IMAGE SEGMENTATION USING NORMALISED GRAPH CUT METHOD

NICKY NIRLIPTA SAHOO
ROLLNO - EE19S042

[1] DEPARTMENT OF ELECTRICAL ENGINEERING,
INDIAN INSTITUTE OF TECHNOLOGY,MADRAS

Abstract. *Normalized Graph cut method was proposed by Jianbo Shi and Jitendra Malik to address the problem of perceptual grouping and organization in vision. Proposed method has provided a powerful way of extracting and understanding of the global impression of the image rather than focusing heavily on the local features and their consistencies in the image data. In this approach image is considered as a Graph of image nodes and apply the solution in the way of graph partitioning problem. This method has really address the drawbacks of existing image segmentation method of using minimum cut. It has provided the new measure of graph partitioning called disassociation measure the normalized cuts by taking total edge connection to all the nodes in the graph to compute the cut cost. Normalized cut method is taking both total dissimilarity between the different groups and total similarities within a group for the graph partitioning process. To optimise this problem generalised eigen value problem has also been implemented.*

$$keywords - Normalisedgraphcut, generalisedeigenvalueproblem, partitioning. \tag{1}$$

1. INTRODUCTION

Image Processing is becoming paramount important technology to the modern world since it is the caliber behind the machine learning and artificial intelligence. Image segmentation is one of the major area of the modern image processing and computer vision. Many computer vision researches that have been carrying out emerge the important of pattern analysis and perceptual vision of a scene and regression of features. Image segmentation is inherent strength of image processing techniques for pattern recognition and regression. This paper basically based on image partitioning in hierarchically downward manner.The hierarchical divisive approach produces a tree ,the dendogram.

1.1. GRAPH AND GRAPH PARTITIONING

A Graph G=(V,E) can be partitioned in to two disjoint set A and B ,where $A \cup B = V$ and $A = \phi$,by removing the edges connecting the two graph.The total dissimilarity between the two parts can be represented by total number of edges being removed,which is known as cut and mathematically can be represented as,

$$cut(A, B) = \sum_{u \in A, v \in B} \mathrm{w}(u, v) \tag{2}$$

Optimal Bi partitioning the graph is the one that minimises the cut value.

1.1.1. MIN CUT

Graph partitioning absorb the global perception of an image rather than focusing on local properties of the image.Wu and Leahy [3] propsed clustering based minimum cut criterion in which the graph is being partitioned into k subgraphs , such that maximum cut across the sub group is minimized.This can be done by recursively finding the minimum cuts and bisect the existing segments.This method uses total edge weight connecting the two partition and no measure or parameter is used to indicate the number of edge connection as a fraction of total connection to all the nodes in the graph. As a result it favors cutting small set of isolated nodes in to graphs and gives bad partitioning when cost (weight) function is inverse to the distance (Similarity) between nodes .

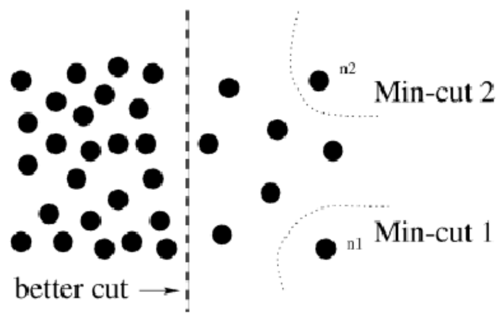

Figura 1. Example of minimum cut giving bad partition

Therefore the problem remains with the isolated point in an image segmentation and clustering when we use minimum cuts as shown in fig(1). Requirement for the novel method with a new measure of association of graph node is accomplished by the normalized cut technique.

1.1.2. NORMALISED CUT

The problem of partitioning small cuts can be resolved by introducing a measure of dissociation between parts.Instead of considering only the value of edge weights connecting two partition, cut cost is computed as a fraction of total edge connection to all nodes in the graph and it is known as Normalised cut (Ncut),

$$N_{cut(A,B)} = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B}{assoc(B,V)} \tag{3}$$

where $assoc(A,V) = \sum_{u \in A, t \in V} \mathrm{w}(u,t)$ is the total number of connections from node A to all nodes in the graph. similarly,$assoc(B,V) = \sum_{u \in B, t \in V} \mathrm{w}(u,t)$ is the total number of connections from node B to all nodes in the graph. Using this dissociation between groups,the cut value with small isolated points will have large Ncut value, as the cut value is large percentage of total connections fro small set to all other nodes. The similarity measure for total Normalised association within the group can be represented as,

$$N_{assoc(A,B)} = \frac{assoc(A,A)}{assoc(A,V)} + \frac{assoc(B,B}{assoc(B,V)} \tag{4}$$

where $assoc(A,B) and assoc(B,B)$ is the total weights of edges within A and B respectively.This reflects how tightly the nodes within groups are connected to each other. Association and dissociation are related as,

$$N_{cut(A,B)} = 2 - N_{assoc(A,B)} \tag{5}$$

Hence,this graph partitioning algorithm minimises the dissociation between the groups and maximizes association within groups.

1.1.3. GENERALISED EIGEN VALUE PROBLEM

To optimise the N_{cut} value eigen vectors corresponding to smallest eigen values of generalised eigen vector is being considered to bi partition the graphs.Rather than solving standard eigen vectors,Genralised eigen vectors give more efficient results.

2. METHODOLOGY

2.1. PARTITION AND GENERALISED EIGEN VALUE PROBLEM

Let the graph be given as G=(V,E),where V is the no of nodes and E is the i.e connection between individual nodes.The weight of each edge is represented by w(i,j) i.e for edge from node i to node j.
Total no of connections from node i to all other nodes is represented as,

$$d(i) = \sum_j \mathrm{w}(i,j) \tag{6}$$

D be the diagonal matrix with d(i,j) on its diagonal.
W be a sparse symmetric matrix, of size $N \times N$ containing the weights of the edges,where $N = noofrowsofimage \times noofcolumnsoftheimage$ Then mathematical expression for optimal Normalised cut partitioning in to A and B is given by,

$$min_(A,B)Ncut(A,B) = min_y \frac{y^T(D-W)y}{y^T Dy} \tag{7}$$

where ,$y = (1+x) - b(1-x)$
x indicator vector with $xi = 1$, if node i is in A and
$xi = -1$, otherwise
$xi > 0$, represented by a vector $\frac{1+x}{2}$ and
$xi < 0$, represented by a vector $\frac{1-x}{2}$
$k = \frac{\sum_{x_i>0} d_i}{\sum_i d_i}$
$b = \frac{k}{1-k}$
And if y is relaxed to take real values ,then eqn(3) can be minimised by solving generalised eigen value system

$$(D-W)y = \lambda Dy \tag{8}$$

2.2. DESIGN STEPS

Step1 : Given image I[nrow,ncol],N=total no of pixels $(nrow \times ncol)$ Compute the graph G=(V,E),where v,the node of the graphs is no of pixels in the image and E is the connecting edges. Step 2 : The weight of the edges depend on the feature similarity and spatial proximity .The mathematical expression for computing weight is,

$$w_{ij} = \exp \frac{-(\|F(i) - F(j)\|_2)^2}{\sigma_I} \times \begin{cases} \exp \frac{-(\|X(i) - X(j)\|_2)^2}{\sigma_X} & if(\|X(i) - X(j)\|_2 < r) \end{cases} \tag{9}$$

Where X(i) is the spatial location of node I and F(i) is the feature vector which is different for different kind of segmentation.

- F(i) =1 segmentation point set
- F(i) =I(i)the intensity value for segmentation brightness (gray scale images)
- F(i)=$[v, v.s. \sin h, v.s. \cos h](i)$ where h,s,v are the HSV values for color segmentation.
- $F(i) = [|I * (f_1)|, ..., |I * (f_n)|](i)$ where f_n is the $DOOGfilter$.

The weight matrix is being stored in a sparse matrix W of $N/timesN$ where N =no of rows$/times$ no of cols
Step 3 : Let d(i) be the total no of connections of pixel i represented as $d(i) = \sum_j \mathrm{w}(i, j)$ then compute the
diagonal matrix D of dimmension $N * N$ whose diagonal elements are d(i).
Step 4 : Solve for $(D-W)y = \lambda$ Dy to obtain the eigen vector with smallest eigen values.
Step 5 : Bi-Partitioning Graph : Eigen vector corresponding to second smallest value has been taken to Bi partition the segment in to two part .In ideal case of bi partitioning ,eigen vector takes only two values or negative and segmentation is being done with respect to zero. But in reality as eigen vectors can take continuous value so splitting point is being consider to partition the segment.Splitting point is being considered among the minimum of mean of the eigen vector value and Normalised cut values.The Normalised cut value can be estimated by using the formula,

$$min_(A, B)Ncut(A, B) = min_y \frac{y^T(D - W)y}{y^T Dy} \tag{10}$$

Step 6 : Recursive Re partition : Bi partition is being repeated till some criteria being satisfied. Threshold value for minimum ncut value,minimum area, and number of iteration is being given by the user. Stability Criterion can be considered to overcome situation with more number of splitting points with similar Ncut values. In this case histogram of the eigen vector is being calculated and the ratio of min to max value of histogram is computed.If the ratio is high means splitting is means eigen vectors vary continuously and histogram has similar value.

3. EXPERIMENTAL RESULTS

The above algorithm is implemented on Berkeley segmentation data set.Here I have used MATLAB software of version 2019A to implement the above algorithm.The results are as follows.

3.1. Image segmentation on BSD data set from 1-25

For BSD data set fro 1-25 the results are as follows,

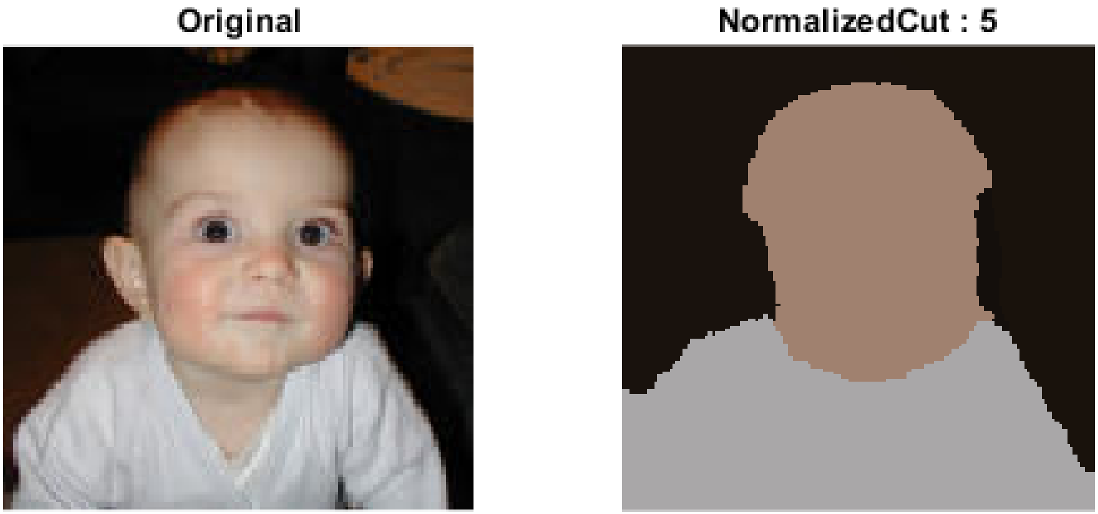

Figura 2. Original image and its segmentation map

Figura 3. Segmented Images

In this segmentation the parameters are tuned as,SI=5 ;SX=6 ;r=1.5 ;SNcut=120 ;SNarea=1.5 ;Size of the image is 130×130 And the result is ,Elapsed time=18.307333 seconds. ;ROOT='ROOT-A-A ncut-A n...' 'ROOT-A-A ncut-B n...' 'ROOT-A-A ncut-B n...' 'ROOT-A-B ncut' 'ROOT-B';Splitting Points=[0.0763] [0.0129] [0.1596] [0.0276] [0.0144]

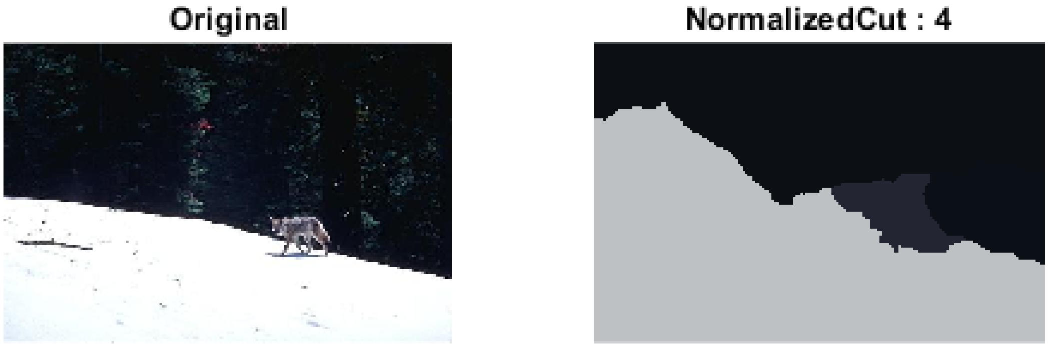

Figura 4. Original image and its segmentation map

Figura 5. Segmented Images

SI=9 ;SX=6 ;r=1.5;SNcut=120 ;SNarea=120;Elapsed Time=131.025seconds ;ROOT='ROOT-A-A ncut-A ncut' 'ROOT-A-A ncut-B ncut' 'ROOT-A-B ncut' 'ROOT-B'; Splitting Points=[0.0286] [0.1704] [0.0273] [0.0080]

Bad-segmentation results:

Figura 6. Original image and its segmentation map

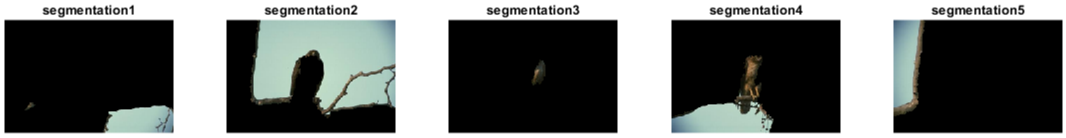

Figura 7. Segmented Images

SI=6 ;SX= 9 ;r= 2.2 ;SNcut=0.21 ; SNarea=240 ;Elapsed Time=133.452 seconds. ;ROOT='ROOT-A-A ncut' 'ROOT-A-B ncut-A ncut' 'ROOT-A-B ncut-B ncut' 'ROOT-B';Splitting Points =[1.59e-14] [0.0112] [0.0110] [0.0216]

Figura 8. Original image and its segmentation map

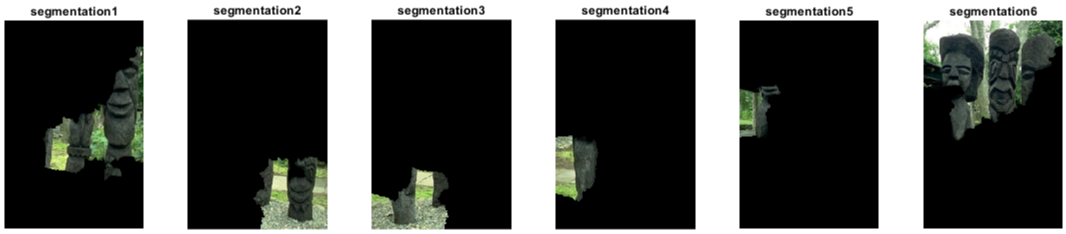

Figura 9. Original image and its segmentation map

SI=6 ;SX=9 ;r=1.5 ;SNcut= ;SNarea= ;Elapsed Time=136.383658 seconds ;ROOT=’ROOT-A-A ncut-A n...’ ’ROOT-A-A ncut-A n...’ ’ROOT-A-A ncut-A n...’ ’ROOT-A-A ncut-B n...’ ’ROOT-A-B ncut”ROOT-B’;Splitting Points=[0.0234] [0.0595] [0.0390] [0.1072] [0.0368] [0.0265];

3.2. Image segmentation on BSD data set from 25-50

For BSD data set fro 1-25 the results are as follows,

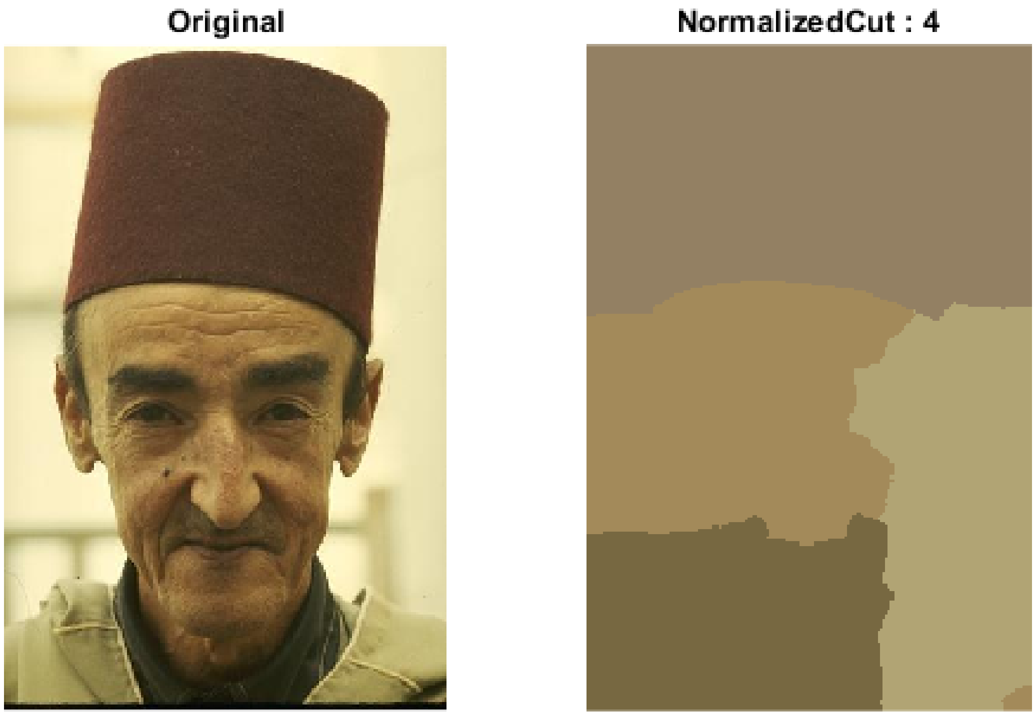

Figura 10. Original image and its segmentation map

Figura 11. Segmented Images

SI=9 ;SX=5 ;r=2 ;SNcut=0.21 ;SNarea= 220 ;Elapsed Time=119.767689 seconds ;ROOT='ROOT-A-A ncut' 'ROOT-A-B ncut-A ncut' 'ROOT-A-B ncut-B ncut' 'ROOT-B';Splitting Points =[1.5289e-17] [0.0162] [0.0310] [0.0236]

Figura 12. Original image and its segmentation map

SI=8 ;SX=5 ;r=1.5 ;SNcut=0.21 ;SNarea=320 ;Elapsed Time=132.074918 seconds ;ROOT='ROOT-A-A ncut-A ncut' 'ROOT-A-A ncut-B ncut' 'ROOT-A-B ncut' 'ROOT-B';Splitting Points= [0.0206] [0.0306] [0.0126] [0.0147]

Figura 13. Segmented Images

3.3. Image segmentation on BSD data set from 51-75

For BSD data set from 51-75 grey images of size 161×241 (resizing by 0.5) the results are as follows,

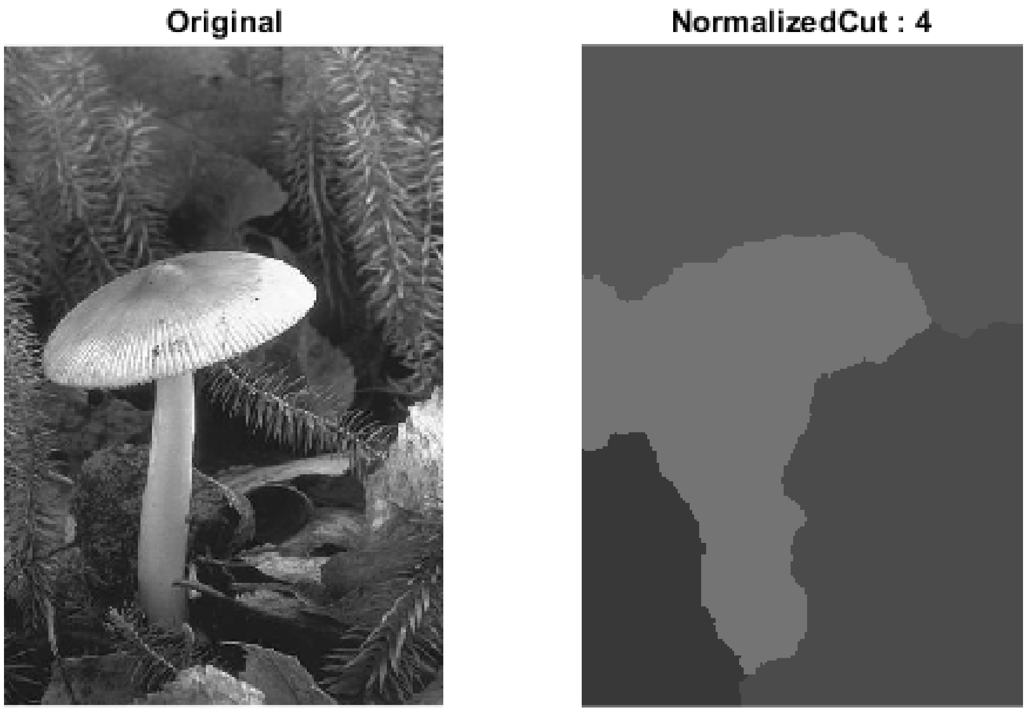

Figura 14. Original image and its segmentation map

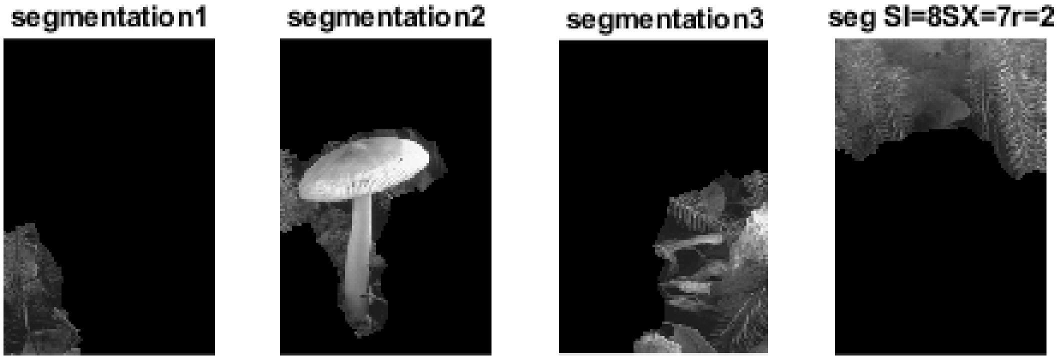

Figura 15. Segmented Images

SI=8 ;SX= 7 ;r=2 ;SNcut=0.21 ;SNarea=320 ;Elapsed Time= 132.022989 seconds ;ROOT='ROOT-A-A ncut-A ncut' 'ROOT-A-A ncut-B ncut' 'ROOT-A-B ncut' 'ROOT-B';Splitting points=[0.0225] [0.0247] [0.0231] [0.0107]

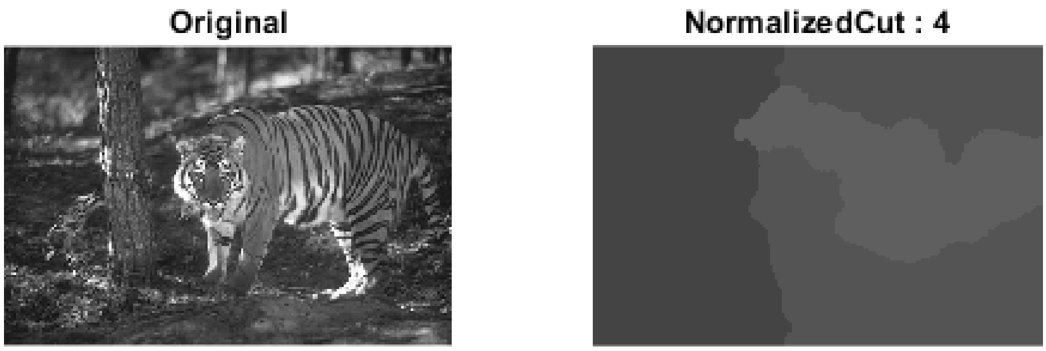

Figura 16. Original image and its segmentation map

Figura 17. Segmented Images

SI=9 ;SX=7 ;r= 2 ;SNcut= 0.21 ;SNarea=320 ; Elapsed Time= 139.742568 seconds ;ROOT= 'ROOT-A-A ncut-A ncut' 'ROOT-A-A ncut-B ncut' 'ROOT-A-B ncut' 'ROOT-B';Splitting Points= [0.0148] [0.0216] [0.0084] [0.0109] ;

4. CONCLUSION

The segmentation of an image is being done using Normalised cuts based image segmentation.Considering only smallest eigen vectors has reduced the computational complexity and also use of generalised eigen vector also has reduce the computation time over standard eigen solver and gives efficient splitting points to partition the graph.Stability criterion is being implemented to overcome the situation with more number of splitting points having similar values. The threshold value for SNarea and r depends largely on the size of image.

5. FUTURE WORK

1 Here I have done the segmentation using Recursive 2 way N-cut which is not that efficient for oscillatory eigen vectors.Rather k-way cut with multiple eigen vector can be implemented using Greedy punning and Global recursive cut to efficiently segment oscillatory eigen vectors.
2 $DOOG$ filter to be designed and implemented for texture image segmentation.
3 Motion based segmentation is to be performed.
4 Optimal tunning for diffrent class and error calculation with respect to ground truth

6. REFERENCE*

[1] J. Shi and J. Malik,'Normalized Cuts and Image Segmentation' , IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 8, Aug 2000
[2] Berkeley SegmentationDataset.⟨https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images.html⟩
[3] Z. Wu and R. Leahy, 'An Optimal Graph Theoretic Approach to Data Clustering: Theory and Its Application to Image Segmentation',IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 15,no. 11, pp. 1,101-1,113, Nov. 1993