

INCORPORATING EDGE INFORMATION INTO BEST MERGE REGION-GROWING SEGMENTATION

James C. Tilton¹ and Edoardo Pasolli²

¹Goddard Space Flight Center, Greenbelt, MD, U. S. A.,
²Purdue University, West Lafayette, IN, U. S. A.

ABSTRACT

We have previously developed a best merge region-growing approach that integrates nonadjacent region object aggregation with the neighboring region merge process usually employed in region growing segmentation approaches. This approach has been named HSeg, because it provides a hierarchical set of image segmentation results. Up to this point, HSeg considered only global region feature information in the region growing decision process. We present here three new versions of HSeg that include local edge information into the region growing decision process at different levels of rigor. We then compare the effectiveness and processing times of these new versions HSeg with each other and with the original version of HSeg.

Index Terms— Image processing, image analysis, image segmentation, image edge detection

1. INTRODUCTION

Described and discussed in Tilton, *et al.* [1] is a best merge region-growing segmentation approach that integrates nonadjacent region object aggregation with the neighboring region merging process usually employed in region growing segmentation approaches. This approach has been named HSeg, because it provides a hierarchical set of image segmentation results. We have noted that in some of the more detailed levels of the HSeg segmentation hierarchy, large and apparently homogeneous areas are sometimes separated into more than one region with region boundaries that do not correspond to any apparent object boundary. It is apparent that HSeg is responding to gradual changes in region features that are not important for most image analysis applications. Realizing that previous versions of HSeg consider only global region feature information in the region growing decision process, we have devised augmentations of HSeg that incorporate local edge information into the region growing process. We expect that these augmentations should make HSeg less sensitive to gradual changes in regions features, and generally improve the performance of HSeg. While any edge operator could be utilized, we have chosen to use edge information generated

by the Frei-Chen edge operator [2] in our augmentations of HSeg. The Frei-Chen edge operator applies a combination of nine convolution masks to generate a normalized edge image that is sensitive to lines and/or edges in the horizontal, vertical and diagonal directions.

The next two sections of this paper summarize the previous HSeg image segmentation approach (version 1.59, which similar to the version described in [1]) and the Frei-Chen edge operator. We then describe three alternate approaches (versions 1.61, 1.71, and 1.81) for incorporating edge information into HSeg. We follow this with an evaluation of the effectiveness of these new versions of HSeg as compared to version 1.59. We conclude with a discussion of the results and considerations for future work.

2. THE HSEG SEGMENTATION APPROACH

The HSeg image segmentation approach is based on hierarchical step-wise optimization (HSWO) as described in [3]. HSWO is an iterative form of region growing, in which the iterations consist of finding the most optimal or best segmentation with one less region than the current segmentation. HSWO is performed by finding a threshold value, T_{merge} , equal to the value of a dissimilarity criterion of the most similar pair of spatially adjacent regions, and then merging all pairs of regions that have dissimilarity equal to T_{merge} . HSeg adds to HSWO a step following each step of adjacent region merges in which all pairs of spatially non-adjacent regions are merged that have dissimilarity $\leq S_w T_{merge}$, where $0.0 \leq S_w \leq 1.0$ is a factor that sets the priority between spatially adjacent and non-adjacent region merges. Note that when $S_w = 0.0$, HSeg reduces to HSWO. HSeg provides choices among several dissimilarity criteria, including one based on minimizing the increase of mean squared error between the region mean image and the original image data (BSMSE), and one based on the Spectral Angle Mapper (SAM) criterion [4]. All of these dissimilarity criterion depend on global region features such as region size (number of pixels), sum of pixel values in each spectral band and sum of the square pixel values in each spectral band. See [1] for a more complete description of HSeg and the dissimilarity criterion utilized by HSeg.

3. THE FREI-CHEN EDGE OPERATOR

Any edge operator can be used to generate edge information for input to the new versions of HSeg, including the well-known Sobel and Prewitt operators. We choose to use the Frei-Chen edge operator because it is sensitive to diagonal edges in addition to the vertical and horizontal edges that the Sobel and Prewitt operators are designed to detect.

The Frei-Chen edge operator was first described in [2]. More recent updates of the Frei-Chen edge operator have resulted in a revision of the weighting factors for the Frei-Chen masks (see [5]). Following [5], the Frei-Chen edge operator consists of the nine following unique 3x3 convolution masks:

$$\begin{aligned}
 G_1 &= \frac{1}{2\sqrt{2}} \begin{bmatrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ -1 & -\sqrt{2} & -1 \end{bmatrix} & G_2 &= \frac{1}{2\sqrt{2}} \begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix} \\
 G_3 &= \frac{1}{2\sqrt{2}} \begin{bmatrix} 0 & -1 & \sqrt{2} \\ 1 & 0 & -1 \\ -\sqrt{2} & 1 & 0 \end{bmatrix} & G_4 &= \frac{1}{2\sqrt{2}} \begin{bmatrix} \sqrt{2} & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & -\sqrt{2} \end{bmatrix} \\
 G_5 &= \frac{1}{2} \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} & G_6 &= \frac{1}{2} \begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{bmatrix} \\
 G_7 &= \frac{1}{6} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix} & G_8 &= \frac{1}{6} \begin{bmatrix} -2 & 1 & -2 \\ 1 & 4 & 1 \\ -2 & 1 & -2 \end{bmatrix} \\
 G_9 &= \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}
 \end{aligned}$$

The first four masks are used for edges, the next four for lines, and the last mask is to provide a mask average for normalization. Since we are interested in image edges, we compute our edge value, E_{value} , as follows:

$$E_{value} = \sqrt{\frac{M}{S}}$$

where $M = \sum_{k=1}^4 (G_k * I)^2$ and $S = \sum_{k=1}^9 (G_k * I)^2$

The value of E_{value} will always be in the range of 0.0 through 1.0, with a value of 1.0 representing the most abrupt possible edge (a ‘‘delta’’ edge). For most remotely sensed images, $\max(E_{value})$ will be less than 0.7. For multi-band images we usually use the band maximum of the edge value at each image pixel, but we can also use the band average or band minimum on an experimental basis. Another alternative is to compute the edge values for the first principal component image.

4. INCORPORATING EDGE INFORMATION INTO HSEG

We describe below three distinct approaches for incorporating edge information into HSeg. However, each of these approaches utilizes a fast merge region growing approach, first proposed in [6], to initialize the image segmentation based on the edge value information. In our implementation of this approach, we set an edge value threshold, E_t , and merge all neighboring pixels that result in regions where $\max(E_{value}) \leq E_t$.

4.1. HSeg Version 1.61

This version is a simple extension of the original version in that one region feature value, E_{max} , is added to each region, which is the maximum value of E_{value} for all pixels in the region. The weighting of this edge feature is controlled by a user settable parameter, E_w . The edge dissimilarity value, E_{dissim} , is taken to be the maximum of E_{max} for the two regions being compared. We normalize the value of E_{dissim} to range from 0.0 to 1.0, by computing $E'_{dissim} = (E_{dissim} - \min_I[E_{value}]) / (\max_I[E_{value}] - \min_I[E_{value}])$, where $\min_I[E_{value}]$ is the minimum value of E_{value} over the entire image, I , and $\max_I[E_{value}]$ is the maximum value of E_{value} over the entire image. An edge factor, E_f , is then computed as follows:

$$E_f = (S_w + (1.0 - S_w)((1.0 - E_w) + E'_{dissim}E_w)) / S_w$$

The effect of this equation is to set $E_f = 1.0$ for $E_w = 1$ and $E'_{dissim} = 0.0$, and $E_f = 1.0/S_w$ for $E_w = 1.0$ and $E'_{dissim} = 1.0$. The combined region dissimilarity is then computed as $C_{dissim} = R_{dissim} * E_f$, where R_{dissim} is the dissimilarity between the region pair for the original version of HSeg. Thus, an adjacent region is treated as a non-adjacent region for $E'_{dissim} = 1.0$, and treated as an adjacent region for $E'_{dissim} = 0.0$, with gradations in-between for $0.0 < E'_{dissim} < 1.0$.

4.2. HSeg Version 1.71

This version is a more complicated extension of the original version in that a region data structure is modified to enable the tracking of the value of E_{value} along the mutual boundary between two regions. The value of the edge dissimilarity value, E_{dissim} , is taken to be the average of E_{value} for all the mutual boundary pixels between the two regions. The value of E_f is then computed based on this value of E_{dissim} in the same way as for HSeg Version 1.61.

4.3. HSeg Version 1.81

This version is a yet more complicated extension of the original version in that a region data structure is modified to not only enable the tracking of the value of E_{value} along the mutual boundary between two regions, but also determine whether or not a region boundary pixel is adjacent to any

other region. For this version, the value of the edge dissimilarity value, E_{dissim} , is taken to be the average of E_{value} for all the boundary pixels between the two regions that only have the other region as a neighboring region. The value of E_f is then computed based on this value of E_{dissim} in the same way as for HSeg Version 1.61.

5. COMPARATIVE RESULTS

To provide continuity with our previous work, we evaluated the effectiveness of the new versions compared to each other and to the previous version using the region-based classification approach and the test data sets utilized in [1]. See [1] for a detailed description of this approach and the test data sets. In summary, a pixelwise classification based on support vector machine (SVM) is applied to the image data. Then, the region classification is obtained by considering a plurality vote rule, i.e., by assigning each spatially connected region from the segmentation result to the most frequently occurring class within the region. Results are presented here for three hyperspectral data sets: Washington DC Mall HYDICE, University of Pavia ROSIS and Indian Pines AVIRIS.

Tables I, II and III report classification accuracy and segmentation processing time results for the three hyperspectral data sets across the compared segmentation approaches. Classification accuracy is reported in terms of overall accuracy (OA), average accuracy (AA) and kappa coefficient (κ), and processing time is provided in minutes and seconds (min:sec). HSeg parameter settings are provided in table notes (the SAM criterion is used here).

TABLE I.
RESULTS FOR WASHINGTON DC MALL HYDICE

	OA	AA	κ	min:sec
SVM	95.76	95.54	94.64	-
HSWO	96.99	96.02	96.19	6:56
HSeg V1.59	96.95	96.17	96.14	25:46
HSeg V1.61	96.83	95.61	95.99	39:10
HSeg V1.71	97.11	95.53	96.34	73:44
HSeg V1.81	96.45	95.90	95.51	25:24

$S_w = 0.1$, $E_t = 0.0$ and $E_w = 1.0$.

TABLE II.
RESULTS FOR UNIVERSITY OF PAVIA ROSIS

	OA	AA	κ	sec
SVM	89.03	89.56	85.46	-
HSWO	95.38	95.50	93.83	9:33
HSeg V1.59	98.35	98.15	97.79	10:52
HSeg V1.61	97.49	97.08	96.64	14:59
HSeg V1.71	97.21	97.26	96.27	20:25
HSeg V1.81	97.47	97.41	96.61	18:47

$S_w = 0.3$, $E_t = 0.0$ and $E_w = 1.0$.

TABLE III.
RESULTS FOR INDIAN PINES AVIRIS

	OA	AA	κ	sec
SVM	76.41	80.77	72.92	-
HSWO	85.33	86.31	83.07	0:09
HSeg V1.59	86.89	89.83	84.84	1:48
HSeg V1.61	86.50	89.62	84.41	2:04
HSeg V1.71	85.14	89.15	82.80	2:18
HSeg V1.81	86.49	90.98	84.40	3:21

$S_w = 0.1$, $E_t = 0.0$ and $E_w = 1.0$.

For these tests, the Frei-Chen edge operator was applied to the first principal component of the hyperspectral images.

The comparative results on these hyperspectral data sets demonstrate no consistent advantage of the new versions of HSeg utilizing edge information over the previous approach (HSeg V1.59). HSeg V1.71 produces a higher OA and κ for the Washington DC Mall HYDICE data set, at a significant cost in processing time. HSeg V1.81 produces a better AA for the Indian Pines AVIRIS data set, but otherwise HSeg V1.59 produces better results.

The processing time results are highly varied, and demonstrate that the processing time required is highly data dependent. While we expect that the processing times for HSeg V1.71 and HSeg V1.81 will generally be longer than processing times for HSeg V1.59 and V1.61, this isn't always necessarily the case.

In the introduction, we noted that with the previous version of HSeg (V1.59), we often see that large and apparently homogeneous areas are sometimes separated into more than one region with region boundaries that do not correspond to any apparent object boundary in some of the more detailed levels of the HSeg segmentation hierarchy. Even if the new versions of HSeg don't demonstrate a clear advantage in the hyperspectral classification examples, do the new versions of HSeg at least solve this problem?

The Ikonos image displayed in Fig. 1(a) is a suitable test case. This image, acquired on May 17, 2000, contains a large homogenous area, namely the Baltimore, MD inner harbor. In this case, the edge information utilized was the band maximum of the Frei-Chen edge operator value at each image pixel. As we can see in Fig. 1(b), HSeg V1.59 splits the inner harbor up into several regions with region boundaries that do not correspond to any obvious image feature. Figs. 1(c) and 1(d) show two different cases where HSeg V1.61 represents the inner harbor as one single region object. Fig. 1(c) shows that using the fast merge initialization step with $E_t = 0.05$ is sufficient to merge the inner harbor into a single region object. In this case the region growing process employed after initialization was

equivalent to that used in V1.59 since E_w was set to 0.0. We can see the additional effect of region growing incorporating edge information in Fig. 1(d), where E_w was set to 1.0. The clearest effect from incorporating edge information is seen in the Patterson Park area in the upper right of the image, where the grassy areas in Patterson Park are more clearly delineated with $E_w = 1.0$. In each case the segmentation result was taken where the global dissimilarity between the image data and region mean was 0.371. The BSMSE criterion was employed in this test.

Figs. 1(e) and 1(f) show the corresponding results for HSeg V1.71 and V1.81, respectively. The significance of the differences in results between V1.61, V1.71 and V1.81 are not clear from the displayed image segmentation results. In one sense, the HSeg V1.81 results are quite different from the V1.61 and V1.71 results in there are 200 region classes at the point where the global dissimilarity reaches 0.371, whereas there are only 15 and 27 region classes at the equivalent point for versions 1.61 and 1.71, respectively. Further explorations before any firm conclusions can be made concerning the relative effectiveness of these new versions of HSeg incorporating edge information.

6. CONCLUDING REMARKS

In this paper we have proposed three alternate approaches for incorporating the edge information into HSeg image segmentation approach. Some quantitative results from a plurality vote classification approach provide mixed results concerning the effectiveness of the new implementations as compared to the previous version of HSeg. However, we have noted that large homogeneous areas are merged into one region much earlier in the region growing process with the new versions, as was desired. We will continue to evaluate and compare these versions of HSeg on other data sets, noting the tradeoffs between computation time and segmentation quality.

7. REFERENCES

- [1] J. C. Tilton, Y. Tarabalka, P. M. Montesano, and E. Gofman, "Best merge region-growing segmentation with integrated nonadjacent region object aggregation," *IEEE Trans. Geosci. and Remote Sensing*, vol. 50, no. 11, pp. 4454–4467, Nov. 2012.
- [2] W. Frei and C. Chen, "Fast boundary detection: A generalization and new algorithm," *IEEE Trans. Computers*, vol. C-26, no. 10, pp. 988–998, Oct. 1977.
- [3] J.-M. Beaulieu and M. Goldberg, "Hierarchy in picture segmentation: A stepwise optimal approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 2, pp. 150–163, Feb. 1989.
- [4] F. A. Kruse, A. B. Lefkoff, J. W. Boardman, K. B. Heidebrecht, A. T. Shapiro, P. J. Barloon, and A. F. H. Goetz, "The spectral image processing system (SIPS) – Interactive visualization

and analysis of imaging spectrometer data," *Remote Sens. Environ.*, vol. 44, no. 2/3, pp. 145–163, May/June. 1993.

[5] D. Rakos "Frei-Chen edge detector," Rastergrid Blogosphere: <http://rastergrid.com/blog/2011/01/frei-chen-edge-detector/>.

[6] J. L. Muerle and D. Allen, "Experimental evaluation of techniques for automatic segmentation of objects in a complex scene," *Pictorial Pattern Recognition*, G.C. Cheng, R.S. Ledley, D.K. Pollock, and A. Rosenfeld (eds), Thompson Book Co., Washington, D.C., pp. 3–13, 1968.

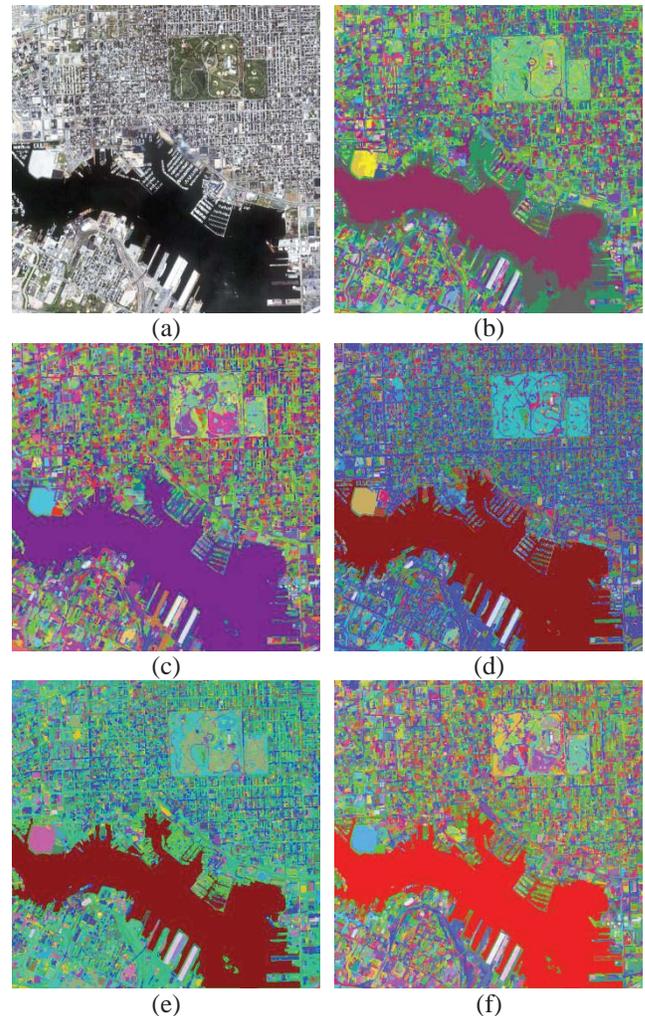


Fig. 1. (a) Ikonos image over the Baltimore, MD Inner Harbor area from May 17, 2000. (b) HSeg V1.59 result at global dissimilarity 0.371 (115 region classes and 9871 region objects). (c) HSeg V1.61 result with $E_t = 0.05$ and $E_w = 0.0$ at global dissimilarity 0.371 (192 region classes and 9954 region objects). (d) HSeg V1.61 result with $E_t = 0.05$ and $E_w = 1.0$ at global dissimilarity 0.371 (15 region classes and 14 513 region objects). (e) HSeg V1.71 result with $E_t = 0.05$ and $E_w = 1.0$ at global dissimilarity 0.370 (27 region classes and 13 306 region objects). (f) HSeg V1.81 result with $E_t = 0.05$ and $E_w = 1.0$ at global dissimilarity 0.371 (200 region classes and 10 785 region objects).