# Object Recognition Using Image Segmentation

**Sumer Singh**

***Abstract*— Object recognition is basically an attempt to mimic the human capability to distinguish different objects in an image. This paper presents Scale-Invariant Feature Transform (SIFT) and segmentation methods such as Graph Cut, K-means, and Linde–Buzo–Gray (LBG) algorithm. In SIFT, interesting points of the object are extracted to provide a "feature description" of the object. Whenever we work with the image in any application, initial step is to segment the image in order to solve its complexity. This description can then be used to identify the object. Image segmentation is the process of dividing the given image into homogenous regions with respect to certain features. The segmentation of images is the basic thing for understanding the images.**

***Index Terms*— Graph cut method, k-means algorithm, Linde-Buzo-Grey algorithm, SIFT, Segmentation.**

## I. INTRODUCTION

Humans can recognize any object in the real world easily without any efforts; on contrary machines by itself cannot recognize objects. Thus object recognition techniques need to be developed which are less complex and efficient. Algorithmic descriptions of recognition task are implemented on machines for object recognition. This paper is the combination of Scale-Invariant Feature Transform (SIFT) and Segmentation methods. SIFT feature is invariant for rotations, scale changes, and illumination changes and it is often used for object recognition, while Graph Cut, K-means and Linde–Buzo–Gray (LBG) algorithms are proposed as a segmentation method. The graph cut technique is used to solve the segmentation problem by using normalized cut method. K-means is a clustering algorithm, which partitions a data set into clusters according to some defined distance measure. The Linde, Buzo, and Gray (LBG) Algorithm, is an iterativealgorithm which requires an initial codebook to start with. Codebook is generated using a training set of images. By combing SIFT and segmentation method the existence of object is recognized.

## II. KEYPOINTS DETECTION USING SIFT

### 2.1 Keypoint Detection

The SIFT algorithm was initially presented in a paper by David Lowe in 1999 and then he summarized his algorithm in 2004.Keypoints are detected by using the DoG image $D(x; y;\sigma)$ which is the difference of smoothed images $L(x; y;\sigma)$. $L(x; y;\sigma)$ is obtained from the convolution of variable scale Gaussian with the input image $I(x; y)$.

$$D(x; y;\sigma) = (G(x; y; k\sigma) - G(x; y;\sigma)) * I(x; y)$$

$$D(x; y;\sigma) = L(x; y; k\sigma) - L(x; y; \sigma) \quad (1)$$

where,

$L(x; y; \sigma) = G(x; y; \sigma) * I(x; y)$ or $L(x; y; \sigma) = \frac{1}{2\pi\sigma^2}\exp(-\frac{x^2+y^2}{2\sigma^2}) * I(x; y)$

It is performed between different scales$\sigma$, and number of DoG images is obtained.

### 2.2 Keypoint Descriptor

The gradient magnitude $m(x; y)$ and orientation $\theta(x; y)$ at each pixel of the smoothed image at the key points are detected& are calculated using the following expressions:

$$m(x; y) = \sqrt{fx(x; y)^2 + fy(x; y)^2} \quad (3)$$

$$\theta(x; y) = \tan^{-1}\frac{fy(x; y)}{fx(x; y} \quad (4)$$

$$fx(x; y) = L(x + 1; y) - L(x - 1; y) \quad (2a)$$

$$fy(x; y) = L(x; y + 1) - L(x; y - 1) \quad (2b)$$

The keypoints orientation achieves rotation invariance, while magnitude for all levels useful in descriptor computation. Multiple orientations assigned to keypoints significantly improve stability of matching.

## III. SEGMENTATION

Image segmentation is an important image processing technique which is useful everywhere whenever we want to analyze what is inside an image. We need image segmentation technique to separate the objects in the image and analyze each object individually. Also, we can identify the diseases in medical imaging, in many applications like face detection, iris detection, fingerprint recognition. The goal of segmentation is to change the representation of an image into something more meaningful and easier to analyze.

### 3.1 Image Segmentation by Normalized Graphcut Method

#### 3.1.1 Graphcut

A graph cut is the process of partitioning a directed or undirected graph into disjoint sets. In the graph cut technique we represent the image in the form of graphs. The set of points in feature space is presented as a weighted graph G=(V,E).

Where,

V: graph nodes (Pixels),

E: edges between connection nodes (Pixel similarity).

A graph G = (V, E) is partitioned into two disjoint complementary sets A and B, B=V-A, by removing the edges connecting two parts. Weight of an edge can be calculated as the similarities between two nodes in a graph so, if there are no similarities in between two nodes then we can cut that edge this called graph cut. That closely relates to a mathematical formulation of a cut:

$$cut\,(A, B)=\sum_{u\epsilon A, v\epsilon B} w(u, v) \quad (3)$$

If we use the minimum cut method to solve the segmentation problem, we didn't get the better segmented image. Because it cuts all the pixels even if there are similarities in the image. So here we go for the normalized cut method.

**Sumer Singh,** Associate Professor, Department of ECE, BIET Sikar, Rajasthan, India

3.1.2 Normalized Graphcut

Shi and Malik propose a modified cost function i.e. normalized cut, to overcome discontinuities in case of the minimum cut technique. The normalised cut is defined as: "a fraction of the total edge connections to all the nodes in the graph":

$N_{cut}(A, B) = \frac{cut(A,B)}{asso\ (A,V)} + \frac{cut(A,B)}{asso\ (B,V)}$, asso (A,V)=$\sum_{u\epsilon A, t\epsilon V} w(u,t)$

Where,

cut (A, B) is Sum of all the edge weights associated with the cut,

asso (B, V) is sum of all the edge weights associated with the cut and all the points in the graph, and

asso (A, V) is the total connection from nodes A to all nodes in the graph.

Assume now we want to separate an image V with size M-by-N into two parts, we need to define two matrices: Wand D, both of size (MN)-by-(MN). With these two matrices, finding the minimum normalized cut of image V into two parts A and B is equal to solve the equation as followed:

$$\min N_{cut} = \min_y \frac{y^T(D-W)y}{y^T Dy} \qquad (4)$$

Equation (7) could be further simplified into a general eigenvector problem as followed:

$$(D - W)\ y = \lambda Dy \qquad (5)$$

Where,

D is diagonal matrix and each diagonal element di contains the sum of all the elements in the $i^{th}$ row in W.

W is the similarity matrix with element $w_{i,j}$ as the similarity between the $i^{th}$ pixel and the $j^{th}$ pixel and

y is an (MN)-by-1 vector with each element indicating the attribute of each pixel into the two groups.

The eigenvector y with the second smallest eigenvalue is selected for imagesegmentation i.e. eigenvector ywith eigenvalues can be used to separate pixels into two groups (partition).So, the segmentation algorithm consists of building the affinity matrix W, computing the eigenvectors and eigenvalues of the system, and thresholding a number of these eigenvectors to obtain partitions of the image. The intersection of these partitions yields the final segmentation. By taking any part of the above divided image, can get a more segmented image.

<u>Algorithmic steps for normalized graphcut</u>

1. Given an image or image sequence, set up a weighted graph G=(V,E) and set the weight on the edge connecting two nodes to be a measure of the similarity between the two nodes.
2. Solve (D-W) y = $\lambda$Dy for eigenvectors with the smallest eigenvalues.
3. Use the eigenvectors with the second smallest eigenvalues to bipartition the graph.
4. Decide if current partition should be subdivided and recursively repartition the segmented parts if necessary.

3.2 Image Segmentation by K-Means Algorithm

k-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) that are known as priori.

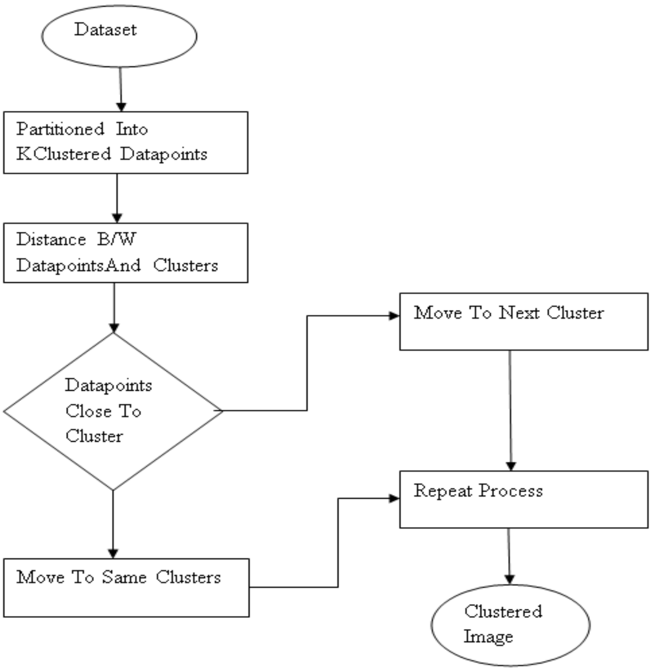


***Fig. 3.2: flow chart for k-means algorithm***

The main idea is to define k centers, one for each cluster. These centers 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 center. When no point is pending, the first step is completed. At this point we need to re-calculate k new centroids as barycenter of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center. A loop has been generated. As a result, centers do not move any more. Finally, this algorithm aims at minimizing an objective function knows as squared error function given by:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} (\|x_i - v_j\|)^2$$

where,

'$\|x_i - v_j\|$' is the Euclidean distance between $x_i$ and $v_j$.

'$c_i$' is the number of data points in $i^{th}$ cluster, and '$c$' is the number of cluster centers.

<u>Algorithmic steps for k-means clustering</u>

Let X = {$x_1$,$x_2$,$x_3$,……..,$x_n$} be the set of data points and V = {$v_1$,$v_2$,…….,$v_c$} be the set of centers.

1) Randomly select '$c$' cluster centers.
2) Calculate the distance between each data point and cluster centers.
3) Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers.
4) Recalculate the new cluster center using:

$$v_i = (1/c_i) \sum_{j=1}^{c_i} x_i$$

where, '$c_i$' represents the number of data points in $i^{th}$ cluster.

5) Recalculate the distance between each data point and new obtained cluster centers.
6) If no data point was reassigned then stop, otherwise repeat from step 3.

3.3 Image Segmentation by LBG Algorithm

The Linde–Buzo–Gray algorithm is introduced by Yoseph Linde, Andrés Buzo and Robert M. Gray. It is similar to the k means method in data clustering.This (LBG) algorithm is an iterative algorithm which requires an initial codebook to start with. Codebook is generated using a training set of images.So first, the training vector space is created and the centroid is obtained. The centroid is considered as the first codevector. Now, constant error is added to the codevector and two new vectors are obtained (fig. 6.3 shows the LBG clustering).v1 and v2 are the new generated codevectors. By using Euclidean distance two clusters are obtained in the first iteration. In the next iteration, this procedure is repeated for both the clusters.

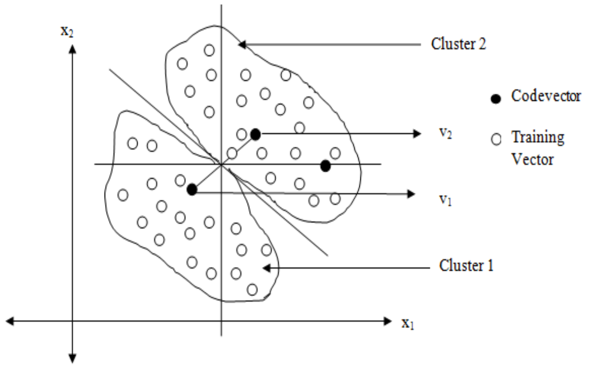


*Fig. 3.3.1: LBG clustering*

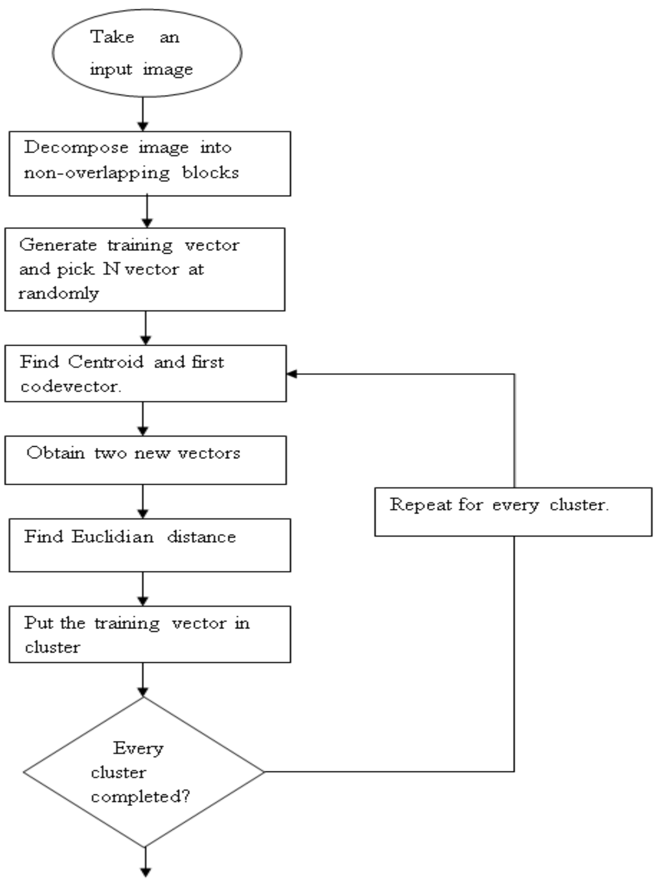


*Fig. 3.3.2: flowchart for lbg algorithm*

Algorithmic steps for LBG

1. Generate the training vector space, T of the image which contains M training vectors.

$T = \{X_1, X_2, X_3 \ldots X_M\}$

Xi is the training vector which is represented as $X_i = \{x_{i1}, x_{i2}, x_{i3}, \ldots x_{ik}\}$

where,

k denotes the dimension.

2. Find Centroid, C of the training vector space by taking the average of each column. This centroid is the first codevector.

$C = \{C_1, C_2, C_3 \ldots C_M\}$

3. Two new vectors are obtained after adding constant error, E to the codevector.

C1=C+E and C2=C-E

4. Find the Euclidian distance of the training vector space with these two vectors.

$D(X_i - C_j) = \sum_{p=1}^{k}(X_{ip} - C_{jp})^2$

where,

$X_i$ is the training vector,

$C_j$ is the codevector

5. Put the training vector in first cluster if the Euclidian distance between the training vector and the codevector C1 is less else put the training vector in the second cluster.

6. Repeat the steps 2 to 5 for every cluster.

7. Stop when desired codebook size is obtained.

## IV. OBJECT RECOGNITION

Here, object is recognized using SIFT by matching process.

Matching process:-

First of all, thedatabase is created by extractingSIFT keypoints frommodel image.Second, SIFT keypoints are extracted fromthe input image.Match keypoints to a database by using nearest neighbour distance ratio:

NNDR $= \frac{d_1}{d_2}$

$d_1, d_2$ : Distances to the 1st nearest and 2nd nearest neighbours.(Here, measure difference as Euclidean distance between feature vectors: d(u, v) = $\left(\sum_i (u_i - v_i)^2\right)^{1/2}$ ).

If NNDR is small, nearest neighbour is a good match.

When Keypoints matches, the system recognizes the existence of object by using SIFT.

## V. CONCLUSION

This system proposes the method of both object recognition and segmentation. For segmentation three algorithms are used and these are normalized graph cut, k-means, and LBG algorithm. Normalized graph cut method segments the image by using eigenvalue and eigenvector while k-means and LBG algorithm segments the image by clustering method. After segmenting the image by above algorithms object is recognized by SIFT using matching process. Scale Invariant Feature Transform (SIFT) is used to extract the features of image by finding and Describing Keypoints.However, if there are few keypoints, the accuracy of recognition and segmentation will fall down. And the computing time increases when the number of models increases. We will consider about these problems in the future.

ACKNOWLEDGMENTS

I would like to thank Dr. N. A. Charniya of EXTC Department at BabasahebNaik College of Engineering, Pusad for their valuable guidance and support.

## REFERENCES

[1]D. G. Lowe, “Object recognition from local scale invariant features”, *International Conference on ComputerVision, Corfu, Greece, pp. 1150-1157, 1999.*

[2]D. G. Lowe, “Distinctive image features from scale invariant keypoints,” *Journal of Computer Vision, 60, 2, pp.91-110, 2004.*

[3]Y. Boykov, M.P.Jolly, “Interactive graph cuts for optimal boundary & region segmentation of objects in N-D image,”*IEEE International Conference on Computer Vision and Pattern Recognition, volume 2, pp. 731-738, 2004.*

[4]Shi. J, and Malik. J, "Normalized cuts and image segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Computer Society, voI.22(8), 2000, pp.888-905.*

[5]A.Suga, K. Fukuda, T. Takiguchi, Y.Ariki, “Object Recognition and Segmentation Using SIFT and Graph Cuts,” *In 19th International Conference on Pattern Recognition, pp. 1-4, 2008.*

[6]A. Kumar Pal and A. Sar, “An Efficient Codebook Initialization Approach For LBG Algorithm,” *International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.1, No.4, August 2011.*

[7]Ms. ChinkiChandhok, Mrs. SoniChaturvedi, Dr. A. A. Khurshid, “An Approach to Image Segmentation using K-means Clustering Algorithm,”*International Journal of Information Technology (IJIT), Volume 1, Issue 1, August 2012.*

[8]H. Borhanifar and V. M. Naeim, “High Speed Object Recognition Based on SIFT Algorithm,” *International Conference on Image, Vision and Computing (ICIVC) 2012.*

[9]Saiful Islam, Dr.Majidul Ahmed, “Implementation of Image Segmentation for Natural Images using Clustering Methods,” *International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 3, March 2013*.