Individual Tree Crown Segmentation in Aerial Forestry Images by Mean Shift Clustering and Graph-based Cluster Merging

YaKun Wang,† , *Young Sung Soh*,†, *and Howard Schultz*††
r
†Dept. of Information Engineering, MyongJi University , San38-2 Namdong, Cheoin-gu, Yongin, Kyunggido, 449-728, Korea
†† Dept. of Computer Science, Univ. of Massachusetts, 140 Governors Dr., Amherst, MA 01003, U.S.A.

Abstract
Individual tree crown segmentation is frequently required in forest inventory, biomass measurement, change detection, tree species recognition, etc. It is almost impossible to do manual segmentation of huge forest by human. In this paper, we present an automatic method for individual tree crown segmentation in aerial forestry images. We first extract treetops using the method in [1]. Next we apply mean shift clustering to group pixels into clusters having homogeneous properties. Then we build a cluster adjacency graph where clusters belonging to the same crown are merged. We tested our method on some forestry images and obtained good results.

Key words:
Mean Shift Clustering, Graph-based Merging, Tree Crown Segmentation.

1. Introduction

It is important to segment individual tree crowns to maintain forest inventory, measure biomass, detect changes, and identify tree species. Since it is almost impossible to do manual delineation of individual tree crowns in huge forest, automatic methods using various sensors were proposed. Some approaches[3-8] use only spectral data whereas some other approaches[1,9] adopt both spectral and range data. In this research, we assume that only spectral data are present. Many approaches were proposed that used only spectral data. They include valley following, template matching, contour finding in scale space, fuzzy thresholding, Brownian motion, and random walk. In chapter 2, we review some of these approaches and pay special attention to Brownian motion-based segmentation since it is proven to perform best among aforementioned approaches in [6]. Chapter 3 outlines the proposed method and chapter 4 illustrates experimental results. In chapter 5, we summarize and discuss our approach.

2 Related works

Erikson[6] gives a brief review on some of the works on tree crown segmentation. Gougeon[3] proposed the valley-following approach where he assumes there is a dark boundary between tree crowns. Thus this does not work well if multiple trees are in close touch and have no distinct dark boundaries. Lindeberg[4] built primal sketch for different scales and an accumulated primal sketch is also constructed in which each tree crown forms a local maximum. Then each tree crown is grown from each maximum within an accumulated primal sketch. The method may not keep the shape of tree crowns since original image was blurred by a series of Gaussian kernels to set up a scale space. Pollock[8] and Olofsson[7] presented a template matching method. They used 3-D parameterized ellipsoid to model tree crowns. They generated a series of synthetic templates with different illumination, viewing angle, and sizes. Local maxima in cross correlation gave the location of tree crowns. Since synthetic templates with limited variety are used, only a rough approximation of original tree shape is possible. Erikson[5] suggested a segmentation approach based on Brownian motion. He found seed points and constructed a new image named *NumPass* that records the number of times each pixel has been visited by a Brownian particle in motion. *NumPass* is thresholded to produce binary *NumPass* and constrained erosion is applied to each region with predefined range of sizes with a seed as a center. In this approach, the selection of parameters such as threshold used for *NumPass* and the range of region sizes is important to get accurate tree crowns. While this method seems to give the best results among the reviewed, it is very time consuming. Moreover this method assumes that there is a dark boundary among tree crowns. However, this assumption frequently does not hold in real situations.

Manuscript received November 5, 2006.
Manuscript revised November 25, 2006.

3. The proposed method

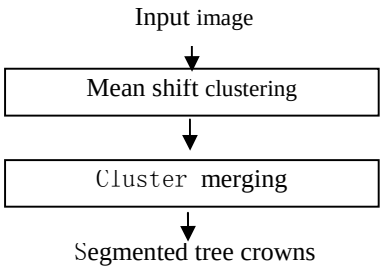


Fig. 3. 1 The block diagram of the proposed approach

Fig. 3.1 shows the block diagram of the proposed method. Given an input image, mean shift clustering is performed to produce clusters with homogeneous properties. Then clusters will be merged if they share an enough boundary and the colors are similar. Each group of clusters merged will be a final segment. Following sections explain in detail the steps in Fig. 3.1.

3.1 Mean Shift Clustering

Comaniciu et al.[2] proposed Mean Shift as a robust method towards feature space analysis. In the method, each feature is considered as coming from a probability distribution so that the whole feature space consists of many pdf(probability density function)s. They used KDE(Kernel density estimation) to estimate the pdfs and applied mean shift to find their modes. To find clusters, mean shift filtering is performed first with predefined kernel bandwidths for range(color) and spatial extent. An iterative procedure with guaranteed convergence is applied to estimate the filtered value of each pixel. After filtering is done, pixels residing in predefined range and spatial bandwidth are grouped to produce clusters.

3.2 Cluster merging

Given clusters generated by mean shift, we merge clusters that seem to stem from a single tree crown. The process is depicted in Fig. 3.2

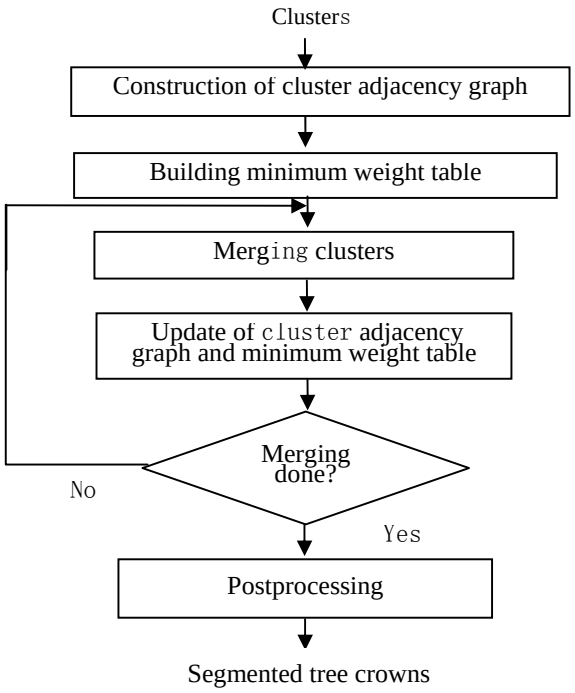
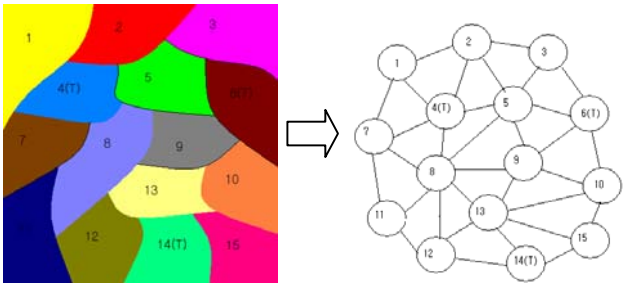


Fig. 3. 2 The block diagram of graph-based cluster merging

3.2.1 Construction of CAG(Cluster adjacency graph)

We first construct CAG using clusters. In CAG, a node corresponds to a cluster and an edge is drawn between two nodes if they are spatially connected. Each node has information such as its color value and the existence of treetop within itself. Treetops were extracted in advance and made available by the procedure described in [1]. Each edge has an associated weight and the method to compute weight will be described in 3.2.2. Fig. 3.3 shows an example of building CAG from cluster image. In Fig. 3.3(a), numbers represent cluster IDs. Each cluster is shown to have the same color. Mean shift filtering and clustering may result in different, but very close color values for pixels in the same cluster. We just average them to use as a representative color for the cluster. In Fig. 3.3(b) CAG is drawn. Nodes with (T) mean the nodes having treetops. The terms node and cluster will be used interchangeably in this paper since they represent the same thing.



(a) (b)

Fig. 3. 3 Example of building CAG(Cluster adjacency graph)

3.2.2 Building MWT(minimum weight table)

Next we build MWT. An example of building MWT from CAG is shown in Fig. 3.4. Each entry of the table corresponds to a node with treetop and has a link to the edge with minimum weight among edge(s) connected to that node. In this particular example, Node 4 has five edges and among them the edge connected to node 5 has minimum weight. We define *edge(i-j)* as an edge connecting node i and node j. Similarly we define W_{ij} as the weight of *edge(i-j).*

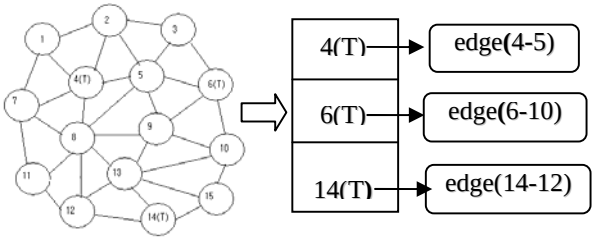


Fig. 3.1 Example of building MWT(Minimum weight table) from CAG

Weight W_{ij} is determined based on two entities : color difference between nodes i and j, and the number of boundary pixels shared by two nodes. Color difference between nodes i and j is computed using equations (1) and (2).

$$GammaMode_i = pow(\frac{R_i - R_{\min}}{R_{\max} - R_{\min}}, \gamma) * 255 (0 < \gamma << 1) \quad (1)$$

$$GammaDiff_{Ti} = abs(GammaMode_T - GammaMode_i) \quad (2)$$

, where R_T and R_i are color values of two adjacent clusters T and i, and T is the cluster with treetop. R_{min} and R_{max} are the minimum and maximum color values among all clusters. Here we do not use the original color value to compute the difference. Instead gamma function output of original value is used where gamma value is far less than 1. The reason why we apply gamma function is that it tends to reduce the difference in high value range and amplify it in low value range. In real situations, treetops usually have high value and the boundary between tree crowns has low value. So by reducing the difference in high range, clusters adjacent to treetop are more mergeable and by amplifying the difference in low value range, different tree crowns become less mergeable.

Next we compute how much two adjacent clusters share boundary pixels using equation (3).

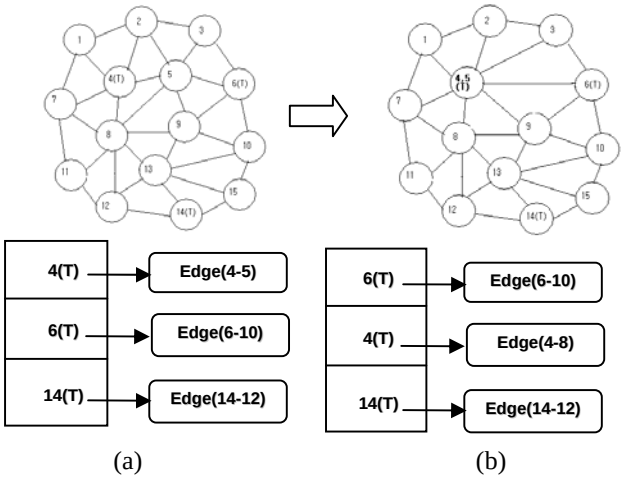
$$ShareRatio_{Ti} = ShareNum_{T\,i} / TotalBoundaryPixelNum_T \quad (3)$$

, where $ShareNum_{T\,i}$ is the number of boundary pixels shared by two adjacent clusters T and i, $TotalBoundaryPixelNum_T$ the number of boundary pixels in cluster T, and $ShareRatio_{Ti}$ the degree that clusters T and i share their boundaries. We combine two measures into a single figure of merit as shown in equation (4).

$$W_{Ti} = \alpha * GammaDiff_{Tj} / 255 + (1-\alpha) * (1 - ShareRatio) \quad (4)$$

3.2.3 Merging clusters and update of CAG and MWT

After building MWT, we sort the entries according to their minimum weights. The entry with the minimum weight among minimum weights of all the entries is chosen and two clusters connected to that edge are merged if the chosen minimum weight is less than a predefined threshold.. Then the CAG is updated. The example of this operation is depicted in Fig. 3.5. In Fig. 3.5(a), cluster 4 has a minimum weight edge among all clusters in the table and that edge is connected to cluster 5. Thus clusters 4 and 5 are merged, *edge(4-5)* is removed, edges connected to both clusters will now point to merged cluster, and MWT will be updated accordingly. (see Fig. 3.5(b)). This process is repeated until no entries have minimum weight less than a predefined threshold. Since initially all entries of MWT were clusters with treetops, merging starts only from treetops

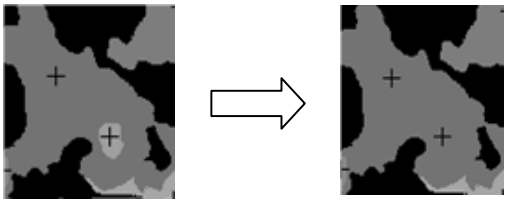


(a) (b)

Fig. 3.2 Example of cluster merging and update of CAG and MWT

3.2.4 Postprocess ing

After cluster merging is done, there may be cases where one (merged) cluster is completely enclosed by another merged cluster as in Fig. 3.6 (a). In this case, the enclosed cluster is absorbed by the enclosing one. (See Fig. 3.6 (b))



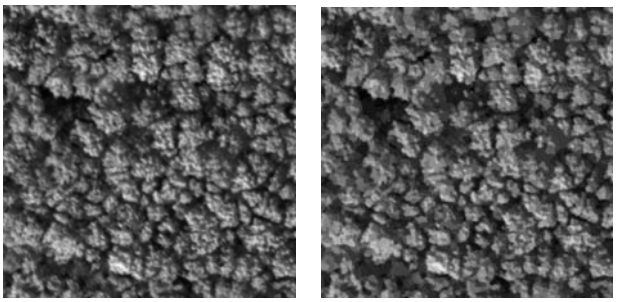
(a) (b)

Fig.3.6 Example of postprocessing

4 Experimental results

4.1 Mean shift clustering

Fig. 4.1 shows the result of mean shift clustering for test image 1. Fig. 4.1(a) is the input gray level image of size 1000 x 1000 with 256 gray levels and Fig. 4.1(b) the result after mean shift clustering where 11,673 clusters are present. There was a data reduction of almost hundred times even though it is not clearly visible due to too much a downscaling of images for viewing purposes.

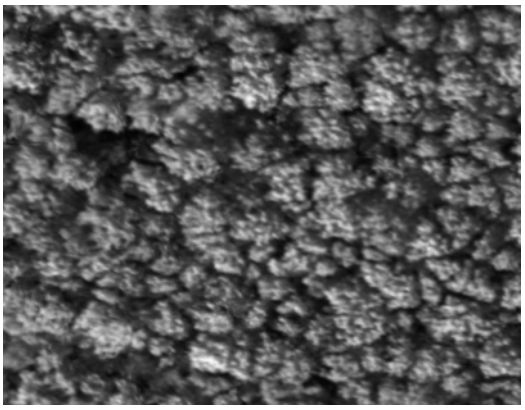


(a) (b)

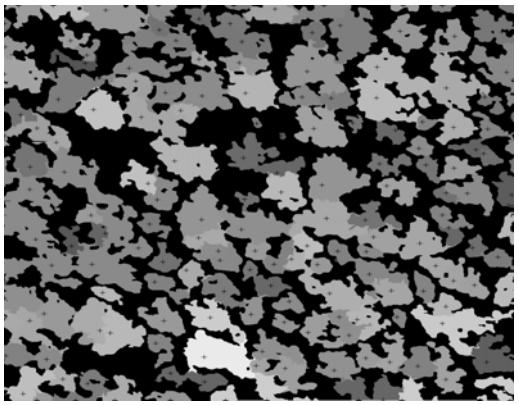
Fig. 4.1 Result of mean shift clustering for test image 1

4.2 Cluster merging

Fig. 4.2 shows the final tree crown segmentation result by cluster merging. Fig. 4.2(a) is the input and Fig. 4.2(b) the segmentation result where cross indicates the location of treetop. In Fig. 4.2(b), spatially disconnected regions and spatially connected regions with different gray levels represent different clusters



(a)



(b)

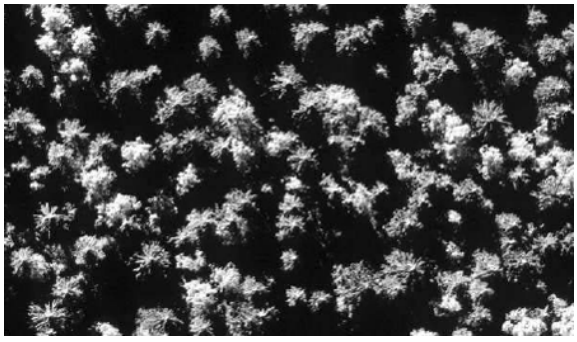
Fig. 4.2 Cluster merging result for test image 1

Since it is not easy to assess the performance visually, we put quantitative analysis result in Table 1. Here GT means Ground Truth and 1-to-1 the correct mapping where a single tree crown in real world is segmented as a single tree crown. Similarly 1-to-M means that a single tree crown in real world is segmented into many tree crowns, whereas M-to-1 signifies the inverse of 1-to-M. There were 120 tree crowns in real world. The proposed method found 102 tree crowns out of which 82 belong to 1-to-1 category. Thus the Segmentation Success Ratio is 68.3%. We have many M-to-1's since several nearby tree crowns are so close each other that they were segmented as a single tree crown. This may be overcome if we can exploit the a priori knowledge on tree species and their sizes.

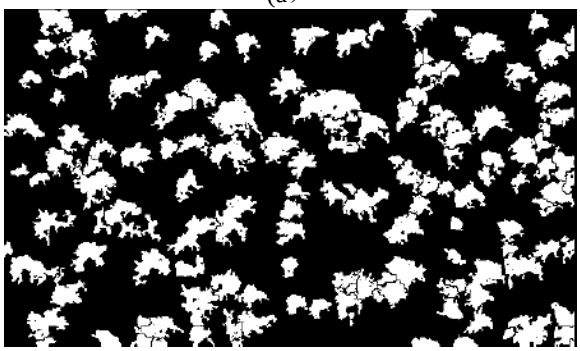
Table 1. Quantitative analysis of segmentation result for test image 1

GT	1-to-1	1-to-M	M-to-1	Segmentation Success Ratio
120	82	5	33	68.3%

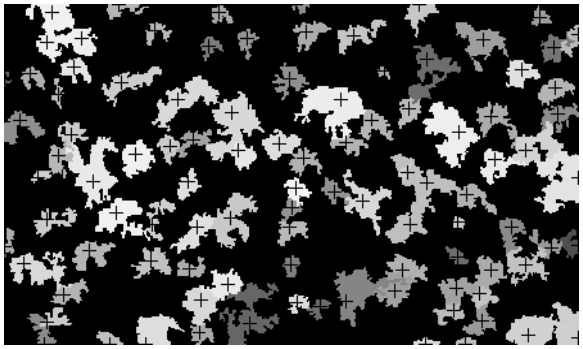
Fig. 4.3 depicts the comparison results between Erikson's Brownian motion method[6] and the proposed. Fig. 4.3(a) is the test image 2 from [6], Fig. 4.3(b) the Erikson's result, and Fig. 4.3 (c) our result. Again, crosses in Fig. 4.3(c) are treetops.



(a)



(b)



(c)

Fig. 4.3 Comparative cluster merging results for test image 2

Though it is not easy to do a qualitative comparison, some of the problems associated with Erikson's are clearly seen especially in mid bottom part of the image. Fig. 4.4 shows the expanded view of that part. Fig. 4.4(a) is the original, Fig. 4.4(b) Erikson's, and Fig. 4.4(c) our result. There are 3 tree crowns in Fig. 4.4(a). Erikson's method found 8 tree crowns whereas ours found 3. Erikson's method exposes severe oversegmentation problem. Moreover, our method seems to produce more accurate boundaries.

(a) (b) (c)

Fig.4. 4 Example of oversegmentation in Erikson's method

Table 2. Comparative quantitative analysis of segmentation results for test image 2

Method	GT	1-to-1	1-to-M	M-to-1	Segmentation Success Ratio
Erikson's	101	78	14	9	77.2%
The proposed		93	2	6	92.9%

Table 2 illustrates the comparison result quantitatively. The conventions used in Table 2. is the same as those in Table 1. There were 101 tree crowns in real world. Our method found 93 correctly whereas Erikson's has 78 correct matches. In contrary to test image 1, we obtain much better results because tree crowns are relatively well separated in test image 2. Our method outperforms Erikson's in Segmentation Success Ratio.

5 Conclusions and discussions

In this paper, we proposed a new tree crown segmentation method based on mean shift clustering and graph-based cluster merging. Mean shift clustering is applied to get

initial clusters having homogeneous properties. Then CAG(cluster adjacency graph) is constructed and MWT(minimum weight table) is built. Iterative process is performed on two data structures until convergence to do cluster merging. Lastly postprocessing is done to take care of complete inclusion cases among merged clusters. We produced one noncomparative and one comparative result. Both results show that our method is promising. As was noted in chapter 4, we have more M-to-1's than 1-to-M's. This seemed to happen in part due to the selection of parameters in cluster merging and in part due to the nature of the images used. This is intended for future research.

References

[1] B. Rawert, Automatic Tree-Crown Segmentation Using LM Filters and Balloons in 2004

[2] D. Comaniciu and P. Meer, Mean Shift: A Robust Approach Toward Feature Space Analysis, IEEE Trans. On PAMI, vol.24. No. 5. May 2002, pp.603-619

[3] F.A. Gougeon, A crown-following Approach to the Automatic Delineation of Individual Tree Crowns in High Spatial Resolution Aerial Images, Canadian Journal of Remote Sensing 21, No.3, 1995, pp.274-284.

[4] T. Lindeberg, Feature Detection with Automatic Scale Selection, International Journal of Computer Vision 30(2), 1998. pp. 79-116.

[5] M. Erikson, Structure-Perserving Segmentation of Individual Tree Crowns By Brownian Motion. 13th Scandinavian Conference, SCIA 2003 Halmstad, Sweden, June 29 – July 2, 2003 Proceedings, LNCS vol.2749, 2003, pp.283-289

[6] M. Erikson, Segmentation and Classification of Individual Tree Crowns in High Spatial Resolution Aerial Images, Printed in Sweden by Universitetstryckeriet, Uppsala University, Uppsala, 2004

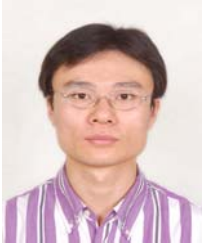
[7] K. Olofsson, Detection of Single Trees in Aerial Images using Template Matching. In ForestSat 2002, Operational tools in forestry using remote sensing techniques. Proceedings CD-ROM, talk FI6.3, session Forestry Inventory 6, Monitoring Forest Establishment and Development, Forest Research, Forest Commission, Heriot Watt University, Edinburgh, Scotland, August 5-9, 2002

[8] R. J. Pollock, The automatic Recognition of Individual trees in Aerial Images of Forests Based on a Synthetic Tree Crown Image Model, Doctoral Thesis, University of British Colombia, Vancouver, Canada. 1996, 172 p.

[9] S. C. Popescu, R. H. Wynne, and R. F. Nelson, Measuring Individual Tree Crown Diameter with Lidar and Assessing its Influence on estimating Forest Volume and Biomass, Canadian J. Remote Sensing, Vol. 29, No. 5, 2003, pp. 564-577

Acknowledgments

This work is supported in part by a grant (IIS-0430742) from the United States National Science Foundation.



YaKun Wang received B.S. in Computer Science from Henan University of Finance and Economics in China in 2001 and M.S. in Information Engineering from MyongJi University in Korea in 2003. He is now pursuing Ph.D in MyongJi University. His research interests include image processing and computer vision.

Young Sung Soh received B.S. in Electrical Engineering from Seoul National University in 1978. He received M.S. and Ph.D in Computer Science from Univ. of South Carolina in 1986 and 1989, respectively. His main research focus is on computer vision and pattern recognition..

Howard Schultz received B.S. and M.S. in Physics from UCLA in 1972 and 1974, and Ph.D in Physical Oceanography from Univ. of Michigan in 1982. His research interests include quantitative methods for image understanding and remote sensing.