Survey on Image Segmentation using Graph Based Methods

Nalina.P, Muthukannan.K

Abstract—The main goal of this paper is to survey the high quality of image segmentation with improved speed and stability. In this paper to segment the image using three different graph based segmentation algorithms. These are Isoperimetric Segmentation Normalisd Cut Segmentation, and Spectral Segmentation. Apply these algorithms in the image and find out segmentation result. Using the segmentation results the performance will be analyzed with speed and stability. To determine stability of image by adding the Additive Noise, Multiplicative Noise, Shot Noise

Keywords—Isoperimetric, Normalized Cut, Performance Evaluation, Spectral, Segmentation.

I. INTRODUCTION

A. General Introduction

Every image is a set of pixel and partitioning those pixels on the basis of the similar characteristics. segmentation is dividing an image into sub partitions on the basis of some similar characteristics like color, intensity and texture is called image segmentation. The goal of segmentation is to change the representation of an image into something more meaningful and easier to analyze. Image segmentation is normally used to locate objects and boundaries that is lines, curves, etc. in images. Segmentation can be done by detecting edges or points or line in the image. When we detect the points in an image then on the basis of similarities between any two points we can make them into separate regions.

Among different segmentation schemes, graph based algorithm ones have several good features in practical applications. It is more flexible and computation more efficient. A lot of work has been done on graph theory in other applications, The merits of graph based method is re-use existing algorithms and theorems developed for other fields in image analysis. The graph based image segmentation is based on selecting edges from a graph, where each pixel corresponds to a node in the graph. Weights on each edge measure the dissimilarity between pixels. The segmentation algorithm defines the boundaries between regions by comparing two quantities – Intensity differences across the boundary and Intensity difference between neighboring pixels within each region. This is useful knowing that the intensity differences across the boundary Literature Review are important if they are large relative to the intensity differences inside at least one of the regions. Graph based image-segmentation is a fast and efficient method of generating a set of segments from an image.

The work of Shi and Malik, 1997 [9]; Presents Segmentation based on eigenvector-based methods these methods are too slow to be practical for many applications. In Ratan et al. (1999)[11], method described in this paper has been used in large-scale image database applications. It is fail to capture perceptually important non-local properties of an image. The Work of Urquhart, 1982; Zahn, 1971 [12&13] Presents Segmentation Based on Early graph-based methods. Main disadvantage is Fixed threshold &Local Measures in Computation. Pedro F. Felzenszwalb and Daniel P. Huttenlocher, 2004 [1] it works Based Krusal's Algoritham drawback of this paper is Low Variability image regions while ignoring detail in High variability regions..It is very difficult for users to choose an appropriate value for an expected segmented size. One reason for this interest is that the segmentation quality of Ncuts and other graph-based segmentation methods [2] is very good. The recently-developed isoperimetric method of graph partitioning [3] has demonstrated that quality partitions of a graph may be determined quickly and that the partitions are stable with respect to small changes in the graph (mask). Additionally, the same method was also applied to image segmentation, showing quality results [17].

II. PROPOSED METHOD

The proposed method of this paper is image segmentation done using three different algorithm segmentation. The original image segmented by Spectral segmentation, Normalised cut segmentation, Isoperimetric segmentation. Using these segmentation result the performance will be analyzed with speed and stability. From the performance evolution the three different result will compared and find out which algorithm is fast more stable.

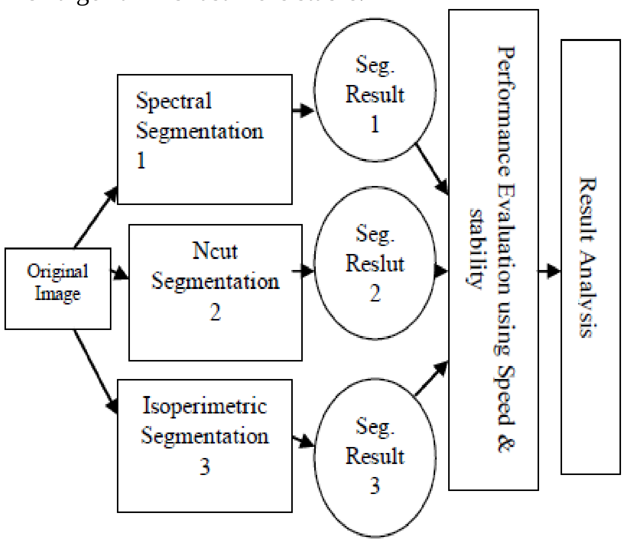


Figure 1. Block Diagram of Proposed Method

Manuscript received on January, 2013.
Nalina.P, Electronics and Communication Engineering, Einstein College Of Engineering, Tirunelveli, India.
Muthukannan.K, , Electronics and Communication Engineering, Einstein College Of Engineering, Tirunelveli, India.

A. *Isoperimetric Problem*

Graph partitioning has been strongly influenced by properties of a combinatorial formulation of the classic isoperimetric problem: For a fixed area, find the region with minimum perimeter.

Define the isoperimetric constant *h*,

$$h = \inf_{S} \frac{|\partial S|}{Vol_S}, \tag{1}$$

where *S* is a region in the manifold, Vol_S denotes the volume of region *S*, $|\partial S|$ is the area of the boundary of region S, and h is the infimum of the ratio over all possible *S*. For a compact manifold, $Vol_s = \frac{1}{2} Vol_{Total}$, and for a noncompact manifold, $Vol_S < 1$ (see [19]). We show in this paper that the set (and its complement) for which h takes a minimum value defines a good heuristic for data clustering and image segmentation. In other words, finding a region of an image that is simultaneously both large (i.e., high volume) and that shares a small perimeter with its surroundings (i.e., small boundary) is intuitively appealing as a "good" image segment. Therefore, we will proceed by defining the isoperimetric constant on a graph, proposing a new algorithm for approaching the sets that minimize *h*, and demonstrate applications to data clustering and image segmentation.

B. *The Isoperimetric Partitioning Algorithm*

A graph is a pair $G = (V;E)$ with vertices (nodes) $v \in V$ and edges $e \in E \subseteq V \times V$. An edge, *e*, spanning two vertices, v_i and v_j, is denoted by e_{ij}. Let $n = |V|$ and $m = |E|$ where | . | denotes cardinality. A weighted graph has a value (typically nonnegative and real) assigned to each edge called a weight. The weight of edge e_{ij}, is denoted by $w(e_{ij})$ or w_{ij}. Since weighted graphs are more general than unweighted graphs (i.e., $w(e_{ij}) = 1$ for all $e_{ij} \in E$ in the unweighted case), we will develop all our results for weighted graphs. The degree of a vertex v_i, denoted d_i is

$$d_i = \sum_{e_{ij}} w(e_{ij}) \quad \forall \quad e_{ij} \in E \tag{2}$$

For the Graph *G*, the Isoperimetric constant h_G is

$$h_G = \inf_{S} \frac{|\partial S|}{Vol_S}, \tag{3}$$

Where $S \subset V$ and

$$Vol_S \leq \frac{1}{2} Vol_V \tag{4}$$

$$|\partial S| \leq \sum_{e_{ij} \in \partial S} w(e_{ij}) \tag{5}$$

In order to determine a notion of volume for a graph, a metric must be defined. Different choices of a metric lead to different definitions of volume and even different definitions of a combinatorial Laplacian operator (see [19], [20]). Dodziuk suggested [21], [22] two different notions of combinatorial volume,

$$Vol_S = |S|, \tag{6}$$

and

$$Vol_s = \sum_i d_i \, \forall \, v_i \in S \tag{7}$$

One may view the difference between the definition of volume in (6) and that in (7) as the difference between what Shi and Malik term the "Average Cut" versus their "Normalized Cut" [11], although the isoperimetric ratio (with either definition of volume) corresponds to neither criterion. The matrix used in the Ncuts algorithm to find image segments corresponds to the combinatorial Laplacian matrix under the metric defined by (7). Traditional spectral partitioning [4] employs the same algorithm as Ncuts, except that it uses the combinatorial Laplacian matrix defined by the metric associated with (6). In agreement with [11], we find that the second metric (and hence, volume definition) is more suited for image segmentation since regions of uniform intensity are given preference over regions that simply possess a large number of pixels. Therefore, we will use Dodziuk's second metric definition and employ volume as defined in equation (7).

C. *Derivation Of Isoperimetric Algorithm*

Define an indicator vector, *x*, that takes a binary value at each node

$$x_i = \begin{cases} 0 & if \; v_i \in \bar{S}, \\ 1 & if \; v_i \in S. \end{cases} \tag{8}$$

Note that a specification of x may be considered a partition. Define the $n \times n$ matrix, *L*, of a graph as

$$L_{v_i v_j} = \begin{cases} d_i & if \; i = j, \\ -w(e_{ij}) & if \; e_{ij} \in E \\ 0 & Otherwise \end{cases} \tag{9}$$

The notation $L_{v_i v_j}$ is used to indicate that the matrix *L* is being indexed by vertices v_i and v_j. This matrix is also known as the **admittance matrix** in the context of circuit theory or the **Laplacian matrix** (see, [23] for a review) in the context of finite difference methods (and in the context of [21]).
By definition of *L*,

$$|\partial S| = x^T L x \tag{10}$$

and $Vol_s = X^T d$, where *d* is the vector of node degrees. If *r* indicates the vector of all ones, minimizing (10) subject to the constraint that the set, S, has fixed volume may be accomplished by asserting

$$Vol_s = X^T d = k, \tag{11}$$

where $0 < k < 1/2 r^T d$ is an arbitrary constant and *r* represents the vector of all ones. We shall see that the choice of k becomes irrelevant to the final formulation. Thus, the isoperimetric constant (3) of a graph, *G*, may be rewritten in terms of the indicator vector as

$$h_G = \min_{x} \frac{x^T L x}{x^T d}, \tag{12}$$

subject to (11). Given an indicator vector, x, then h(x) is used to denote the isoperimetric ratio associated with the partition specified by x. Note that the ratio given by (12) is different from both the "ratio cut" of [6], [7] and the "average cut" of [11]. Although the criterion in (12) rewards similar partitions to the normalized cut, average cut and ratio cut (i.e., large segments with small boundaries), what appears as a minor difference in the formulation allows us to use a solution to a system of linear equations instead of solving an eigenvector problem.

Note that the ratio cut technique of [6], [7] is distinct (in algorithm and pertinent ratio) from the ratio cut of [1], which applies only to planar graphs. The advantages of a system of linear equations over an eigenvector problem will be discussed below.

The constrained optimization of the isoperimetric ratio is made into a free variation via the introduction of a Lagrange multiplier [24] and relaxation of the binary definition of x to take nonnegative real values by minimizing the cost function

$$Q(x) = x^T Lx - \wedge (x^T d - k) \quad (13)$$

Since L is positive semi-definite (see, [25],) and $x^T d$ is nonnegative, $Q(x)$ will be at a minimum for any critical point. Differentiating $Q(x)$ with respect to x yields

$$\frac{dQ(x)}{dx} = 2Lx - \wedge d, \quad (14)$$

Thus, the problem of finding the x that minimizes $Q(x)$ (minimal partition) reduces to solving the linear System

$$2Lx = \wedge d, \quad (15)$$

Henceforth, we ignore the scalar multiplier 2 and the scalar $\wedge$ since, as we will see later, we are only concerned with the relative values of the solution.

Unfortunately, the matrix L is singular: all rows and columns sum to zero (i.e., the vector r spans its null space), so finding a unique solution to equation (15) requires an additional constraint.

We assume that the graph is connected, since the optimal partitions are clearly each connected component if the graph is disconnected $(i.e., h(x) = h_G = 0)$. Note that in general, a graph with c connected components will correspond to a matrix L with rank $(n-c)$ [25]. If we arbitrarily designate a node, v_g, to include in S $(i.e., fix\ x_g = 0)$, this is reflected in (15) by removing the gth row and column of L, denoted by L_0, and the gth row of x and d, denoted by x_0 and d_0, such that,

$$L_0 x_0 = d_0, \quad (16)$$

This is a nonsingular system of equations. Solving equation (16) for x_0 yields a real-valued solution that may be converted into a partition by setting a threshold (see below for a discussion of different methods). In order to generate a segmentation with more than two parts, the algorithm may be recursively applied to each partition separately, generating sub partitions and stopping the recursion if the isoperimetric ratio of the cut fails to meet a predetermined threshold. We term this predetermined threshold the **stop** parameter and note that since $0 \le h(x) \le 1$, the *stop* parameter should be in the interval (0,1). Since lower values of $h(x)$ correspond to more desirable partitions, a stringent value for the *stop* parameter is small, while a large value permits lower quality partitions (as measured by the isoperimetric ratio).

D. Choosing Edge Weights

In order to apply the isoperimetric algorithm to partition a graph, the position values (for data clustering) or the image values (for image segmentation) must be encoded on the graph via edge weights. We employ the standard [11], weighting function

$$w_{ij} = \exp\left(-\beta\left(I_i - I_j\right)^2\right) \quad (17)$$

where β represents a parameter we call **scale** and I_i indicates the intensity value at node v_i. Note that $(I_i - I_j)^2$ may be replaced by the squared norm of a Euclidean distance in the case of vector valued data or coordinates, in the case of a clustering problem. In order to make one choice of β applicable to a wide range of data sets, we have found it helpful to normalize the intensity differences for an image before applying (17).

E. Algorithm Steps

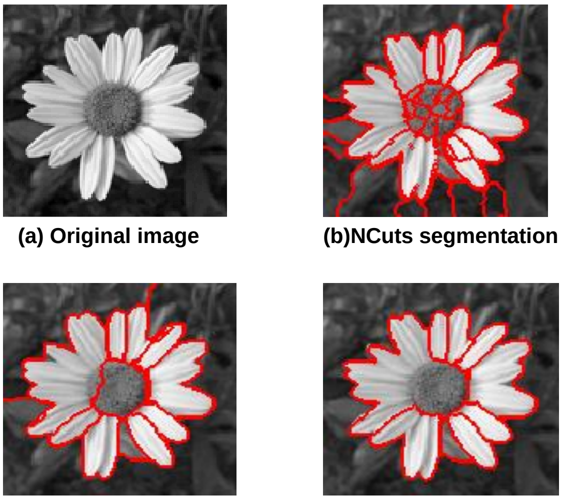
The isoperimetric algorithm is controlled by only two parameters: the *scale* parameter β of equation (17) and the *stop* parameter used to end the recursion. The scale affects how sensitive the algorithm is to changes in feature space (e.g., RGB, intensity), while the stop parameter determines the maximum acceptable isoperimetric ratio a partition must generate in order to accept it and continue the recursion.

1) Initialize the *Stop* and *Scale* Parameter.
2) Find weights for all edges using equation (17).
3) Build the L matrix (9) and $d = diag(L)$ vector.
4) Choose the node of largest degree as the ground node, v_g, and determine L_0 and d_0 by eliminating the row/column corresponding to v_g.
5) Solve equation (16) for x_0.
6) Threshold the potentials x at the value that gives partitions corresponding to the lowest isoperimetric ratio.
7) Continue recursion on each segment until the isoperimetric ratio of the sub partitions is larger than the *stop* parameter.
8) Perform Isoperimetric, Spectral, Ncuts Segmentation algorithm.
9) Stability analysis relative to additive, multiplicative and shot noise.

III. EXPERIMENTAL RESULTS

It contain to main parts these are segmentation analysis and stability analysis. In the analysis four different images are used to segmentation. Apply the three algorithms to that four images checking the segmentation result. Adding the three different noises to images and determine the stability.

A. Segmentation Analysis



(a) Original image (b)NCuts segmentation

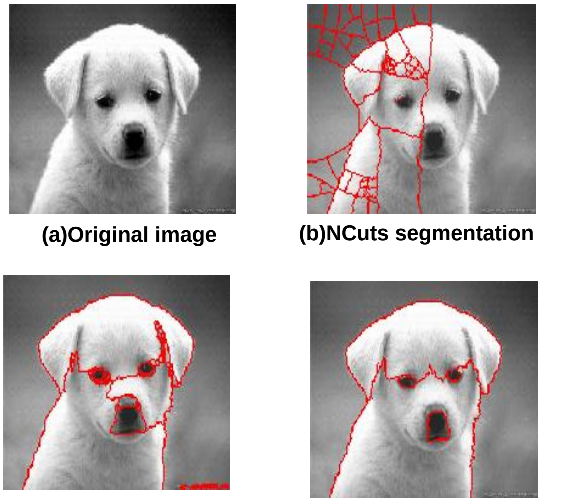
(c)Spectral segmentation (d)Isoperimetric segmentation

Fig 2. Segmentation Result of Flower image

segmentations result produced by the isoperimetric algorithm using the parameters ($\beta = 95$, $stop = 10^{-5}$) and Ncuts algorithm using the parameters ($\beta = 35$, $stop = 10^{-2}$).

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

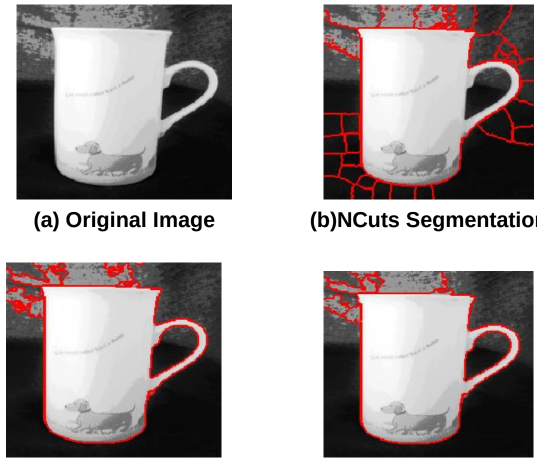
From the Fig.6. x-axis represents an increasing noise variance for the additive and multiplicative noise, and an increasing number of "shots" for the shot noise. The y-axis indicated the number of segments found by each algorithm. The solid line represents the results of the isoperimetric algorithm and the dashed line represents the results of the Ncuts algorithm. The underlying graph topology was the 4-connected lattice with $\beta = 95$ for the isoperimetric algorithm and $\beta = 35$ for the Ncuts algorithm. Ncuts stop criterion = 10^{-2} (relative to the Ncuts criterion) and isoperimetric stop criterion = 10^{-5}. In all cases, the isoperimetric algorithm outperforms Ncuts, most dramatically in response to shot noise. The β and stop values for each algorithm were chosen empirically to produce the best results for that algorithm in response to noise.



(a)Original image (b)NCuts segmentation

(c)Spectral segmentation (c)Isoperimetric segmentation

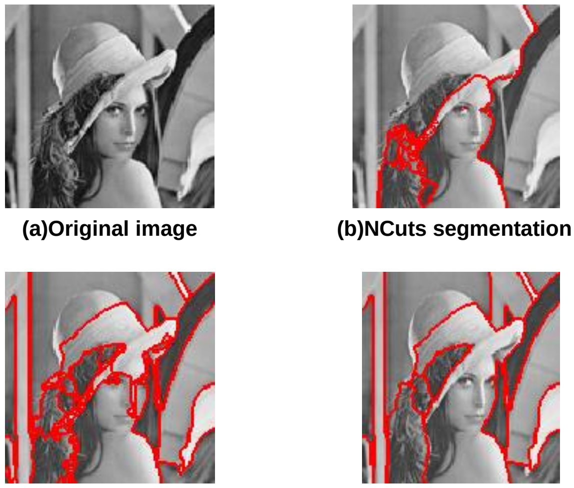
Fig 3. Segmentation Result of Dog image



(a) Original Image (b)NCuts Segmentation

(c)Spectral segmentation (d)Isoperimetric segmentation

Fig 4. Segmentation Result of Mug image



(a)Original image (b)NCuts segmentation

(c)Spectral segmentation (d)Isoperimetric segmentation

Fig 5. Segmentation Result of Lenna mage

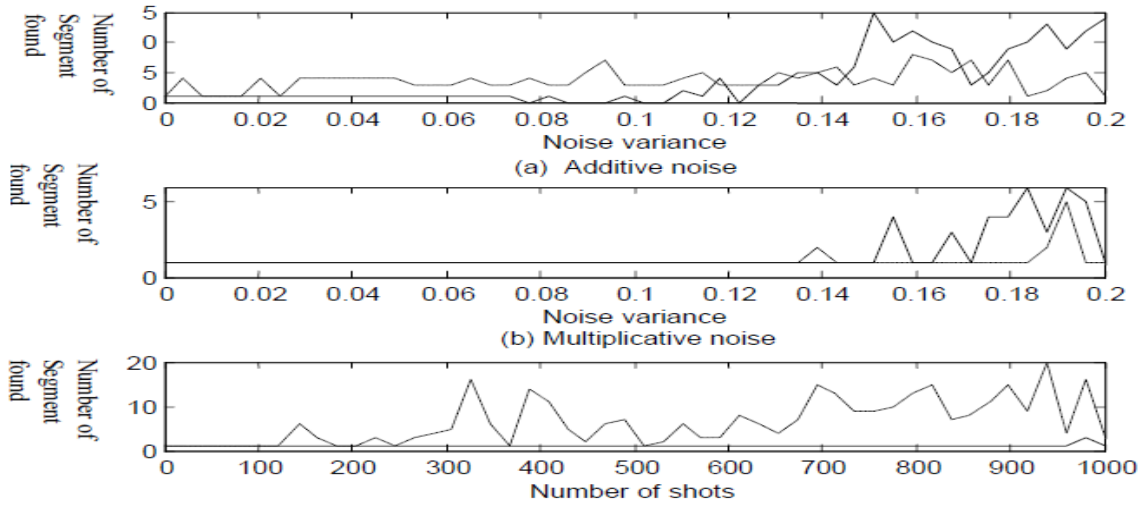


Fig.6. Stability Analysis Relative To Additive, Multiplicative And Shot Noise

IV. PERFORMANCE EVALUATION

A. Stability Analysis

Fig.7 Stability Analysis Result Relative To Additive, Multiplicative And Shot Noise For Mug Image.

The sensitivity of Ncuts (our implementation) and the isoperimetric algorithm to noise is compared using a quantitative and qualitative measure. First, each algorithm was applied to an artificial image, using a 4-connected lattice topology. Increasing amounts of additive, multiplicative and shot noise were applied, and the number of segments output by each algorithm was recorded. Results of this comparison are recorded in Fig.6. In order to visually compare the result of the segmentation algorithms applied to progressively noisier images, the isoperimetric and Ncuts algorithms were applied to a relatively simple Mug image. The isoperimetric algorithm operated on a 4-connected lattice, while Ncuts was applied to an 8-connected lattice, since we had difficulty finding parameters that would cause Ncuts to give a good segmentation of the original image if a 4-connected lattice was used.

In Fig.7.Each row represents an increasing amount of noise of the appropriate type. The top row in each subfigure is the segmentation found for the flower.tif image. Each figure is divided into three columns representing the image with noise, isoperimetric segmentation and Ncuts segmentation from left to right respectively. The underlying graph topology was the 4-connected lattice for isoperimetric segmentation and an 8-connected lattice for Ncuts segmentation (due to failure to obtain quality results with a 4-connected lattice) with $\beta = 95$ for the isoperimetric algorithm and $\beta = 35$ for the Ncuts algorithm. Ncuts *stop* criterion $= 5 \times 10^{-2}$ (relative to the Ncuts criterion) and isoperimetric *stop* criterion $= 10^{-5}$. Results were slightly better for additive noise, and markedly better for multiplicative and shot noise. Note that the β and *stop* values for each algorithm were chosen empirically to produce the best results for that algorithm in response to noise

(a) Additive noise (b) Multiplicative noise. (c) Shot noise.

In both comparisons, additive, multiplicative, and shot noise were used to test the sensitivity of the two algorithms to noise. The additive noise was zero mean Gaussian noise with variance ranging from 1-20% of the brightest luminance. Multiplicative noise was introduced by multiplying each pixel by a unit mean Gaussian variable with the same variance range as above. Shot noise was added to the image by randomly selecting pixels that were fixed to white. The number of .shots. ranged from 10 to 1,000.The above discussion of stability is illustrated by the comparison in Fig.6. Although additive and multiplicative noise heavily degrades the solution found the Ncuts algorithms, the isoperimetric algorithm degrades more gracefully. Even the presence of a significant amount of shot noise does not seriously disrupt the isoperimetric algorithm, but it significantly impacts the convergence of Ncuts to any solution.

B. Speed Comparison

In this section the four different images are segmented by using Spectral, Ncuts, Isoperimetric segmentation. Compare the these three algorithms and find out which algorithm will produced less time to segment the image. The Speed will be calculated for Spectral, Ncuts and Isoperimetric Segmentation from Table.1 in seconds.

From the Table.1 the flower image take 2.6877 seconds to produce segmentation result by Spectral segmentation, 5.4910 seconds to produce segmentation result by Normalised cut segmentation and 0.4931 seconds to produce segmentation result by Isoperimetric Segmentation. Similarly dog, mug, lenna images segmentation speeds are calculated. So the Isoperimetric segmentation has less time to segment the image compare to other two segmentation algorithms.

Table.1 Speed Comparison For Different Segmentation Algorithm

Image	Spectral Time (Seconds)	Ncuts Time (Seconds)	Isoperimetric Time (Seconds)
Flower	2.6877	5.4910	0.4931
Dog	17.6260	24.2674	3.4213
Mug	8.3818	8.6905	1.6977
Lenna	7.0222	3.4304	1.1101

V. CONCLUSION

In this paper image is segmented by three different segmentation algorithm. The original image segmented by Spectral segmentation, Normalised cut segmentation, Isoperimetric segmentation. Using these segmentation result the performance was analyzed with speed and stability. Consider the Flower image the speed was calculated. The segmentation time of Spectral segmentation is 2.6877 seconds, Ncuts Segmentation is 5.4910 seconds, and the Isoperimetric segmentation is 0.4931 seconds. The isoperimetric segmentation has very less time to segment the image compare to spectral and Ncuts segmentation. Adding three different Noises to Images the Isoperimetric Algorithm is more stable compare to Ncuts segmentation. So the isoperimetric segmentation algorithm is faster and more stable.

REFERENCE

1. Pedro F. Felzenszwalb and Daniel P. Huttenlocher. "Efficient Graph-Based Image Segmentation" International Journal of Computer Vision, Volume 59: 167–181, Number 2, September 2004.
2. Sudeep Sarkar and Padmanabhan Soundararajan, "Supervised learning of large perceptual organization: Graph spectral partitioning and learning automata",. IEEE Trans. on Pat. Anal. and Mach. Int., vol. 22, no. 5, pp. 504.525, May 2000.
3. L. Grady and E. L. Schwartz, "Isoperimetric partitioning: A new algorithm for graph partitioning," SIAM Journal on Scientific Computing, vol. 27, no. 6, pp. 1844–1866, June 2006.
4. Alex Pothen, Horst Simon, and Kang-Pu Liou, "Partitioning sparse matrices with eigenvectors of graphs",. SIAM Journal of Matrix Analysis Applications, vol. 11, no. 3, pp. 430.452, 1990.
5. Charles J. Alpert and Andrew B. Kahng, .Recent directions in net list partitioning: A survey,. Integration: The VLSI Journal, vol. 19,pp. 1.81, 1995.
6. Y.C. Wei and C.K. Cheng, "Ratio cut partitioning for hierarchical designs,. IEEE Trans. on CAD, 1991.
7. C. Hagen and A. Kahng, "New spectral methods for ratio cut partitioning and clustering" IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 11, no. 9, pp. 1074.1085, 1992.
8. Mei Yeen Choong, Wei Leong Khong, et al. "Graph-based Image Segmentation using K-Means Clustering and Normalised Cuts" Fourth International Conference on Computational Intelligencce computer networks, 2012.
9. Shi, J. and Malik, J "Normalized cuts and image segmentation" In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 731–737 1997.
10. Weiss,Y. "Segmentation using eigenvectors: A unifying view" In Proceedings of the International Conference on Computer Vision, pp. 975–982, 1999.
11. Ratan, A.L., Maron, O., Grimson, W.E.L., and Lozano-Perez, T "A framework for learning query concepts in image classification" In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 423–431, 1999.
12. Urquhart, R "Graph theoretical clustering based on limited neighborhood sets" In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition pp. 173–187, 1982.
13. Zahn, C.T "Graph-theoretic methods for detecting and describing gestalt clusters" IEEE Transactions on Computing, Vol. 20:68–86, 1971.
14. J. Ning,L. Zhang, D. Zhang and C. Wu "Interactive Image Segmentation by Maximal Similarity based Region Merging" In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 43, pp. 445-456, Feb, 2010.
15. Tranos Zuva, Oludayo O. Olugbara, et.al., "Image Segmentation, Available Techniques, Developments and Open Issues" Canadian Journal on Image Processing and Computer Vision Vol. 2, No. 3, March 2011
16. Ming Zhang, Reda Alhajj, "Improving the Graph-Based Image Segmentation Method" Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'O6), 2006, IEEE.
17. L. Grady and E. L. Schwartz, "Isoperimetric graph partitioning for image segmentation," IEEE Trans. on Pat. Anal. and Mach. Int., vol. 28, no. 3, pp. 469–475,March 2006.
18. Yoram Gdalyahu, Daphne Weinshall, and Michael Werman, .Self-organization in vision: Stochastic clustering for image segmentation, perceptual grouping and image database organization,. IEEE Pattern Anal. and Mach. Int., vol. 23, no. 10, pp. 1053.1074, October 2001, 288.
19. Bojan Mohar, "Isoperimetric inequalities, growth and the spectrum of graphs",. Linear Algebra and its Applications, vol. 103, pp.119.131, 1988.
20. Fan R. K. Chung, "Spectral Graph Theory", Number 92 in Regional conference series in mathematics. American Mathematical Society, Providence, R.I., 1997.
21. Jozef Dodziuk, "Difference equations, isoperimetric inequality and the transience of certain random walks",. Transactions of the American Mathematical Society, vol. 284, pp. 787.794, 1984.
22. Jozef Dodziuk and W. S Kendall, "Combinatorial Laplacians and the isoperimetric inequality",. in From local times to global geometry, control and physics, K. D. Ellworthy, Ed., vol. 150 of Pitman Research Notes in Mathematics Series, pp. 68.74. Longman Scientific and Technical, 1986.
23. Russell Merris, "Laplacian matrices of graphs: A survey", Linear Algebra and its Applications, vol. 197,198, pp. 143.176, 1994.
24. George Arfken and Hans-Jurgen Weber, Eds., "Mathematical Methods for Physicists", Academic Press, 3rd edition, 1985.
25. Norman Biggs, "Algebraic Graph Theory", Number 67 in Cambridge Tracts in Mathematics. Cambridge University Press, 1974.
26. David et al Martin, "a database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics".
27. Jefferey Cheeger, .A lower bound for the smallest eigenvalue of the Laplacian,.in Problems in Analysis, R.C. Gunning, Ed., pp. 195.199. Princeton University Press, Princeton, NJ, 1970.
28. Gene Golub and Charles Van Loan, Matrix Computations, The Johns Hopkins University Press, 3rd edition, 1996.
29. Bruce Hendrickson and Robert Leland, .The Chaco user's guide,. Tech. Rep. SAND95-2344, Sandia National Laboratory, Albuquerque, NM, July 1995.

AUTHORS PROFILE

Nalina.P received her B.E degree in Electronics and communication Engineering from Einstein College of Engineering, Tirunelveli, India in 2011.Currently she is pursing M.E degree Applied Electronics in Einstein College of Engineering, Tirunelveli, India.

Muthukannan.K received B.E degree in Electronics and Communication Engineering from Anna University, Chennai, India in 2005,M.E degree in Computer Science and Engineering from Manonmaniam Sundaranar University, Tirunelveli, India in 2010 and pursuing Ph.D in Anna University Chennai..Currently; he is the Assistant professor in department of Electronics and Communication Engineering, Einstein College of Engineering, Tirunelveli, India. He has published five papers in National Conferences and three papers in International Conferences and his paper is online access in IEEE explore and IJCA. He has more than six years teaching experience and one year Industrial experience. His research interests include Image Processing and Pattern Recognition.