ISSN 1512-1232

AN ACCURATE IMAGE SEGMENTATION USING REGION SPLITTING TECHNIQUE

[1] Dr.P.Raviraj, [2] Angeline Lydia, [3] Dr.M.Y.Sanavullah

[1] Assistant Professor, Dept. of IT, Karunya University, Coimbatore, TN, India.
[2] PG Scholar, Dept. of CSE, Karunya University, Coimbatore, TN, India
[3] Professor, Dept. of EEE, VMKV Engineering College, TN, India

Abstract

This paper aims to show the good accuracy in an image segmentation using split and merge method. In an existing MRF (Markov Random Field) based unsupervised segmentation, the MRF model parameters are typically estimated. Those global statistics are far from accurate for local areas if the image is highly non-stationary, and hence will generate false boundaries. So, the proposed region splitting method provides the possibility of building a hierarchical representation of the image content and allows various region features and even domain knowledge to be incorporated in the segmentation process. The algorithm has been successfully tested on several artificial images. Thus our proposed method is an improvement over the method called the iterative region growing using semantics (IRGS).

Keywords: *MRF (Markov Random Field), IRGS (Iterative Region Growing using Semantics)*

1. Introduction

Image segmentation is the process of segmenting an image into several disjoint regions whose characteristics such as intensity, color, texture etc, are similar. It is a key step in early vision problem and it has been widely investigated in the field of image processing. A large number of segmentation techniques are available in the literature. But, there does not existing a general algorithm that can excellently perform the segmentation task even for all light intensity images, which is the most common type. Available segmentation techniques include thresholding, region growing, clustering, classifier, neural network-based approaches, deformable models, MRF model based approaches. Thresholding does not take into account the spatial characters of an image. This makes it sensitive to noise. The primary disadvantage of region growing is that it requires manual interaction to obtain the seed point. Clustering algorithms do not require training data, but they do require an initial segmentation. Classifier methods are pattern recognition techniques to partition a feature space derived from the image using data with known labels.

The methodology of using MRF models to the problem of segmentation has emerged later and has created a lot of interest. Markov random fields have been, and are increasing being used to model a prior beliefs about the continuity of image features such as region labels, textures, edges and so on [1]. The main disadvantage of MRF-based methods is that the objective function associated with most nontrivial MRF problems is extremely non-convex and as such the minimization problem is computationally very taxing. To reduce the computational burden, some approaches based on multi resolution techniques have been reported [2]. The essence of MRF frame work on multi-resolution is that it starts processing images at a coarse resolution, and then progressively refines them to finer resolution.

In this paper, we use Split and merge method which is an improvement over the IRGS. The existing method was used for improved image segmentation. Segmentations of images having larger floe sizes are also investigated. As expected, the IRGS have higher accuracies with the growth of the floe sizes. However, such a tendency is not obvious for other methods as those pixel-based methods typically locate local minima and are less sensitive to the correctness of the prior model. The existing IRGS method is deterministic and the merging step is not reversible. Fast speed

ISSN 1512-1232

is achieved at the expense of accuracy. More accurate results may be obtained with the incorporation of a region splitting process.

The paper is organized as follows: Section 2 presents existing method for image segmentation using Iterative Region Growing using semantics. Section 3 the proposed split and merge algorithm is discussed. In section 4 the experimental results are given. The final conclusion of the paper is given in section 5.

2. Review of Basic Systems

We begin with a brief overview of the previous model i.e., Iterative Region Growing using semantics (IRGS) [3].The image segmentation method named iterative region growing using semantics (IRGS), is characterized by two aspects. First, it uses graduated increased edge penalty (GIEP) functions within the traditional Markov random field (MRF) context model in formulating the objective functions. Second, IRGS uses a region growing technique in searching for the solutions to these objective functions.

More generally, the MRF can be defined on irregular graphs rather than the regular image lattice. This allows the image segmentation problem to be based on a set of interconnected groups of pixels (referred to here as “regions”), with the MRF spatial context model based on a region adjacency graph (RAG) [9]. Here, the labeling is not on single pixels but on regions, where the regions are commonly obtained by a deliberate over segmentation. Each node in the RAG represents a region and a link between the nodes represents the existence of a common boundary between the regions. Defined on the RAG, the MRF models the behaviors of the regions in a similar way as for pixel.

2.1 Graduated Increased Edge Penalty (GIEP)

This section focuses on one aspect of the proposed IRGS method: using a sequence of edge penalty functions to approximate the MRF spatial context model. This aspect alone provides a basis for a novel segmentation method and it will be referred to as the graduated increase edge penalty (GIEP). The GIEP has two attractive features. First, it utilizes edge information to improve the segmentation on non stationary situations. Second, it provides a simple and elegant method of simultaneously estimating model parameters and searching solutions for the MRF-based formulation.

The MRF spatial context model functions as a penalty for the existence of boundary site pairs. Instead of penalizing equally for all boundary site pairs, a greater penalty can be applied to weak edge and a lesser penalty to strong edge so that local statistic, such as edge strength, can be incorporated. Therefore, the penalty term can be replaced with some monotonically decreasing function g(.) (defined below) of the strength of the edge between the two neighboring sites astride the boundary.

The edge penalty function g(.) can be any monotonically decreasing function so that the greater the edge strength is, the smaller the penalty. Suppose the edge strength for ∇_{st} any boundary site pair s and t has been normalized to [0, 1].Then, the penalty function can be formulated as

$$g(\nabla_{st}) = e^{-(\nabla_{st}/K)^2} \tag{1}$$

The parameter K defines how fast the edge penalty decays with the increase of edge strength. As K increases, the penalty difference between weak and strong edges decreases. When K approaches infinity, all edge penalties are equally 1.

2.2 Iterative Region Growing Using Semantics (IRGS)

In this section, we extend the GIEP method in a hierarchical manner, leading to the IRGS method. Many researchers have applied the MRF model on a hierarchical structure of pixels instead of single pixels [4], [5], [6], [7], since pixel-based methods are either easily trapped in local minima or extremely slow in convergence. However, the fixed hierarchical structure in those approaches

ISSN 1512-1232

does not allow efficient descriptions of the underlying image contents and may produce undesirable artifacts. There are a number of papers that construct data-adaptive structures using graph cuts [8], [9] or region growing [10] techniques. The sequence of objective functions described in the preceding section are introduced naturally as the merging criteria corresponding to various scales and, here, a simple region growing technique is used to construct the hierarchical data-adaptive structures of the image for optimization purposes.

2.3 Region Growing

Similarly to the GIEP, the IRGS method is also iterative. For each iteration, an optimization process for finding a local minimum of the objective energy function for a given K. Consider a hierarchical clustering process for such an optimization purpose. The process begins with a deliberate over segmentation configuration with many classes and tries to reduce to the true number of classes. Examining each pair of classes, the energy of the configuration obtained by merging the two classes is computed and compared with that before merging. If there is a decrease, the merging is justified. In computing such an energy difference, energy terms unrelated to the current two classes can be cancelled and a simple merging criterion can be derived.

3. Proposed Methodology

Pure merging methods are, however, computationally expensive because they start from such small initial regions (individual points). We can make this more efficient by recursively splitting the image into smaller and smaller regions until all individual regions are coherent, then recursively merging these to produce larger coherent regions. First, we must split the image. Start by considering the entire image as one region.

1. If the entire region is coherent (i.e., if all pixels in the region have sufficient similarity), leave it unmodified.

2. If the region is not sufficiently coherent, split it into four quadrants and recursively apply these steps to each new region.

The characteristics of the split and merge algorithm is that is based on the use of the segmentation tree, which is normally a quad tree. Each node say k, in the tree corresponds to a square region of the image, and has stored with it a maximum and minimum brightness value. Merging is done by comparing adjacent groups of four nodes with a common parent. If they obey the criterion they are replaced by their parent node. Similarly splitting is done by breaking non uniform nodes into the four children nodes at the lower level. The process is constrained by the tree structure, so, when all the splitting and merging is complete, the final outset may be grouped into regions.

3.1 An Efficient Image Segmentation using Region Splitting Technique

The basic idea of region splitting is to break the image into a set of disjoint regions which are coherent within themselves. If only a splitting schedule is used then the final segmentation would probably contain many neighboring regions that have identical or similar properties. Thus, a merging process is used after each split which compares adjacent regions and merges them if necessary. Algorithms of this nature are called split and merge algorithms. The merge-split routine is an optional stage of our region growing based segmentation scheme. It requires a threshold as an input. This threshold determines which blocks can be merged into a single block and which blocks can be split into smaller blocks based on the difference between the maximum and minimum intensities in each block. If the max-min difference of a block is close to the max-min difference of its neighbors (i.e., difference between blocks is within the threshold), then the blocks are merged into a single block. A block is split in half if the max-min difference of the block exceeds the threshold.

The merge-split mechanism is a quad tree structure, meaning that the merging and splitting of blocks goes from 4 to 1 and 1 to 4 respectively. This process is done recursively until, no blocks

ISSN 1512-1232

satisfy the criteria to be split or merged. Thus a block whose max-min difference exceeds the threshold will continue to be split until the max-min difference of the subsequent block(s) are within the threshold or the block size reaches one pixel, in which case the max-min difference is zero. There is also a minimum block size argument which allows the user to specify the smallest block size that can be generated through splitting. This allows the user to force the segmenting algorithm to end up with a small number of regions by ensuring that the output of the merge-split algorithm has blocks that are no smaller than a specified size. Without this feature there is a potential for the merge-split routine to return many small blocks. If these blocks are not successfully merged by the region growing algorithm, undesirable results are likely.

The algorithm for the proposed region splitting method is as follows.

Algorithm:

Step 1: Define an initial segmentation into regions.

Step 2: If a region R in the pyramid data structure them. If no region can be is not homogeneous, split it into child regions. If any four regions with the same parent can be merged into single homogeneous region merge them. If no region can be split or merged goto step 3.

Step 3: If there are any two adjacent regions R_i, R_j (even if they are in different pyramid level or do not have the same the same parent) that can be merged into homogeneous regions merge them.

Step 4: Merge small regions with the most similar adjacent region if it is necessary to remove small size regions.

Figure 1. Proposed Region-Splitting Algorithm

The overall flowchart for the improvement of accuracy using the region splitting method is as below. It gives the information about the work process which should be done in a sequential manner.

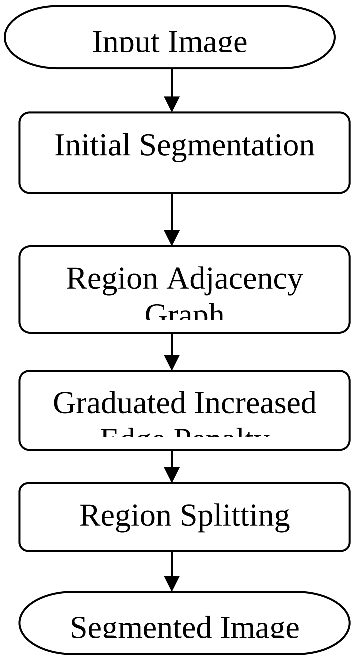


Figure 2. Flow diagram about the work done

ISSN 1512-1232

3.2 Region Splitting

The basic idea of region splitting is to break the image into a set of disjoint regions which are coherent within themselves:

- Initially take the image as a whole to be the area of interest.
- Look at the area of interest and decide if all pixels contained in the region satisfy some similarity constraint.
- If TRUE then the area of interest corresponds to a region in the image.
- If FALSE split the area of interest (usually into four equal sub-areas) and consider each of the sub-areas as the area of interest in turn.
- This process continues until no further splitting occurs. In the worst case this happens when the areas are just one pixel in size.
- This is a *divide and conquer* or *top down* method.

If only a splitting schedule is used then the final segmentation would probably contain many neighboring regions that have identical or similar properties. Thus, a *merging* process is used after each split which compares adjacent regions and merges them if necessary. Algorithms of this nature are called *split and merge* algorithms.

The following are the steps that are used in image splitting

3.2.1 Input Image

At first an image is taken as input as shown in Figure 3. Here SAR sea ice image is taken as input. A satellite SAR system is able to provide a continuous and regular imaging of the ice-field over extended areas. The SAR backscatter is mainly dependent on two factors the electrical property of the target ice and the roughness of the ice surface. Modeling of the SAR backscatters of sea ice is thus extremely difficult and becomes even more complex with the existence of speckle noise.

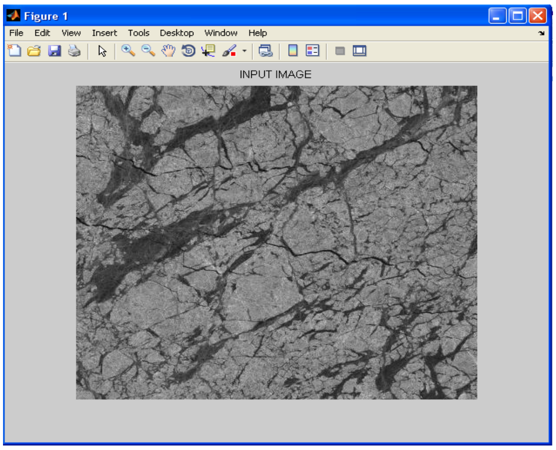


Figure3. SAR Sea image given as input

3.2.2 Initial Segmentation (Watershed Segmentation)

The input image is taken and is given for initial segmentation. At first an image is being read and watershed segmentation is performed on that image. This generally eliminates the plateaus defined as the uniform regions, by converting the image to a floating point representation. Here watershed segmentation is done as initial segmentation. The watershed algorithm from mathematical morphology is powerful for segmentation. However, it does not allow incorporation of a priori information as segmentation methods that are based on energy minimization. In particular, there is no control of the smoothness of the segmentation result. The output image of the initial segmentation is given in the Figure 4.

ISSN 1512-1232

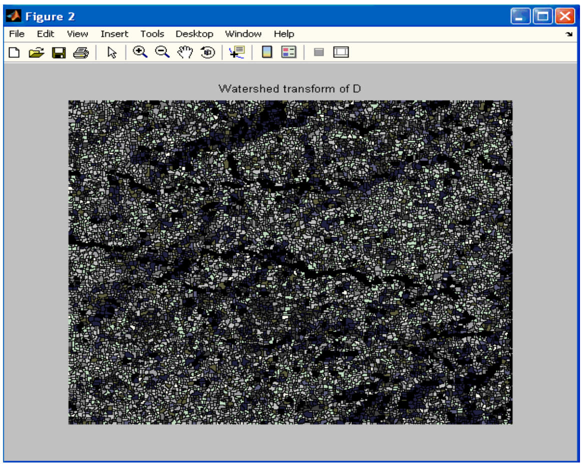


Figure 4. Image obtained after initial segmentation

3.2.3 Graduated Increased edge penalty

This section focuses on one aspect of the proposed IRGS method: using a sequence of edge penalty functions to approximate the MRF spatial context model. This aspect alone provides a basis for a novel segmentation method and it will be referred to as the graduated increase edge penalty (GIEP). The GIEP has two attractive features. First, it utilizes edge information to improve the segmentation on non stationary situations. Second, it provides a simple and elegant method of simultaneously estimating model parameters and searching solutions for the MRF-based formulation. Since the last summation term is nonzero only at boundary sites (that is, having at least one neighbor belonging to a different region), the corresponding MRF spatial context model functions as a penalty for the existence of boundary site pairs. Instead of penalizing equally for all boundaries site pairs, a greater penalty can be applied to weak edge and a lesser penalty to strong edge so that local statistic, such as edge strength, can be incorporated. Therefore, the penalty term can be replaced with some monotonically decreasing function g(.) of the strength of the edge between the two neighboring sites astride the boundary. The Output of IRGS is shown in Figure 5.

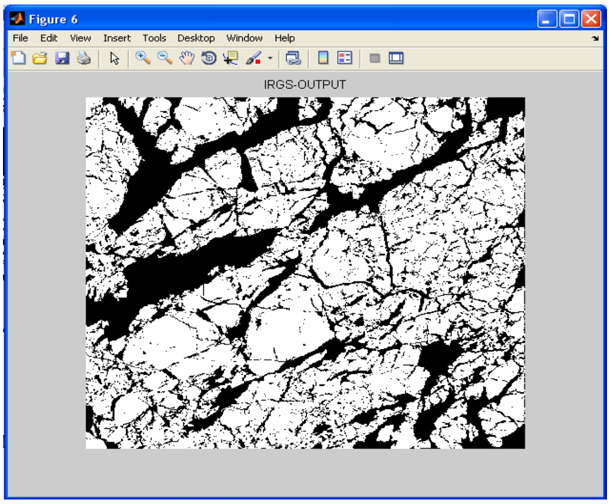


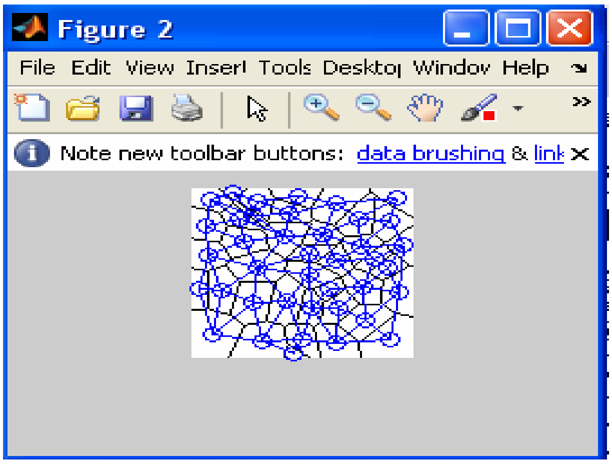
Figure 5. IRGS Output Screen

3.2.4 Region Adjacency Graph

Then Region Adjacency Graph is drawn for the segmented image. This method equivalently can be used as edge detection and result is satisfied to the topological requirement. This method has a hierarchical structure and is guaranteed to produce connected regions. This iteratively joins adjacent regions based on some predefined criterion. The image is mapped onto a graph one pixel

ISSN 1512-1232

corresponding to one vertex. Costs for every possible merge are then calculated, sorted and stored in a table. In a situation, where the image has 128 X128 pixels and the graph is 4- connected, there 2 X 128 X 127 initial connections, corresponding to the same number of possible merging. Merging is always conducted in such a way that the distortion or cost caused by merging is minimized. It is therefore a stepwise optimal data driven approach, and there is no further restriction imposed on the shape of the final segmentation result. Moreover, it can also offer the possibility of embedding flexible merging criteria in context with application. The output of this step is shown is Figure 6.



Figur 6. Region Adjacency Graph

3.2.4.1 The choice of Cost function:

The choice of cost function depends on the desired purpose of segmentation. A common cost measure is sse (sum square error): where g(x,y) is the approximation function and h(x,y) is the original data.

$$sse = \sum_{x,y \in region} (g(x,y) - h(x,y))^2 \tag{2}$$

$$g(x,y) = \frac{1}{n} \sum_{x,y \in region} h(x,y) \tag{3}$$

The algorithm will produce the maximum SNR with respect to the minimum number of regions also the produced boundaries are all edges due to the inherent property of penalty calculation. In general, the criterion is the maximum benefit return. Specifically the criterion for merging is as follows

$$C_{Total} = C_{1+2} - C_1 - C_2 \tag{4}$$

Where C_{1+2} is the cost of coding the merged region, C_1 & C_2 are the costs of coding regions 1 and 2 respectively.

3.2.5 Region Splitting Steps

In order to detect candidate regions or groups for splitting, the following test can be utilized:

Step 1: At the beginning of the model-based region grouping stage, check each region. If that region's model fitting cost is larger than a threshold, then it is a splitting candidate.

Step 2: At the end of merging, for each group, if its model fitting cost is larger than the threshold, then it is a splitting candidate.

The split threshold can be obtained through statistical analysis for a set of training examples. Since the number of regions in the over-segmentation result is large typically, we only utilize the splitting operation after the model-based merging stage. If new regions occur in the model based splitting stage, then the model-based merging stage is invoked again to refine the final result.

ISSN 1512-1232

If a region is a splitting candidate, then we must determine a list of the possible cut end points (places to cut the region boundary). Ideally, the method used in this step should be insensitive to small changes of shape due to differences among the members of an object class, or due to small differences in view position. Because curvature computation is more complex and sensitive to noise, we do not utilize the minima rule for determining possible split boundaries instead, we will search for cut end points in concave segments on the region boundary. Cut end points are generally located in the region boundary's concave segments, which are most likely inside the fitted model.

Making use of the difference between the region boundary and model boundary can make the splitting algorithm more robust and more efficient since knowledge about the object can be employed. After we have determined the overlap between the fitted model and the underlying region, we can find the curve segments of the region boundary inside the model area. The search for candidate cut end points will be limited to these segments. Two cut point search strategies have been tested in our experiments: the concave segments detection method, and the distance from the model boundary method.

It is difficult to say which of the region growing algorithms worked better in general. The two algorithms each had difficulties dealing with various image features. The max-min algorithm did a better job of preserving edges and handled some textures better than the mean algorithm. The mean algorithm did better on images with speckle. In the end the choice of the algorithm really depends on the image you are dealing with. The final output of the region splitting technique is shown in Figure 7.

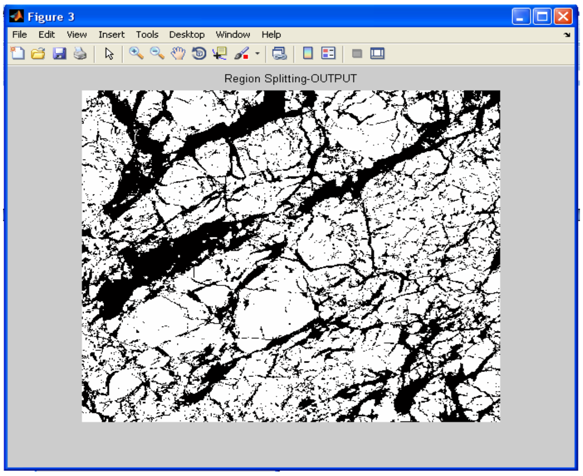


Figure 7. Region Splitting output screen

4. Performance Analysis

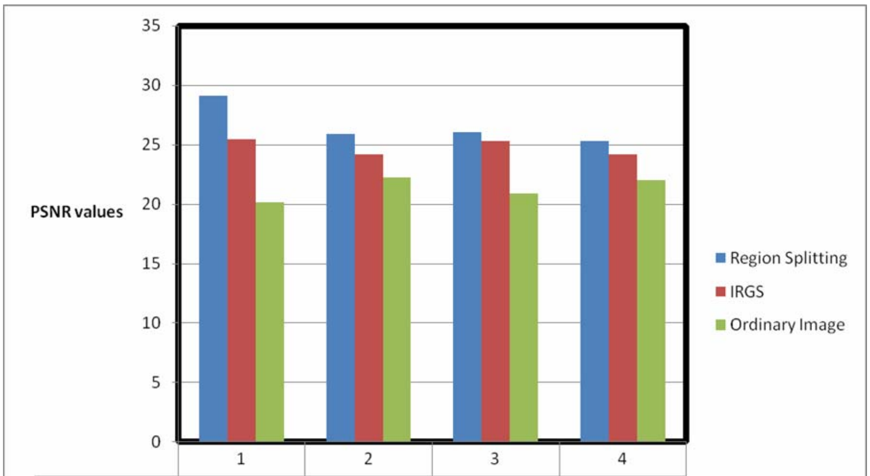
The performance analysis is done based on the comparison of the PSNR values. When the value of the PSNR increases, the higher the quality of the image can be obtained. The following table provides the comparison of the various PSNR values

Region Splitting	Ordinary Image	IRGS
29.1297	25.2138	25.4634
25.8747	25.2203	23.2150
26.0254	25.3240	24.9324
25.3061	25.2203	22.0321

Table 1. PSNR values

ISSN 1512-1232

The PSNR values of various images are analyzed and a graph is drawn by comparing the various values, such as the Region Splitting method (proposed method), IRGS (existing method), Ordinary gray scale image.
The graph is drawn as follows in Figure 8 with the various PSNR values.



	1	2	3	4
Region Splitting	29.1297	25.8747	26.0254	25.3061
IRGS	25.4634	24.2203	25.324	24.2203
Ordinary Image	20.1238	22.215	20.9343	22.0321

Figure 8. PSNR Graph

5. Conclusion and Future Scope

The merge-split algorithm due to its use of criteria based on the difference between the maximum and minimum pixel values within the region tends to act like an edge detection algorithm. In smooth (no noise or textures) and low gradient images, edges are the only areas where large differences in pixel values tend to occur. As a result near edges, the merge-split algorithm tends to split blocks down to individual pixels. Larger merged blocks appear in the interiors. So for this class of images, merge-splitting is an effective first stage in segmentation, and region growing can take place faster. For images with complex sub regions, fine detail, patterns, and gradients such as the plane, merge-splitting with a max-min criteria doesn't buy you that much. Too low a merge split threshold creates too many small pixel size regions. Too high a merge split threshold creates too many large blocky regions. Using merge splitting prior to region growing tends to result in sharper edges. On the other hand, the region growing techniques without merge-splitting will be generated images with blurry edges.

As a future work combination of the Region splitting along with Semantic Texton Forests can be done to yield better results. Because of its Efficient and powerful new level features such as (i) Acts directly on Image pixels, (ii)Extremely fast to train and test. By which faster segmentation can be done yielding more accurate results.

ISSN 1512-1232

References

1. Q. Lu and T. Jiang, “Pixon-based image denoising with Markov random fields”, *Pattern Recognit.*, vol. 34, 2001, pp. 2029-2039.
2. A. Sarkar, M. K. Biswas, and K. M. S. Sharma., “A simple unsupervised MRF model based image segmentation approach”, *IEEE Trans. Image Processing,* vol. 9, 2000, pp. 801-812.
3. Qiyao. Yu and David.A. Clausi, “IRGS:Image Segmentation Using Edge Penalties and Region Growing,” *IEEE Transactions on Pattern Analysis And Machine Geoscience and Intelligence*, Vol. 30, No. 12, December 2008.
4. C. Bouman and M. Shapiro, “A Multiscale Random Field Model for Bayesian Image Segmentation,” *IEEE Trans. Image Processing*, vol. 3, no. 2, 1994.
5. Z. Kato, M. Berthod, and J. Zerubia, “A Hierarchical Markov Random Field Model and Multitemperature Annealing for Parallel Image Classification,” *Graphic Models and Image Processing*, vol. 58, no. 1, pp. 18-37, 1996.
6. J. Laferte, P. Perez, and F. Heitz, “Discrete Markov Image Modeling and Inference on the Quadtree,” *IEEE Trans. Image Processing,* vol. 9, no. 3, 2000.
7. R. Wilson and C. Li, “A Class of Discrete Multiresolution Random Fields and Its Application to Image Segmentation,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 1, pp. 42-56, Jan. 2002.
8. A. Barbu and S.C. Zhu, “Generalizing Swendsen-Wang toSampling Arbitrary Posterior Probabilities,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1239-1253, Aug. 2005.
9. Y. Boykov, O. Veksler, and R. Zabih, “Fast Approximate Energy Minimization via Graph Cuts,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222-1239, Nov. 2001.
10. S. Wesolkowski and P. Fieguth, “Hierarchical Regions for Image Segmentation,” *Proc. Int’l Conf. Image Analysis and Recognition*, 2004.
11. Lifeng Liu and Stan Sclaroff, “Deformable model-guided region split and merge of image regions” *Image and Vision Computing* 22 (2004) 343–354.

Article received: 2010-06-10