Integrated Nonadjacent Region Object Aggregation with Best Merge region growing Segmentation

K.A.Nithya[1],Research Scholar
Department of Computer Science
Hindusthan College of Arts and Science
Coimbatore , India
nithyamsccs@gmail.com,

G. Delucta Mary[2] U. Pushpa Lingam[3]
Department of Computer Science and Engineering
RVS college of Engineering and Technology
Coimbatore, India
deluctamary@gmail.com , pushpalingam23@gmail.com

***Abstract*- Image segmentation is a first step in the analysis of high spatial resolution remotely sensed imagery using object based image analysis. The segmentation quality is important in the analysis of remotely sensed imagery. Hierarchical image segmentation (HSeg) is a hybrid of region growing and spectral clustering that produces a hierarchical set of image segmentations based on the detected natural convergence point. Computing time of HSeg is high. It can be reduced by recursive version of HSeg (RHSeg). But it cannot be used in high spatial resolution images. So the refined version of HSeg is introduced to reduce the computing time. The computing time is reduced by limiting the region object aggregation to regions containing a dynamically varied minimum number of pixels.**

In HSeg similarity calculation was based on cues (color/texture). To reduce the computing time and increase the segmentation quality in this paper the refined HSeg is extended by adding probabilistic approach known as Bayesian network. The proposed work takes color and shape for similarity measurement. Execute a sequence of bottom up aggregation steps in which pixels are gradually merged to produce larger and larger regions. In each step consider pairs of adjacent regions and provide a probability measure to access whether they should be included in the same segment or not. The probabilistic formulation takes into account intensity, color, texture distribution in a local area around each region. Finally posteriors based on color, shape and texture combined. It improves the segmentation accuracy and reduces time.

***Index Terms*-** Hierarchical segmentation, Bayesian network, Spectral clustering, cues, natural convergence point.

I. INTRODUCTION

Image segmentation is a fundamental yet still a challenging problem in computer vision and image processing. In particular, it is an essential process for many applications such as object recognition, target tracking, content-based image retrieval, and medical image processing. For remotely sensed images of the Earth, an example is a map that divides the image into areas labeled by distinct Earth surface covers such as water, snow, and types of natural vegetation, rock formations, crops, and other man-created objects. Generally the goal of image segmentation is to partition an image into a certain number of pieces that have coherent features (color, texture, etc.) and, in the meanwhile, to group the meaningful pieces together for the convenience of perceiving.

Most image segmentation approaches can be placed in one of three categories [1]:

a. Characteristic feature Thresholding or clustering
b. Boundary detection
c. Region growing

Characteristic feature Thresholding or clustering does not exploit spatial information, and thus ignores information that could be used to enhance the segmentation results. While boundary detection does exploit spatial information by examining local edges found throughout the image data, it does not necessarily produce closed connected region boundaries. For simple noise-free data, detection of edges usually results in straightforward region boundary delineation. However, edge detection on noisy, complex image data often produces missing edges and extra edges that cause the detected boundaries to not necessarily form a set of close connected curves that surround connected regions. Region growing approaches to segmentation are preferred here because region growing exploits spatial information and guarantees the formation of closed, connected regions. In image analysis, the group of image data points contained in each region provides a statistical sampling of image data values for more reliable labeling based on image feature values.

With the spatial resolution of remotely sensed imagery increased, traditional pixel-based remote sensing analysis may have some limits, which leads to the development of an object-based image analysis (OBIA) method [2]. Image segmentation is the first step of OBIA. It is to partition an image into meaningful homogeneous regions corresponding to real world objects. The effectiveness of OBIA is directly affected by the segmentation quality. Hence, an evaluation of segmentation results is very important for the subsequent analysis. OBIA is a key factor in determining the level of performance for these image analysis approaches.

A popular approach for performing image segmentation is best merge region growing The principle of best merge is given below.

1. Define a (dis)similarity criterion for pairs of regions
2. .Define a stopping criterion
3. While stopping criterion is not met do
   i. Compute for all adjacent pairs of regions their similarity value.
   ii. From these merge the single most similar pair.

The best merge region growing approach was first fully described in the archival literature by Beaulieu and Goldberg [3]. In this approach proposed the hierarchical stepwise optimization (HSWO), which employs a sequence of optimization processes to produce hierarchical segmentation results of different levels of details, and has been widely used for analysis of remote sensing images. HSWO is best defined iteratively: Start with an image and a segmentation of that image into *N* regions in which (i) every picture element (pixel) is in a region, (ii) and each region is connected, (*i.e.* composed of contiguous image pixels). Then compare all spatially adjacent regions with each other (*e.g.*, compute a vector norm between the region means of the spatially adjacent regions). Merge the most similar pair of spatially adjacent regions. Continue to compare spatially adjacent regions and merge the most similar pair of spatially adjacent regions until either a specified number of regions are reached or the dissimilarity between the most similar pair of spatially adjacent regions reaches a specified threshold.

Similar approaches of best merge region growing described earlier in conference proceedings[4]-[7]. Similar approaches of best merge region growing described earlier in conference proceedings [4]–[7]. Many variations on best merge region growing have been described in the literature. As early as 1994, Kurita [8] described an implementation of HSWO that utilized a heap data structure [9] for efficient determination of best merges and a dissimilarity criterion based on minimizing the mean squared error between the region mean image and original image.

SEGEN [10] is an efficient region growing algorithm for the segmentation of multi-spectral images in which the complexity of the most time-consuming operation in region growing, merging segment neighborhoods, is significantly reduced. In addition, considerable improvement is achieved by preprocessing, where adjacent pixels with close colors are gathered and used as initial segments. The preprocessing provides substantial memory savings and performance gain without a noticeable influence on segmentation results. In practice, there is an almost linear dependency between the runtime and image size. It is relatively pure implementation of best merge region growing, optimized for efficiency in performance, memory utilization, and image segmentation quality. The process for selecting the best merges is much more involved than the relatively straightforward evaluation and comparison of region dissimilarity functions utilized by HSWO and SEGEN.

There are several drawbacks in the image segmentation of remotely sensed imagery. In some places sensor noise or irregularities in land cover features (*e.g.* too much bare soil showing through a vegetation canopy in one small area of a field) also leaves isolated pixels in the middle of otherwise homogeneous segments. This problem frequently occurs in remotely sensed imagery. Other problems of segmentation are: may not preserve spatial relationships, potentially high computational complexity, Segmentation primarily uses color intensity, Single condition for when to stop segmentation and segmentation result is non-optimal in which uncertainties exist. If the similarity threshold is set too low the growing process will leave many pixels unassigned to segments. If the similarity threshold is too high, segments representing different land cover parcels will be incorrectly merged together. Another problem occur in remotely sensed imagery is the within –field variation. Due to the natural causes for example wet spots, dry spots, different soil types etc the spectral of neighboring pixels are not necessarily similar. To overcome these drawbacks hybrid technique is introduced. In this paper proposed the hierarchical image segmentation. It is the hybrid of region growing and spectral clustering.

In complex scenes, such as remotely sensed images of the Earth, objects with similar spectral signatures (e.g., lakes, agricultural fields, buildings, etc.) appear in spatially separated locations. In such cases, it is useful to aggregate these spectrally similar but spatially disjoint region objects together into groups of region objects that we call region classes. This aggregation may be performed as a postprocessing step. However, best merge region growing, as exemplified by HSWO, may be modified to integrate this aggregation directly into the region growing process. This is the basis of our hierarchical segmentation (HSeg) algorithm.

The approach taken for spatially disjoint region object aggregation requires excessive computing time in the original formulation of HSeg. A recursive divide-and conquer approach, called recursive HSeg (RHSeg), was previously developed to overcome this computational problem. In this paper, introduce for the first time a refined implementation of nonadjacent region object aggregation in HSeg that reduces the computational requirements of HSeg without resorting to the recursive approximation. The key idea of this refinement is that region object aggregation is limited to region objects containing no less than a dynamically specified minimum number of image pixels.

The HSWO, HSeg, and RHSeg algorithms naturally produce a segmentation hierarchy in the form of a set of several image segmentations at different levels of detail in which the segmentations at coarser levels of detail can be produced from simple merges of regions at finer levels of detail. This hierarchy may be useful for applications that require different levels of image segmentation details depending on the characteristics of the particular image objects segmented. A unique feature of a segmentation hierarchy that distinguishes it from most other multilevel representations is that the segment or region boundaries are maintained at the full image spatial resolution for all levels of the segmentation hierarchy.

This paper is organized as follows. First, we provide a full description of the original HSeg and RHSeg algorithms. Then introduce refinement of HSeg and note how this refinement of HSeg impacts RHSeg. Next introduce refined HSeg with Bayesian network. Using this proposed method the computation time is considerably reduced as shown in the segmentation and also shown that the segmentation quality also increased. The computational demands of HSWO, the original HSeg, the RHSeg utilizing the original HSeg, the refined HSeg algorithm, compared using different remote sense imagery. Next, evaluate image segmentation quality. Then show that the refined HSeg algorithm leads to improved flexibility in segmenting moderate- to large sized high spatial resolution images. results for the refined version of HSeg with similar classification results from HSWO, SEGEN.

II. ORIGINAL HSEG

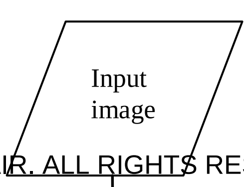
The hierarchical image segmentation approach described herein, called HSeg, is a hybrid of region growing and spectral clustering that produces a hierarchical set of image segmentations based on detected natural convergence points[11]. A hierarchical set of image segmentations is a set of several image segmentations at different levels of segmentation detail in which the segmentations at coarser levels of detail can be produced from simple merges of regions from segmentations at finer levels of detail. Maintaining region boundaries at full image spatial resolution avoids compounding the "mixed pixel" problem which adversely impacts other multiresolution segmentation schemes in which the coarser resolution segmentations are produced from spatially degraded versions of the imagery data.

HSeg is the same as HSWO, except that HSeg optionally alternates merges of spatially adjacent regions with merges spatially non-adjacent regions. In addition, HSeg also offers a wide choice of cost functions. Currently implemented are cost functions based on vector norms (1-norm, 2-norm and infinity-norm), and mean squared error. Other cost functions can be implemented (e.g. statistical hypothesis testing, constraining image entropy, normalized vector distance, and others).

The HSeg algorithm is very computationally intensive, and cannot be performed in a reasonable amount of time (less than a day) on moderately sized data sets, even with the most 2 powerful (single processor) computer currently available. For example, for a 6-spectral band Landsat TM image, a 128x128 pixel section takes about 25 minutes to process on a 1.2 GHz single processor computer. A 256x256 pixel section of the same image takes over 7.5 hours to process on the same computer. By extrapolation, a 512x512 pixel section of the same image would easily take several days.

The hierarchical segmentation algorithm extends to hyperspectral images . The original HSeg algorithm augments best merge region growing with the inclusion of constrained merging of spatially nonadjacent regions. In Hierarchical segmentation nonadjacent region objects merging are controlled by the input parameter S_{wght}. This parameter values adjust from 0.0 to 1.0. The algorithm is as follows

1) Initialize the segmentation by assigning each image pixel a region label. If a presegmentation is provided, label each image pixel according to the presegmentation. Otherwise, label each image pixel as a separate region.
2) Calculate a dissimilarity criterion value *d* between all pairs of regions (if S_{wght} = 0.0, the dissimilarity criterion only needs to be calculated between all pairs of spatially adjacent regions).
3) Set the merge threshold T_{merge} equal to the smallest dissimilarity criterion value *d* between pairs of spatially adjacent regions.
4) Merge pairs of spatially adjacent regions with $d = T$merge.
5) If $S_{wght} > 0$, merge pairs of nonadjacent regions with $d \leq S_{wght} \cdot T_{\text{merge}}$.
6) Output the segmentation result if the output criterion is satisfied (more on this later).
7) Stop if convergence has been achieved. Otherwise, go to step 8. Convergence is normally considered to be achieved when a specified number of regions have been reached (by default, two regions).
8) Update the dissimilarity criterion values *d* for the regions affected by merges, and return to step 3.



No

Yes

Fig 1. Flowchart of HSeg Algorithm

Since segmentation results with a large number of regions are usually severely over segmented and thus not of interest, HSeg does not normally output the hierarchical segmentation results until the number of regions reaches a user-specified value (by default, 255 regions). After that point, HSeg normally outputs a subsequent hierarchical segmentation result at the iteration just prior to the iteration at which any region would be involved in more than one merge since the last result was output. Alternatively, HSeg can be set to output hierarchical segmentation at a user specified list of number of regions or list of merge thresholds. One can select from a number of criteria for evaluating how dissimilar one region is from another in HSeg. These dissimilarity criteria include criterion based on vector norms, minimizing the mean square error difference or the change in entropy between the region mean image and the original image, among others ([12]).

When *Swght* = 0.0, spatially nonadjacent region merges (step 5) are not performed, and HSeg reduces to straightforward best merge region growing. This serves as implementation of HSWO. With S_{wght} = 1.0, merges between spatially adjacent and spatially nonadjacent regions are given equal priority. For values of S_{wght} between 0.0 and 1.0, spatially adjacent merges are given priority over spatially nonadjacent merges by a factor of $1.0/S_{wght}$. Thus, for *Swght* > 0.0, region objects (i.e., spatially connected regions) may be aggregated into spatially disjoint groupings that called region classes.

What regions are considered to be spatially adjacent to other regions depends on the definition of a neighborhood relationship. HSeg use the usual *n*-nearest neighbor concept to define spatial adjacency for image pixels, most commonly four nearest neighbors (north, south, east, and west;
referred to as 4 nn) or eight nearest neighbors (including the diagonal pixels; referred to as 8 nn). Regions adjacent to a region are the union of the region memberships of the neighbors of the pixels on the boundary of that region.

Benefits of hierarchical image segmentation are improved analytical capabilities, increased speed, refined results, maximized flexibility and control, increased accuracy, enhanced ease of use and the applications are aircraft or satellite remote sensing, monitoring agricultural crops, identifying buildings and roadways, determining population densities and areas with the greatest growth, analyzing ground-penetrating radar data, medical imaging and Chest imaging screening for lung cancer, computer-aided detection (CAD), cervical cancer imagery, computed tomography (CT) scans, magnetic resonance imaging (MRI), and ultrasound imagery and X-ray image analysis, image data mining and knowledge discovery and feature searches in large image database ,image data fusion, facial recognition, sonar and radar data analysis etc.

III. RECURSIVE HSEG

With the addition of alternating iterations of spectral clustering in the HSEG algorithm, the computational demands significantly increase. This is caused primarily because of the requirement to update or calculate the dissimilarity criterion values for all pairs of regions in steps 2 and 8. For a 1024 x 1024 pixel image, this leads to the order of 10000000 comparisons in the initial processing stage. Nevertheless, this computational obstacle is surmounted by the recursive formulation of the HSEG algorithm, RHSEG. This recursive form not only limits the number of comparisons between spatially non-adjacent regions to a more reasonable number, but also lenses itself to a straightforward and efficient implementation on parallel computing platforms. In regards to the definition of RHSEG, it follows the same as the RHSWO and includes the definition of processing window artifact elimination.

The recursive formulation of HSEG (RHSEG), however, can process moderately sized images in a reasonable amount of time on currently available PCs and workstations.RHSeg was an excellent choice because it provided the image segmentations required for input, based on three key factors: (1) the high spatial fidelity of image segmentations produced by RHSeg, (2) the ability of RHSeg to automatically group spatially connected region objects into region classes, and (3) the hierarchical set of image segmentations that RHSeg automatically produced. The Algorithmic Description of RHSeg:

1. Specify the number of levels of recursion required (rnb_levels) and pad the input data set, if necessary, so the width and height of the data set can be evenly divided by $2^{\text{rnb\_levels-1}}$. Set level = 1.
2. Call recur_hseg(level,data).
3. Execute the HSeg algorithm using as a pre-segmentation the segmentation output by the call to rhseg() in step 2. (Continue executing HSeg past the point that the number of regions reaches chk_nregions and save the segmentation results as specified.)

Outline of recur_hseg(level,data):

1. If level = rnb_levels, go to step 3 below. Otherwise, divide the data set into four equal subsections and call recur_hseg (level+1, sub_data) for each subsection of the data set.
2. After the calls to recur_hseg( ) for each data set subsection from step 1 complete processing, reassemble the data segmentation results.
3. Execute the HSeg algorithm as described in the HSeg Algorithm Description above with the following modification: Terminate the algorithm when the number of regions reaches the preset value min_nregions (if level = 1, terminate at the greater of min_nregions or chk_nregions) and do not check for critval or output any "raw" segmentation results.

The above divide-and-conquer approach limits the number of regions that are processed at any time in step 4 of the HSeg algorithm. This limit leads to a significant reduction in processing time versus the non-recursive approach for even relatively small data sets. An efficient parallel implementation of RHSeg leads to additional significant reduction in processing time.

Another problem emerges when the RHSeg algorithm is used to process moderate to large images. Processing window artifacts may arise from the recursive division of the image data into four equal subsections. This artifacts can be eliminated, however, by the addition of a fourth step to the outline of recur_hseg(level,data) as follows:

4. If level = rnb_levels, exit. Otherwise, switch the region assignment of certain pixels in the following manner: For each region, determine which other regions may contain pixels that may more similar to it than the region to which they are currently assigned. Then for each of these regions compute the dissimilarity each pixel contained in the region to its current region (own_region_dissim) and to each region to which it may potentially be more similar. If a pixel is found to have own_region_dissim>switch_pixels_factor*other_region_dissim, switch the region index for

that pixel to the region with the minimum other_region_dissim value Exit.

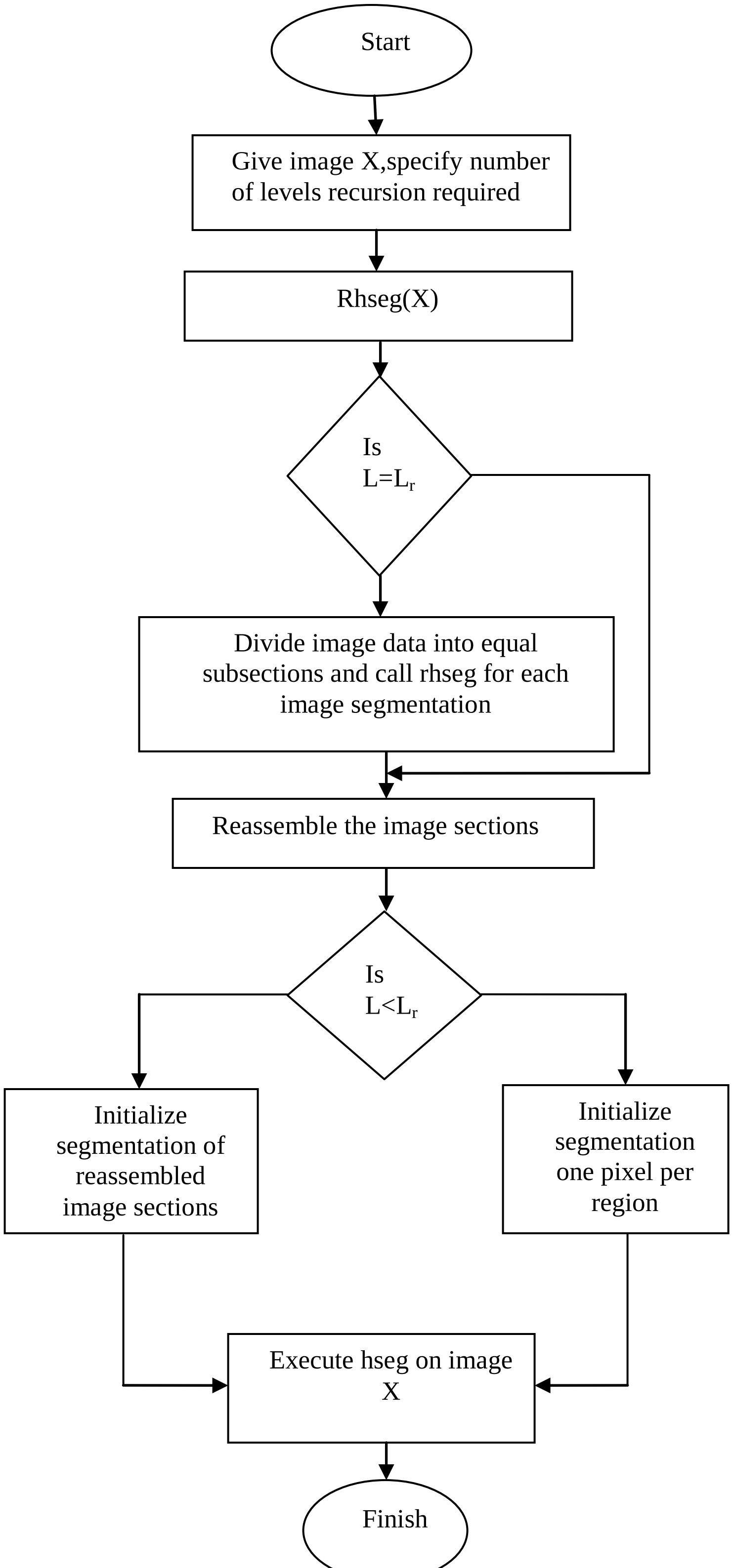


Figure2. Flow chart of RHSeg

IV. REFINED HSEG

In this refined implementation of nonadjacent region object aggregation in HSeg that reduces the computational requirements of HSeg without resorting to the recursive approximation. In this refinement, HSeg's region intercomparisons among nonadjacent regions are limited to regions of a dynamically determined minimum size. This refined version of HSeg can process moderately sized images in about the same amount of time as RHSeg incorporating the original HSeg. The refined HSeg algorithm leads to improved flexibility in segmenting moderate- to large sized high spatial resolution images.

Initially set P_{min} to the smallest value such that $N_{large} \leq S_{max}$. If this results in $N_{large} < S_{min}$, the value of P_{min} is reduced by one (unless it is already equal to one), and the value of N_{large} with this new value of P_{min} is determined. If this new value of P_{min} results in $N_{large} > 6 \cdot S_{max}$, the value of P_{min} is incremented back up by one. Finally, if this later adjustment results in $N_{large} < 2$, the value of P_{min} is again reduced by one, regardless of whether this results in $N_{large} > 6 \cdot S_{max}$. Whenever the value of P_{min} is changed, "local" values of S_{max} and S_{min} are determined (call them S_{max} and S_{min}), and the value of P_{min} is checked only when the number of "large regions" becomes less than S_{min} (and the value of P_{min} is more than one) or becomes larger than S_{max}. This prevents performing unnecessary computations when it is unlikely that the value of P_{min} would be changed. The values of S_{min} and S_{max} are recalculated whenever P_{min} is checked for adjustment. For S_{min}, let $S_{min} = N_{large}$. However, if $N_{large} \leq S_{max}$, compute temp $= S_{max} - 2 \cdot (S_{max} - N_{large})$, and if temp $> S_{min}$, let S_{min} = temp. If $S_{min} > Nr$ (the current number of regions, both "large" and "small"), let $S_{min} = Nr$. Compute max $S_{min} = S_{max} - 0.05 \cdot (S_{max} - S_{min})$. If $S_{min} >$ max S_{min}, let $S_{min} = maxS_{min}$. For S_{max}, if $N_{large} > S_{max}$, let $S_{max} = N_{large}$. Otherwise, let $S_{max} = S_{max}$. Like the original versions, the refined version of HSeg includes an option for small region merge acceleration.

V. PROPOSED METHOD

REFINED WITH BAYESIAN HSeg

In the existing method the segmentation quality and reduction of processing time was improved using different algorithm. The proposed method also used to reduce the processing time and increase the segmentation quality better than the existing method. In this better performance is obtained by using last existing method (refined HSeg) is added to the Bayesian network. That is the refined version of hierarchical image segmentation is added to the Bayesian network. Using the Bayesian network the similarity calculation was performed.

Bayesian networks (BNs), also known as belief networks (or Bayes nets for short), belong to the family of probabilistic graphical models (GMs). BNs became extremely popular models in the last decade. They have been used for applications in various areas, such as machine learning, text mining, natural language processing, speech recognition, signal processing, bioinformatics, error-control codes, medical diagnosis, weather forecasting, and cellular networks..

Bayesian refers to methods in probability and statics. Bay's theorem gives the relationship between the probabilities. Bayesian probability is one of the different interpretations of the concept of probability and belongs to the category of evidential probabilities. The Bayesian interpretation of probability can be seen as an extension of propositional logic that enables reasoning with propositions whose truth or falsity is uncertain. The Bayesian interpretation provides a standard set of procedures and formulae to perform this calculation. For example probabilities of A and B is P(A) and P(B) and the conditional probabilities of A given B and B given A, P(A|B) and P(B|A). In its most common form, it is:

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$

The Bayesian methods are characterized by the following concepts and procedures:

1)The use of random variables to model all sources of uncertainty in statistical models. This includes not just sources of true randomness, but also uncertainty resulting from lack of information. 2) The sequential use of the Baye's formula: when more data become available after calculating a posterior distribution, the posterior become the next prior. 3) For the frequentist a hypothesis is a proposition. So that the frequentist probability of a hypothesis is either one or zero. In Bayesian statistics, a probability can be assigned to a hypothesis that can differ from 0 or 1 if the true value is uncertain.

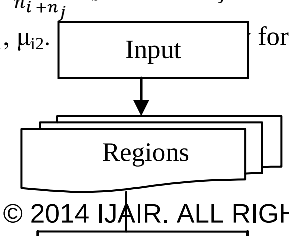
The Bayesian region merging probability is a significant contribution since: In the presence of uncertainty, when parameter estimates are poor, the Bayesian region merging probability gives an appropriate measure of the likelihood of merging two regions. The formalism applies to a wide class of statistical image models. Since the approach is Bayesian, a straightforward extension to multiple, independent image models are available. The formalism applies to a wide class of statistical image models.

a)Dissimilarity Criterion

The dissimilarity criterion is important for this approach. Selection of an appropriate dissimilarity criterion is generally dependant on the application the resulting segmentations will be used for, and on the characteristics of the image data. Nevertheless, a few general purpose similarity criteria for use with this algorithm, including criteria based on minimizing mean-square error and minimizing change in image entropy, and the "Normalized Vector Distance". One dissimilarity criterion is based on minimizing the increase of mean squared error between the region mean image and the original image data. The BSMSE between regions X_i and X_j with region mean vectors u_i and u_j and region size (number of pixels) n_i and n_j is given by

$$d_{BSMSE}(X_i,X_j)=\frac{n_i n_j}{n_i+n_j}\sum_{b=1}^{B}(\mu_{ib}-\mu_{jb})^2$$

Where $u_i = (\mu_{i1}, \mu_{i2}.$ for u_j).



No

Yes

Figure 3. Flow chart of proposed method

VI. CONCLUSION AND FUTURE WORK

In this paper explained different version of hierarchical image segmentation. The computing time of original version of HSeg is high. So the recursive HSeg was introduced to overcome the drawback of HSeg. Then the refined HSeg was proposed for efficient segmentation. In this paper refined HSeg with Bayesian network was proposed and processed using MATLAB.

The performance of the different version of hierarchical image segmentation is analyzed. The proposed method is compared to the existing method using the time consumption graph. The proposed method reduces the processing time. The different feature probabilities such as shape, color, texture are classified and merged accordingly. The proposed method limits the region object aggregation step to region, so the speed of process is increased. As a result of proposed method segmentation is useful for remotely sensed imagery and it can overcome the order dependence problem.

In future RHSeg utilizing the refined version of Hseg is still needed to process large images due to its lower needs for computer memory and the availability of a straightforward parallel implementation and increase the segmentation accuracy using NPR, then compare to existing method.

REFERENCES

[1] K. S. Fu and J. K. Mui, "A survey on image segmentation," *Pattern Recognition*, Vol 13, pp. 3-16, 1981.
[2] T. Blaschke, S. Lang, and G. J. Hay, Eds., "Object-Based Image Analysis: Spatial Concepts for Knowledge-Driven Remote Sensing Applications". Berlin, Germany: Springer-Verlag, 2008.
[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] J.-M. Beaulieu and M. Goldberg, "Hierarchical picture segmentation by approximation," in *Proc. Can. Commun. Energy Conf.*, Montreal, QC, Canada, 1982, pp. 393–396.
[5] J. C. Tilton and S. C. Cox, "Segmentation of remotely sensed data using parallel region growing," in *Proc. IGARSS*, San Francisco, CA, 1983, pp. 130–137.
[6]M.Willebeek-LeMair and A. P. Reeves, "Region growing on a highly parallel mesh-connected SIMD computer," in *Proc. 2nd Symp. Frontiers Massively Parallel Comput.*, Fairfax, VA, 1988, pp. 93–100.
[7] J. C. Tilton, "Image segmentation by iterative parallel region growing with applications to data compression and image analysis," in *Proc. 2nd Symp. Frontiers Massively Parallel Comput.*, Fairfax, VA, 1988, pp. 357–360.
[8] T. Kurita, "An efficient agglomerative clustering algorithm for region growing," in *Proc. MVA, IAPR Workshop Mach. Vis. Appl.*, Kawasaki,
Japan, Dec. 13–15, 1994, pp. 210–213.
[9] J.W. J.Williams, "Heapsort," *Commun. ACM*, vol. 7, no. 12, pp. 347–348,Dec. 1964.
[10] E. Gofman, "Developing an efficient region growing engine for image segmentation," in *Proc. ICIP*, Atlanta, GA, Oct. 2006, pp. 2413–2416.
[11]J. C. Tilton, "Image segmentation by region growing and spectral clustering with a natural convergence criterion," in *Proc. IGARSS*, Seattle, WA,1998, pp. 1766–1768.
[12]J. C. Tilton, *RHSeg User's Manual: Including HSWO, HSeg, HSegExtract, HSegReader and HSegViewer*, version 1.55, Jan. 4, 2012.
[13] Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Multiple spectral-spatial classification approach for hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 4122–4132, Nov. 2010.
[14] G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, Jun. 2005.
[15] Y. Tarabalka, J. A. Benediktsson, and J. Chanussot, "Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 8, pp. 2973– 2987, Aug. 2009.

