International Journal For Technological Research In Engineering
Volume 5, Issue 8, April-2018 *ISSN (Online): 2347 - 4718*

EDGE DETECTION IN IMAGE USING K-MEANS CLUSTERING

Dr.Mridul Kumar Mathur[1], Manesh Singh[2]
[1]Asst. Prof. (Selection Grade), Dept. of Comp. Sc., LMCST, Jodhpur (Raj.)
[2]Ph.D. Research Scholar, Dept. of Comp. Engineering,
Jodhpur National University (JNU), Jodhpur (Raj.)

ABSTRACT: In this work K-means image clustering technique is also used for edge detection in an image. This K-means clustering method is suitable to find various segments of images very clearly. As the number of clusters is increased it is possible to find the different part of the image which can be used to perform different experiments on an image.
Keywords: K-means, edge detection, image segmentation

I. INTRODUCTION

In this paper edge detection method named k-means clustering to find out edges in an image to segment it.

1.1 K-means Clustering Algorithm:

The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early group is done. At this point, it is need to re-calculate k new centroids as centers of the clusters resulting from the previous step. After these k new centroids, a new binding has to be done between the same data points and the nearest new centroid. A loop has been generated. As a result of this loop it may notice that the k centroids change their location step by step until no more changes are done.

This algorithm aims at minimizing an objective function, e.g. a squared error function. The objective function is expressed as:

$$\sum_{j=1}^{k}\sum_{i=1}^{n}\left\|\boldsymbol{x_i}^{(j)}-\boldsymbol{c_j}\right\|^2$$

Where $\left\|\boldsymbol{x_i}^{(j)}-\boldsymbol{c_j}\right\|$is a chosen distance measure between a $\boldsymbol{x_i}^{(j)}$data point and the cluster centre *cj*, is an indicator of the distance of the *n* data points from their respective cluster centers.

This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

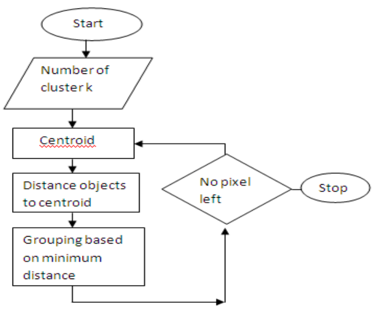

Figure **Error! No text of specified style in document.** Flow Chart of K-means Clustering Algorithm

The advantages of K-Means Clustering are as below:

- K-Means Clustering works well when clusters are not well separated from each other, which is frequently encountered in images.
- K-Means algorithm is easy to implement.
- Its time complexity is *O(n)*, where n is the number of patterns. It is faster than the hierarchical clustering.
- It works perfectly fine with all natural images.

The disadvantages of K-Means Clustering are as below:

- The number of clusters is to be predefined in each iteration, which creates a problem for of huge image databases.
- K-means clustering has problems when clusters are of different sizes, Densities, and Non-globular shapes.
- K-means clustering has problems when the data contains outliers.
- We cannot show the clustering details as Hierarchical clustering does.

II. METHODOLOGY AND RESULTS

2.1 Algorithm and working of K-means clustering algorithm

Colour image is constituted with 3 dimensional colourspace to enhance its image quality. This method segments the colours in in an S*p*q colour space and K-means clustering. The steps of this methods are as follows:

Step 1: Input the color image and convert it into a three dimenstional matrix using matlab .

Step 2: The Decorrelation stretching is applied for colour separation of an image

Step 3: Convert Image from RGB Color Space to S*p*q* Color Space

Step 4: Classify the Colors in 'p*q*' Space Using K-Means Clustering. In this K-means to cluster the objects into three clustersusing the Euclidean distance metric.

Step 5: Every pixel is labelled in the Image Using the Results from K-MEANS

Step 6: Create Images that Segment the Image by Color.

Step 7: Nuclei is separated into a Separate Image

2.2 Experimental Results

Different types of edges of image are detection by varying number of centres in the images. This centres varies according to clusters presented in that image .The more number of colours and images which is shown in figure 2 to figure 6.

Figure 2 : Objects in Cluster 1

Figure 3: Objects in Cluster 2

Figure 4: Objects in Cluster 3

Figure 5: Objects in Cluster 4

Figure 5 shows most concentrated part of this image in the form of a cluster.

Figure 6: Objects in Cluster 5

III. CONCLUSION

Above figures shows different segments in the image. This K-means clustering method is suitable to find various segments of images very clearly. As the number of clusters is increased it is possible to find the different part of the image which can be used to perform different experiments on an image.

REFERENCES

[1]S. Dhawan, "A Review of Image Compression adn Comparison of its Algorithm," International Journal of Electronics & Communication Technology, vol. 2, no. 1, pp. 22-26, March 2011.

[2] S. Sharma and S. Kaur, "Image Compression using Hybrid of DWT, DCT and Huffman Coding," International Journal for Science and Emerging Technologies with Latest Trends, vol. 5, no. 1, pp. 19-23, January 2013.

[3] D. L. E. George and N. A. Minas, "Speeding Up Fractal Image Compression Using DCT Descriptors," Journal of Information and Computing Science, vol. 6, no. 4, pp. 287-294, 2011.

[4] A. E. Jacquin, "Image Coding Based on a Fractal Theory of Iterated Contractice Image Transformations," IEEE TRANSACTIONS ON IMAGE PROCESSING. , vol. 1, no. 1, pp. 18-30, January 1992.

[5] Kulbir Singh, Rajiv Saini and Rajiv Saxena, "Performance of Wavelet, Fractional Fourier and Fractional Cosine Transform in Image Compression," International Journal of Recent Trends in Engineering, vol. 2, no. 7, pp. 43-45, November 2009.

[6] N. Dara, "A Survey on Compression Techniques," Interntional Journal of Computer Science & Management Studies, vol. 13, no. 10, pp. 28-31, December 2013.

[7] M. Sharma, "Compression Using Huffman Coding," International Journal of Computer Science adn Network Security, vol. 10, no. 5, pp. 133-141, 2010.

[8] A. Averbuch, "Image Compression Using Wavelet Transform and Multiresolution Decomposition," IEEE Transactions on Image Processing, vol. 5, no. 1, 1996.

[9] K. Cabeen and P. Gent, Image Compression and the Discrete Cosine Transform, College of Redwoods.

[10] M. Theirschmann, U. Martin and R. Rosel, "New Perspective on Image Compression," D. Fritsh& D. Hobbie, Eds. , pp. 189-199, 1997.

[11]A. S. Lewis and G. Knowles, "Image Compression Using 2-D Wavelet Transform," IEEE Transactions on Image Processing, vol. 1, no. 2, pp. 244-250, 1992.

