Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 3, pp. 314-324

Image Segmentation using Euler Graphs

T.N. Janakiraman, P.V.S.S.R. Chandra Mouli

T.N. Janakiraman
Department of Mathematics
National Institute of Technology, Trichy, India.
E-mail: janaki@nitt.edu

P.V.S.S.R. Chandra Mouli
School of Computing Science and Engineering
V.I.T. University, Vellore, India.
E-mail: mouli.chand@gmail.com

Abstract: This paper presents a new algorithm for image segmentation problem using the concepts of Euler graphs in graph theory. By treating image as an undirected weighted non-planar finite graph (G), image segmentation is handled as graph partitioning problem. The proposed method locates region boundaries or clusters and runs in polynomial time. Subjective comparison and objective evaluation shows the efficacy of the proposed approach in different image domains.
Keywords: Image Segmentation, Graph theory, Euler Graphs, Cycles.

1 Introduction

Image segmentation can be treated as a graph partitioning problem which is solved by making use of cuts in a weighted graph based on certain criterion. The proposed method deals the image segmentation problem in a diverse manner. An excellent review for image segmentation is available in [8], [9], [15]. Earlier approaches to image segmentation are categorized into three groups: (1) Cluster the low level feature, such as histogram thresholding by [14], k–means / k–centroid by [12], [27] and mixture of Gaussians (MoG) by [2], (2) Edge linking such as dynamic programming by [26], relaxation approach by [16] and saliency network by [25] and (3) Region operations, such as region splitting and merging by [24], [23], region growing methods by [3] and by [13], by [17], by [11] and region competition by [20]. Applications of segmentation are abundant. It is heavily used in medical imaging. For example, segmentation of internal brain nuclei in MRI images as discussed in [38]. This work is aimed to bring robust image segmentation using graph theoretic concepts like Euler graphs and cycles. The proposed method finds the cycles of a given graph so that the image regions are formed by connecting all relevant pixels together. The relevancy of pixels is determined based on two parameters namely, edge weight similarity and node label similarity, which are described in the subsequent sections. The algorithm may end up at a particular stage when there is no possibility of refinement due to constraints imposed on cycle formation. Such paths are tried for further refinement. If the refinement is not possible then those paths are treated as open paths and may be treated as cuts. All the procedures of the proposed method run in polynomial time. The rest of the paper is organized as follows. In Section 2, a brief review on graph based segmentation is discussed. The basic definitions related to Euler graphs and some of its properties are presented in Section 3. In Section 4, the proposed algorithm and the experimental results are presented. Section 5 concludes the work.

2 Graph Approaches

Recently graph based image segmentation has attracted growing interest. Graph Theory and its concepts has been dominating in image processing research. The concepts of graph theory like maximum flow, maximum clique, shortest path, minimum spanning tree etc have been used for image processing problems. [21] discussed the various types of graph algorithms in computer vision. A special issue on graph based image processing is published in [32]. Early graph-based methods include by [4], [18] and more recent formulations in terms of graph cuts by [22], [31] and spectral methods by [30]. The notion of a connectivity graph was introduced by [19] to allow for image processing on a foveal sensor. This notion is introduced specifically to model the sampling of the macaque retina in [5]. The work of Zahn (1971) presents a segmentation method based on the minimum spanning tree (MST) of the graph. The segmentation criterion in Zahn's method is to break MST edges with large weights. The algorithm proposed by Urquhart (1982) normalizes the weight of an edge using the smallest weight incident on the vertices touching that edge. Work by Wu and Leahy (1993) introduced such a cut criterion, but it was biased toward finding small components. This bias was addressed with the normalized cut criterion developed by Shi and Malik (2000), which takes into account self-similarity of regions. These cut-based approaches to segmentation capture non-local properties of the image, in contrast with the early graph-based methods. Weiss (1999) has shown how the eigenvector-based approximations developed by Shi and Malik relate to more standard spectral partitioning methods on graphs. However, all such methods are too slow for many practical applications. An alternative to the graph cut approach is to look for cycles in a graph embedded in the image plane. [10] described the quality of each cycle is normalized in a way that is closely related to the normalized cuts approach. [7] described an efficient graph-based segmentation in which they defined a predicate for measuring the evidence for a boundary between two regions. Using that predicate, an algorithm is developed which makes greed decisions to produce segmentations that satisfy global properties. The literature in the most recent times reveal many improvements over these existing methods but for comparison and evaluation, the methods by Shi and Malik, Pedro F. Felzenzwalb etc are treated as benchmark works. A method to build a hierarchy of partitions of an image is introduced by [29] in which they build a hierarchy of partitions of an image by comparing in a pairwise manner the difference along the boundary of two components relative to the differences of components' internal differences. They stated the drawback of this method as the maximum and minimum criterion introduced are very sensitive to noise, although in practice it has a small impact. A MST pyramid based segmentation is carried out by [28] using dual graph contraction. For evaluating the segmentation results of the proposed methods with other existing methods, Precision, Recall and F-measure have been implemented since Berkeley Images [34] for segmentation have been evaluated using these three measures. The methods considered for comparison are [35], [36] and [37].

3 Background

Leonhard Euler discussed [6] graphs for the first time while solving the famous Seven Bridges of Königsberg problem. The following are some of the terms and their definitions used in this work. These definitions are taken as they are defined in [1].

3.1 Basic Definitions

Let G(V,E) be the given graph with V and E representing the vertex set and edge set respectively.

Definition 1. A trail that traverses every edge of G is called an Euler trail. It is named as Euler trail because Euler was the first to investigate the existence of such trails in graphs.

Definition 2. An Euler tour is a tour which covers all the edges of G.

Definition 3. A graph is an Euler graph or Eulerian if it contains an Euler tour.

Euler proved the following theorem and corollary through which a graph has Euler tour can be determined. The following characterizations are taken as they are defined and proved in Bondy and Murty (1982).

Theorem 4. *A non-empty connected graph is Eulerian if and only if it has no vertices of odd degree.*

Corollary: A connected graph has an Euler trail if and only if it has at most two vertices of odd degree.

3.2 Extraction / Development of Euler graphs from non-Euler graphs

If a graph does not have an Euler circuit, it still might be interested in knowing how it could be traveled with as few retraced edges as possible (starting and ending at the same vertex). Eulerian can be obtained in two ways. (i) By adding one spurious multiple edge which joins two adjacent odd degree vertices and (ii) By deleting the edges joining two adjacent odd degree vertices.

4 Proposed Method

The Euler graph and its properties are exposed in this work for solving image segmentation problem. The basic idea is that Euler graph is decomposed into edge disjoint cycles. The steps of the proposed method are given below:

```
Step-1: Representation of image as a grid graph
Step-2: Conversion of grid graph into Eulerian
Step-3: Segmentation Procedure
Step-4: Refinement of segments
```

These stages are discussed in detail in the following sub-sections.

4.1 Representation of Image as a Grid Graph

The image to be segmented is represented as a graph $G(V,E)$. To do so, each pixel is treated as a vertex of the graph. Edges are defined based on 8–connectivity of the neighborhood vertices. An edge $(v_i, v_j) \in E$ corresponds to a pair of neighboring vertices. The graph G, thus obtained is an undirected weighted non-planar graph. Clearly, an image of size $N \times N$ contains N^2 vertices, $(N-1)N$ vertical edges, $N(N-1)$ horizontal edges and $2(N-1)^2$ diagonal edges. Thus, in total there are $(N-1)N + N(N-1) + 2(N-1)^2 = 4N^2 - 6N + 2$ edges. Let $M = 4N^2 - 6N + 2$. The graph thus formed is visualized as a grid and hence called as grid graph. A sample grid graph of size 8×8 is shown in Figure 1. The weights are assigned to the edges by using the absolute intensity difference between the adjacent pixels.

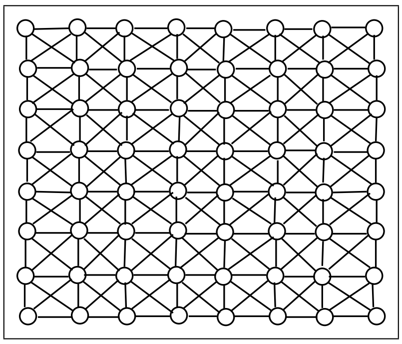

Figure 1: Grid graph of an image

4.2 Conversion of Grid Graph into Eulerian

The grid graph thus obtained is a connected non-Eulerian because some of the vertices have odd degree. The procedure for the conversion to Eulerian guarantees the formation of cycles covering all edges since all the vertices are of even degree. Border vertices are the vertices on the first row, last row, first column and last column. For this reason, the grid graph can be converted to Eulerian so that all vertices have even degree. This can be achieved in two ways. In the first case i.e., by adding one extra multiple edge for each of disjoint pair of adjacent odd degree vertices. The same weight is allocated to both duplicated and original edge to avoid ambiguity. The process is repeated until no such pair exists. In Figure 2, (a) and (d) show two grid graphs of size $4 \times 4$and 5×4 respectively. Figure 2(b) and (c) represent the two possible Euler graphs of (a), (e) and (f) represent the Euler graphs of (d).

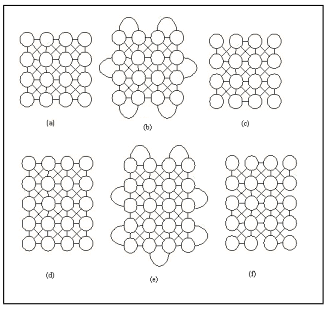

Figure 2: Grid graph and its corresponding Euler graphs

In the second case, instead of adding duplicate edges to the pair of adjacent vertices of odd degree, alternate edges are removed at the boundary to maintain even degree. It is found that there is no loss of information from images by removing such edges because all the edges removed are due to border vertices. In practice, there is not much information available at the border vertices and experimentally it is found that there is no variation in the segments formed in either way.

4.3 Segmentation Procedure

Once the given image is represented as Eulerian, the segmentation procedure is carried out over the Eulerian. The algorithms for image segmentation and segments_formed are given below.

```
1. Color all the edges as white.
2. Call segments_formed procedure.
3. Call regions_refinement procedure.
```

```
1. Select arbitrarily a white colored edge.
2. The selected edge is included in the
         temporary_growing_vector if it satisfies the threshold.
3. If the temporary_growing_vector forms a cycle then the
         closed path is stored in cycles_formed vector.
3a. The cycle formed is treated as a region. In the region
         formed, the edges in the closed path represent the
         boundary edges of the region. The edges present inside
         the region are internal edges of that region. The
         corresponding vertices are called boundary vertices
         and internal vertices respectively.
3b. The boundary edges of the region are colored black and
         internal edges are colored gray.
```

```
4. If the temporary_growing_vector has no cycle then choose
         the next minimum weighted white colored adjacent edge
         satisfying the threshold and goto step-2.
4a. If there is no edge available or which satisfies the
         threshold then backtrack its parent and search for
         next white colored adjacent edge satisfying the
         threshold.
4b. If no parent exits then the temporary_growing_vector is
         stored in open_paths vector. Color all the edges in
         open_paths as black.
4c. else
4d. remove the last included edge in the temporary_growing_vector.
4e. choose the next white colored adjacent edge and goto
      step-2.
5. If all the vertices are not covered or all edges are not
         colored gray or black then goto step-1.
6. If all the vertices are covered in the region either as
         boundary or as internal vertices then it induces the
         initial segmentation for the given threshold.
7. If all the edges are colored either gray or black
         representing the internal or boundary edges then the
         segmentation is subjected for refinement.
```

The algorithm uses a color structure which labels the edges as given below:

- Initially all edges are in WHITE color
- A visited edges is in GRAY color
- An edge in BLACK color indicates that it is a part of the boundary of a region.

The BLACK colored edges are marked permanently so that they are not considered for refinement. Only WHITE and GRAY colored edges are subjected for refinement. The criteria that is imposed on every edge to form a segment is defined in Equation (1). The Equation (1) refers to the difference of the maximum and minimum vertex labels in the cycle formed. In this case, it is used as the difference of the maximum and minimum vertex labels in the temporary_growing_vector.

$$T = \frac{maxv - minv}{2} \tag{1}$$

The algorithm starts by randomly choosing a white colored edge. At the first execution, the edge chosen is included directly in the temporary_growing_vector. Since a cycle cannot be formed with one edge, line 4 is executed where the algorithm tries to choose a white colored edge adjacent to the previously chosen edge. The edge is selected based on the threshold criteria. If no minimum weighted, white colored adjacent edge is available then, the algorithm backtracks to its parent and searches for another minimum weighted white colored adjacent edge. If it finds, then the last included edge in the temporary_growing_vector is removed since the algorithm could not traverse from that edge and adds the newly selected edge into temporary_growing_vector.

Line 3 of the algorithm checks for any cycle in temporary_growing_vector. To check this, BFS algorithm is used. Each cycle is treated as one region. If cycle is formed, then the closed path is stored in cycles_formed vector. The edges of the closed path are colored black indicating that they are boundary vertices. These edges are not chosen for forming any other cycles. The edges present inside the region are colored gray. These edges may be used for forming cycles once the white colored edges are exhausted. This will help in avoiding self overlapping region formation that means that the traversal starts from an internal edge and traverses to outside the region is termed as self overlapping. Self overlapping is avoided at the initial stage in order to get maximum number of non–overlapping regions but it is carried out in region refinement stage, if necessary.

During the execution of the algorithm, if it chooses a white colored edge outside any region and on its traversal, overlaps the existing region, then it is allowed because the internal edges of one region act like boundary edges of another region.

Another possibility during the traversal is that the temporary_growing_vector may not grow further because no further edge satisfies the criteria at any level (neither at the current edge nor at any of its parent edges), then the temporary_growing_vector stops traversing. By nature, Eulerian guarantees cycle formation but due to the threshold criteria, it may not form cycles all the cases. In such case, the temporary_growing_vector contains an open path and such paths are stored separately in open_paths vector.

In this way, the algorithm tries to traverse until it covers all vertices. This completes the first stage where, it induces an initial segmentation of image.

Refinement of Regions Formed

At this stage, all the edges are labeled to either gray or black. Refinement of black colored edges is not possible because they represent the boundaries of the regions already formed. The gray colored edges are subjected for refinement. The same procedure is used to form regions by choosing any randomly selected gray colored edge and for further traversals.

In this way, the algorithm tries to refine the segmentation for regions formation. Too much of refinement leads to over segmentation and no refinement leads to under segmentation. A moderate level of refinement is necessary. This is controlled by threshold selection.

5 Experimental Results

The proposed method is tested on standard Berkeley Image database. Two trivial synthetic images have been created and tested the algorithm on them. The results of the two synthetic images and the corresponding results are shown in Figure 3. The results presented in Figure 3 are the induced segmentations obtained before refinement process.

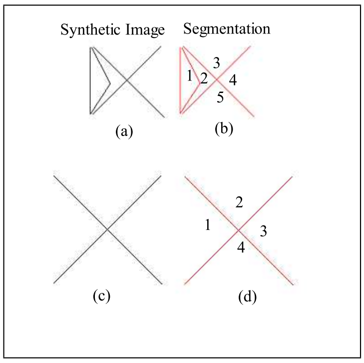

Figure 3: Segmentation results-I of synthetic images

In Figure 3, (a) and (c) are the two synthetic images created and the corresponding segmentations are shown in (b) and (d). These two synthetic images are created in such a way to study the behavior of the algorithm in open_paths case. As mentioned in the algorithm, the temporary_growing_vector stops traversing when there is no suitable edge satisfying the criteria. In such case the path is not closed and hence it is stored in open_paths vector. In Figure 3b, the segmentation result shows two open_paths (cross lines). The two different ends of the two open_paths are adjacent to one region formed. Thus, segmentation output gives a visualization that there are two closed regions labeled 1 and 2; and three open regions labeled 3, 4 and 5. In Figure 3d, the segmentation output shows two open_paths for which no end is adjacent to any other region. The four open regions formed by the two open_paths are labeled 1,2,3 and 4 in Figure 3d. After applying the refinement procedure, the segments obtained are shown

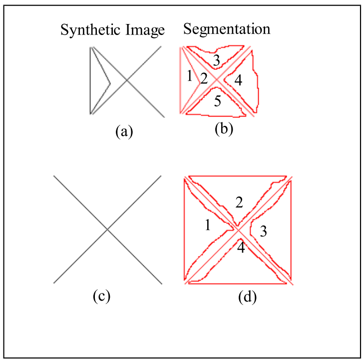

Figure 4: Segmentation results-II of synthetic images

in Figure 4. In Figure 4b, the segmentation output shows five regions labeled. Similarly in Figure 4d, there are 4 closed regions. The refinement process, in these cases, tried to get closed regions and in that process lead to over segmentation. This may be true in real images also. Hence, the refinement procedure is executed depending on the user's choice. The results of some real images taken from Berkeley Image database are shown in Figure 5. In Figure 5, the first and third columns represent the original image and second and fourth columns represent the segmentation result obtained.

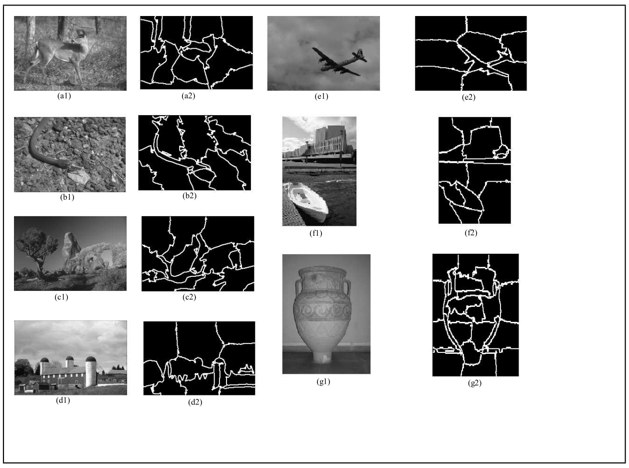

Figure 5: Segmentation results of some sample images in Berkeley Image Database

The proposed algorithm is executed on the 100 images in the database. The results of 50 images have been tabulated in Table 1. From the table, the following observations have been made. It is observed that those images having uniform background or average intensity range obtained best results; images having overlapping of objects or having complex structures, the statistical results are almost equivalent to the other existing methods chosen for comparison and for those images having high overlapping of objects or very dark images which cannot be visualized perfectly with the human eye, the proposed method could not segment the images and the statistical results revals that the F-measure for such images for the existing methods is better compared to the proposed method. The graphical representation of the results is shown in Figure 6 respectively.

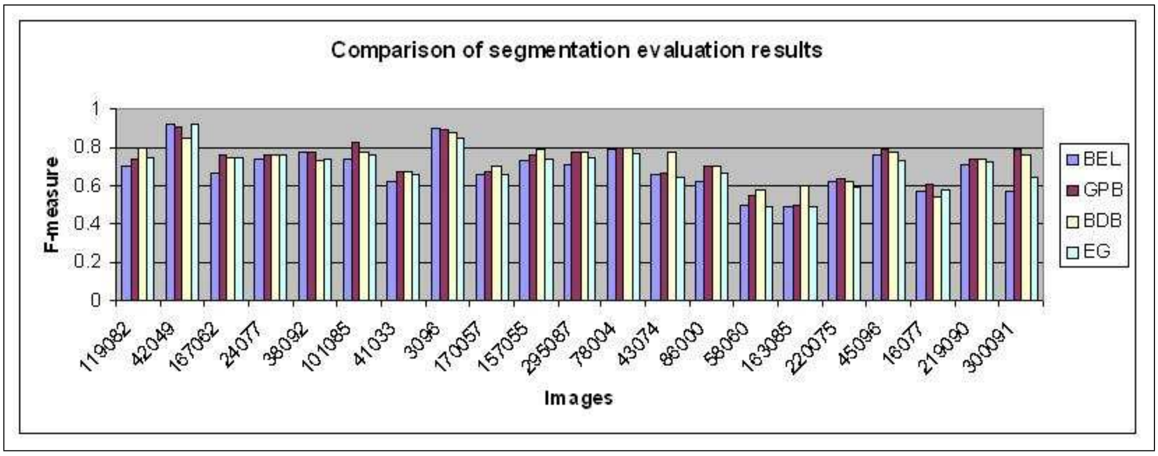

Figure 6: Comparison of segmentation evaluation results

Table 1: Comparison of segmentation evaluation results with other existing methods

Image Name	BEL	GPB	XREN	EG	Image Name	BEL	GPB	XREN	EG
119082	0.7	0.74	0.8	0.75	89072	0.68	0.71	0.71	0.69
42049	0.92	0.91	0.85	0.92	126007	0.72	0.78	0.76	0.75
167062	0.67	0.76	0.75	0.75	296007	0.66	0.69	0.69	0.65
24077	0.74	0.76	0.76	0.76	175032	0.49	0.62	0.63	0.6
38092	0.78	0.78	0.73	0.74	103070	0.68	0.68	0.62	0.65
101085	0.74	0.83	0.78	0.76	285079	0.71	0.72	0.71	0.69
41033	0.62	0.68	0.68	0.66	167083	0.61	0.75	0.75	0.7
291600	0.57	0.61	0.59	0.6	271035	0.73	0.73	0.71	0.71
130026	0.52	0.51	0.47	0.52	12084	0.48	0.52	0.49	0.5
241004	0.85	0.81	0.81	0.85	69015	0.79	0.82	0.79	0.75
147091	0.71	0.77	0.75	0.75	58060	0.5	0.55	0.58	0.49
189080	0.78	0.8	0.77	0.79	163085	0.49	0.5	0.6	0.49
14037	0.65	0.7	0.65	0.71	220075	0.62	0.64	0.62	0.59
62096	0.79	0.79	0.78	0.78	45096	0.76	0.79	0.78	0.73
227092	0.75	0.88	0.85	0.88	16077	0.57	0.61	0.54	0.58
253027	0.63	0.65	0.69	0.68	219090	0.71	0.74	0.74	0.72
229036	0.67	0.76	0.72	0.75	300091	0.57	0.79	0.76	0.65
3096	0.9	0.89	0.88	0.85	156065	0.66	0.67	0.64	0.63
170057	0.66	0.68	0.7	0.66	76053	0.61	0.61	0.62	0.59
157055	0.73	0.76	0.79	0.74	304034	0.47	0.49	0.47	0.41
295087	0.71	0.78	0.78	0.75	86016	0.39	0.52	0.42	0.48
78004	0.79	0.8	0.8	0.77	8023	0.41	0.5	0.42	0.4
43074	0.66	0.67	0.78	0.65	108082	0.43	0.46	0.47	0.43
86000	0.62	0.7	0.7	0.67	69040	0.5	0.55	0.57	0.52

6 Conclusion

In this paper, a novel algorithm for segmenting an image into different regions using Euler graphs has been proposed. The algorithm starts by randomly choosing an edge and tries to form closed regions. In cases, open paths are formed. The color look up table is used for edges to trace their transition. A white color indicates unvisited edge, a gray color indicates visited and may go for refinement and black color indicates visited and marked permanently for no refinement since it is already a part of a region boundary. The procedures discussed run in polynomial time. The MST and cycles method performs better compared to Euler Graph method in terms of precision, recall and F measures.

Bibliography

[1] J.A. Bondy and U.S.R. Murty. *Graph Theory with Applications, Fifth printing*. Elsevier Science Publishing Co., Inc., 52, Vanderbilt Avenue, New York, 1982.

[2] H.Greenspan C.Carson, S.Belongie and J.Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, 2002.

[3] C.R.Brice and C.Fennema. Scene analysis using regions. *Artificial Intelligence*, 1(3–4):205–226, 1970.

[4] C.T.Zahn. Graph theoretical methods for detecting and describing gestalt clusters. *IEEE Transactions on Computation*, 20:68–86, 1971.

[5] E.L.Schwartz. Spatial mapping in the primate sensory projection: Analytic structure and relevance to perception. *Biological Cybernetics*, 25(4):181–194, 1977.

[6] Euler. Solutio problematis ad geometriam situs pertinentis comment. *Academiae Sci. I. Petropolitanae*, 8:128–140, 1736.

[7] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[8] K.S. Fu and J.K. Mui. A survey of image segmentation. *Pattern Recognition*, 13:3–16, 1981.

[9] R.M. Haralick and L.G. Shapiro. Survey, image segmentation techniques. *Computer Vision, Graphics and Image Processing*, 29:100–132, 1985.

[10] I.Jermyn and H.Ishikawa. Globally optimal regions and boundaries as minimum ratio weight cycles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4):1075–1088, 2001.

[11] L.Vincent and P.Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, 1991.

[12] R.Velthuizen L.Hall D.Goldgof L.Clarke M.Clark and M.Silbiger. Mri segmentation using fuzzy clustering techniques. *IEEE Engineering in Medicine and Biology Magazine*, 13(5):730–742, 1994.

[13] O.Monga. An optimal region growing algorithm for image segmentation. *PRAI*, 1(4):351–375, 1987.

[14] N. Otsu. A threshold selection method from grey level histograms. *IEEE Transactions on System, Man and Cybernetics*, 9:62–66, 1979.

[15] N.R. Pal and S.K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26:1277–1294, 1993.

[16] P.Parent and S.W.Zucker. Trace inference, curvature consistency, and curve detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8):823–839, 1989.

[17] R.Adams and L.Bischof. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):641–647, 1994.

[18] R.Urquhart. Graph theoretical clustering based on limited neighborhood sets. *Pattern Recognition*, 15(3):173–187, 1982.

[19] P.W.Ong. R.Wallace and E.Schwartz. Space variant image processing. *International Journal of Computer Vision*, 13(1):71–90, 1994.

[20] S.C.Zhu and A.L.Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900, 1996.

[21] M.Pelillo S.Dickinson and Ramin Zabih. Introduction to the special section on graph algorithms in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1049–1052, 2001.

[22] J. Shi. and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[23] S.L.Horowitz and T.Pavlidis. A graph-theoretic approach to picture processing. *JACM*, 7(2):282–291, 1976.

[24] S.L.Horowitz and T.Pavlidis. Picture segmentation by a tree traversal algorithm. *JACM*, 23(2):368–388, 1976.

[25] S.Ullman and A.Shaashua. Structural saliency: The detection of globally salients structures using a locally connected network. Technical report, Cambridge, MA, USA, 1988.

[26] U.Montanari. On the optimal detection of curves in noisy pictures. *Communications. ACM*, 14(5):335–345, 1971.

[27] A.R. Weeks and G.E. Hague. Color segmentation in the hsi color space using the k-means algorithm. In *SPIE, p. 143-154, Nonlinear Image Processing VIII, Edward R. Dougherty; Jaakko T. Astola; Eds*, Volume 3026, pages 143–154, 1997.

[28] Y.Haxhimusa and W.Kropatsch. Hierarchy of partitions with dual graph contraction. *Lecture Notes in Computer Science*, 2781:338–345, 2003.

[29] Y.Haxhimusa and W.Kropatsch. Segmentation graph hierarchies. In *Proceedings of Structural, Syntactic, and Statistical Pattern Recognition*, Volume 3138, pages 343–351. LNCS, 2004.

[30] Y.Weiss. Segmentation using eigenvectors: A unifying view. In *Proceedings of the International Conference on Computer Vision*, Volume 2, pages 975–982, 1989.

[31] Z.Wu and R.Leahy. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1101–1113, 1993.

[32] Dickinson, S., Pelillo, M. and Zabih, R. Introduction to the special section on graph algorithms in computer vision *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1049–1052, 2001.

[33] D. Martin and C. Fowlkes and D. Tal and J. Malik A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics *Proc. 8th Int'l Conf. Computer Vision*, Volume 2, pages 416–423, 2001.

[34] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/

[35] Piotr Dollar, Zhuowen Tu, and Serge Belongie Supervised Learning of Edges and Object Boundaries *Proc. IEEE Computer Vision and Pattern Recognition, CVPR*, June, 2006.

[36] Michael Maire, Pablo Arbelaez, Charless Fowlkes and Jitendra Malik Using Contours to Detect and Localize Junctions in Natural Images *Proc. IEEE Computer Vision and Pattern Recognition, CVPR*, 2008.

[37] Xiaofeng Ren Multi-Scale Improves Boundary Detection in Natural Images *Proc. ECCV Conference*, 2008.

[38] Marius George Linguraru, Miguel *Á* ngel Gonz *á* lez Ballester, Nicholas Ayache Deformable Atlases for the Segmentation of Internal Brain Nuclei in Magnetic Resonance Imaging *International Journal of Computers, Communications & Control*, Volume 2, No. 1, pages 26–36, 2007.

T. N. Janakiraman (born on February 14, 1953) received his Ph.D. in graph theory and its applications from Madras University, India in 1991. He did his P.D.F. from the same university in graph theory and its applications in 1994. He is currently associated with Department of Mathematics, National Institute of Technology, Trichy, India. He has two sponsored research projects to his credit and published around 40 papers in refereed international journals. His main research interests are in graph theory and its applications in digital image processing, wireless ad hoc networks and digital video processing.

P.V.S.S.R. Chandra Mouli (born on May 21, 1976) received his Ph.D. from National Institue of Trichy in 2010. He is currently working as Associate Professor in School of Computing Science and Engineering, VIT University, Vellore. His research interests include Image Segmentation, Pattern Classification and Wireless Ad–hoc networks. He has published 10 refereed research papers in various international journals and conferences. He was co–investigator to research project sponsored by Defence Research Development Organization (DRDO), New Delhi and also worked as a research fellow in another research project sponsored by DRDO, New Delhi, India. He is a life member of ISTE, and also member of CSI. He is reviewer for many international journals.