image which has been segmented into 5 different clusters with help of our proposed method. The original image is shown in Figure 6(a) and the different clusters images have been shown in 6(b), 6(c), 6(d), 6(e) and 6(f).


Figure 7.(a)

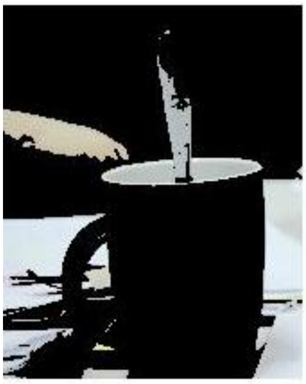
Figure 7(b)


Figure 7.(c)


Figure 7.(d)

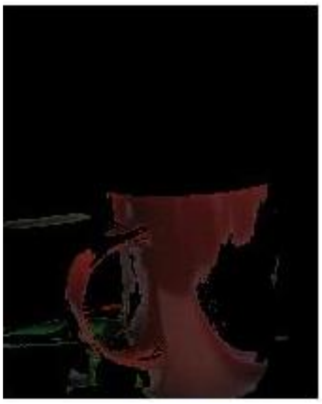
Figure 7.(e)

Here is another example of real life RGB image which has 4 clusters segment. The original image is shown in 7(a) and clusters segmentation images have been shown in 7(b), 7(c), 7(d) and 7(e).

References

[1] G. Sanguinetti, J. Laidler and N. D. Lawrence, "Automatic Determination of the Number of Clusters Using Spectral Algorithms, Department of Computer Science, University of Sheffield, 211 Portobello Street, Sheffield S1 4DP.

[2] R. A. Mercovich, A.Harkin and D. Messinger, "Automatic clustering of multispectral imagery by maximization of the graph modularity", Center for Imaging Science and School of Mathematical Sciences, Rochester Institute of Technology.
[3] L. Lucchese and S. K. Mitra, "Unsupervised Segmentation of Color Image Based on k-means Clustering in the Chromaticity Plane", Dept. of Electrical and Computer Engg., University of California, Santa Barbara, Dept of Electronics and Informatics, University of Padua, Italy.
[4] K. Koonsanit, C. Jaruskulchai and A. Eiumnoh, "Determination of the Initialization Number of Clusters in K-means Clustering Application Using Co-Occurrence Statistics Techniques for Multispectral Satellite Imagery", International Journal of Information and Electronics Engineering, vol. 2, no. 5. **(2012)**.
[5] K. Wagsta, C. Cardie, S. Rogers and S. Schroedl, "Constrained K-means Clustering with Background Knowledge", Department of Computer Science, Cornell University, Ithaca, NY 14853 USA, DaimlerChrysler Research and Technology Center, 1510 Page Mill Road, Palo Alto, CA 94304 USA, Proceedings of the Eighteenth International Conference on Machine Learning, **(2001)**, pp. 577-584.
[6] S. Ray and R. H. Turi, "Determination of Number of Clusters in *K*-Means Clustering and Application in Colour Image Segmentation", School of Computer Science and Software Engineering, Monash University, Wellington Road, Clayton, Victoria, 3168, Australia.
[7] S. R. Dubey, P. Dixit, N. Singh and J. P. Gupta, "Infected Fruit Part Detection using K-Means Clustering Segmentation Technique", GLAU, Mathura, India, Dept. of Inform. Tech., Dr. M.P.S Group of Institutions College of Business Studies, Agra, India, Dept. of Comp. Engg. & Applications, Poornima Group of Colleges, Jaipur, India, Systems Engineer in Infosys Limited, Bangalore, India, International Journal of Artificial Intelligence and Interactive Multimedia, vol. 2, no. 2.
[8] S. M. A. Burney and H. Tariq, "K-Means Cluster Analysis for Image Segmentation", International Journal of Computer Applications, vol. 96, no.4, **(2014)**, pp. 0975 – 8887.
[9] K. Srinivas and Dr. V. Srikanth, "A Scientific Approach for Segmentation and Clustering Technique of Improved K-Means and Neural Networks", K.L.University,Vaddeswaram A.P., India, International Journal of Advanced Research in Computer Science and Software Engineering, vol. 2, no. 7, **(2012)**.
[10] M. K. Pakhira, "Finding Number of Clusters before Finding Clusters", Kalyani Government Engineering College,Kalyani, India, SciVerse ScienceDirect, Procedia Technolog, vol. 4, **(2012)**, pp. 27-37.
[11] A. K. Pujari, "Data Mining Techniques", Universities Press, **(2009)**.

