BTP REPORT

Image segmentation by region growing

by
Utkarsh Kumar Shah
B.Tech, 4th year
Y6510
Guide: Prof. A. Mukerjee
Dept of Computer Science and Engineering
IIT Kanpur - 208 016
INDIA

April 15, 2010

1 ABSTRACT

This project presents an approach of segmenting an image by dividing it into multiple regions or fragments. Region growing is based on selecting initial seed points and adding neighboring pixels to the region depending on the suitable membership criteria such as color similarity. Initially seed regions are identified based on calculation of variance of R, G and B components and selecting the first seed point having the minimum R, G, B variance product. 4 connected neighbors are used to grow from the seed point. This process runs iteratively until no change is found in two successive stages . Experimental results on a number of images demonstrate the effectiveness of the technique. A comparison of this approach with mean shift segmentation is also presented.

2 IMAGE SEGMENTATION

Image segmentation is an important aspect of digital image processing. It basically aims at dividing an image into subparts based on certain feature. Features could be based on certain boundaries, contour, color, intensity or texture pattern, geometric shape or any other pattern. It provides an easier way to analyze and represent an image. In all segmentation is a process of assigning a label to pixels pertaining similar characteristics.

2.1 Key Questions

- What determines a segment?
- How can we pose the problem mathematically?

The answer to first question lies in perception of a segment. In real world suppose we consider clouds scattered in the sky as different segments but in digital world they would be often considered as a single segment due to similar in color and pattern of the clouds as well as sky. Hence, there can be difference in real world and digital world segment based on semantic and psychological understanding. An image segment can be based on color feature, texture feature or intensity of pixels etc. In order to segment an image we need to perform some computation on the image. So, for this purpose we need some mathematical representation of pixels and then develop an algorithm to compute the segments.

Image segmentation is mainly used to locate objects or object boundary, lines etc in an image so it can be used in applications which involve a particular kind of object recognition such as:

- Face Recognition
- Fingerprint Recognition
- Medical imaging (locating tumors and other deformities)
- Locating objects in satellite images

Since it deals with object detection, so it can also be used in applications involving object tracking such as traffic surveillance etc.

2.2 Objective

There are various existing algorithms for image segmentation. Among which the widely used one is k means clustering algorithm. The other techniques used so far includes histogram based methods, edge detection, graph partitioning method, watershed transformation, level set methods, model based segmentation, multi scale segmentation, neural network segmentation and lot more.

As a review on image segmentation methods in [5] suggests some of these methods use only gray level histogram. Most of these methods are not suitable for noisy environments and those which are robust to noise are computationally expensive. The k means clustering algorithm has disadvantage that it takes number of cluster as an input parameter whose inappropriate choice may yield poor results.

It is difficult to identify significant valleys and peaks in the image for histogram based methods. Model based segmentation assumes that region of interest has a repetitive form of geometry. Hence we can see many of these algorithm result in over segmentation or poor segmentation due to one or the other reason.

Given below is an example of segmentation of the famous lena image by multi scale segmentation algorithm in which segmentations are computed at multiple scales and by the region growing method.

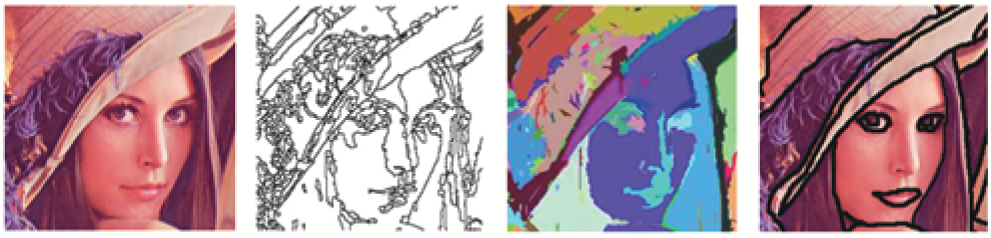

Figure 1: An example of image segmentation: (I) Original image, (II) Segmentation result by multi scale segmentation method showing over segmentation, (III) Segmentation result by region growing method described in this project, (IV) Manually marked proper segments in the image

The quality of the segmentation depends a lot on image. Smooth surfaces are usually ideal for segmentation. Many segmentation algorithms work well with simple images but they give bad results on complex images having clutter and camouflage.

Level set method in combination with region growing method has the advantage that it works by evolving the contour of the seed region based on the likelihood image obtained by a cost function involving a level parameter and the mahalanobis distance of the image feature vector to the appearance of the region modeled as a single gaussian using the mean and the covariance of the features in the region.

We have used region growing method with level set formulation in this project. It takes a seed as input and the pixels are further allocated to corresponding seed region based on comparison with unallocated neighboring pixels on criteria such as difference between pixel's intensity and the mean intensity of the region. Typically, a region growing algorithm starts with a seed point or seed area and then progressively evaluates and adds or discards neighbors to the region based on their similarity to the region until a stopping criterion is met.

A level set is defined as a set $S_L = \{ \text{x} \mid \text{f(x)} > \text{l} \}$ such as points corresponding to f(x) value greater than the predefined level l are included in the level set. Based on the value of l, a considerable number of points can be included in the set. In two dimension, it is more like a closed contour formed by including points of level set and whose covered area can be increased or decreased based on the value of l.

3 SEGMENTATION ALGORITHM USED

3.1 Approach

Here we have used a simple region growing approach [2, 6] to fragment the object and scene in a computationally efficient manner. For input we primarily consider image RGB color here, although similar techniques can be used with texture information. The main motivation to use this algorithm is that it segments the image based on similarity of neighboring pixels by evolving contour of seed region using likelyhood image generated by level set formulation. Also this algorithm is found to be much faster than several other algorithms. A comparison of the this algorithm's result given in [2] and mean shift [3] algorithm's result given in [2] is given in figure 2.

I II III

Figure 2: Segmentation result of region growing algorithm compared with other methods. I. Original Image, II. Region Growing based on algorithm in [2], III. Mean Shift based on algorithm in [3]. Mean shift approach takes roughly 30 seconds to segment an image of size 320X240. Region growing approach completes the segmentation process in less than a second as given in [2]

3.2 Region Segmentation

3.2.1 Seed point:

The seed points for growing the regions are identified by computing a score for every pixel:

n(x) = varR*varG*varB

where varR, varG and varB are variance of the R, G and B component of pixel respectively taken over a window of 20 pixels taking x as center (boundary pixels are taken care accordingly).

Each pixel i is stored as a set, V = (x_i, y_i, n_i) where x_i and y_i denotes the respective co-ordinates.

Now, a set of all the pixels is stored in S, S = (V1, V2,..., Vn).

3.2.2 The region segmentation iteration

The minimum element in S signifies that the region around this pixel is more homogeneous and can serve as a good seed point for the region growing algorithm. Following steps are taken then:

- Pick the minimum element in S. Create a region to hold the pixel. The seed area Ω is initialized using the window Rj centered at the this current minimum.
- The state information of the region growing algorithm, ϕ and L, are then initialized as given in equations mentioned later.
- Compute Mean μ_j and Covariance Σ_j of the region Rj.
- Compute the likelihood which is defined as:

  ψ_j(x) = MD (f(x), N(μ_j, Σ_j)) - T

  where value of T can be defined manually in order to get best segmentation results and MD(f(x), N(μ_j, Σ_j)) represents the Mahalanobis Distance of the image feature vector f(x) = [fr(x), fg(x), fb(x)] to the appearance of the region modeled as a single gaussian using the mean μ_j and the covariance Σ_j of the features in the region.

  The function ψ_j(x) is the level set formulation with level defined by the value of T.
- Grow the region using region growing algorithm as follows:

  Update μ_j, and Σ_j , as a new pixel is added

  Remove the corresponding element in S if a pixel is added
- Continue above steps if S is not empty.

3.2.3 The details of likelihood term computation

I have used two methods to compute likelihood term ψ_j(x).

The first one is using mean and variance of the R,G and B component of the pixel individually.

- Compute the μ_R, μ_G and μ_B of the image.
- Compute the var_R, var_G and var_B of the image.
- ψ_j(x) is calcualted as follows:
  ψ_R(x) = sqrt((x_R - μ_R)*(x_R - μ_R)/var_R) - T
  ψ_G(x) = sqrt((x_G - μ_G)*(x_G - μ_G)/var_G) - T
  ψ_B(x) = sqrt((x_B - μ_B)*(x_B - μ_B)/var_B) - T

The image obtained by this ψ_j(x) is given in Figure 3

Figure 3: The likelihood image obtained by mean and variance. The threshold T is taken as 1.0. The R component value is set to be 255 in output image to the pixels with ψ_R component greater than 0 in input image, the G component value is set to be 255 in output image to the pixels with ψ_G component greater than 0 in input image and the B component value is set to be 255 in output image to the pixels with ψ_B component greater than 0 in input image.

But, since we needed a single ψ(x) value for each pixel and this gives R, G and B components, so a second method is used to calculate ψ(x) using mean and covariance matrix.

- Compute μ and Σ of the image
- ψ(x) is calculated as follows:
  ψ(x) = sqrt($[x-\mu]^T$ S^{-1} [x - μ]) - T
  where S is the 3X3 covariance matrix.

The image obtained by this ψ(x) is given in Figure 4 with different values of T.

Iteration of the region growing algorithm:

do {

- Pick a seed point that is not associated to any fragment
- Grow the fragment from the seed point based on the similarity of the pixel and its neighbor's appearance
- Stop growing the fragment if no more similar pixels are present in the neighborhood of the fragment

} until all pixels are assigned

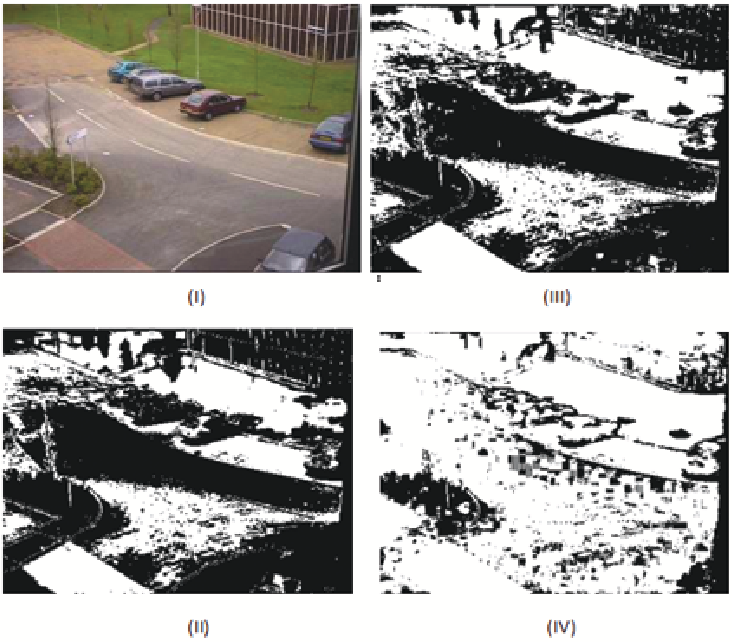

Figure 4: The likelihood image obtained by mean and covariance matrix. The threshold T is taken as 2000.0 for (II), 1500.0 for (III) and 1000.0 for (IV). The R, G and B component values are set to be 255 in output image for the pixels with ψ greater than 0 in input image.

3.3 Region Growing

Region growing is the process of expanding or contracting a region based on its properties and those of its neighborhood. It starts with a seed region,Ω,and uses two kinds of representation: an explicit representation using a singly linked list, L, and an implicit representation,ϕ, similar to the level sets approach. ϕ is initialized as follows:

ϕ(x)= { +1: x $\in \Omega$; -1: otherwise }

The singly linked list, L, is initialized as:

L = {x: x $\in \Omega$, $\exists$ x' $\in N_4$(x) such that x' !$\in \Omega$ }

L represents the boundary of the region being grown

Another representation ψ(x)is used to represent the likelihood term obtained from a confidence measure of each pixel belonging to the actual region R

The region growing procedure updates the frontier L for every iteration by considering only the neighbors of the current frontier. $\forall$ x $\in$ L:

$$
\begin{aligned}
L^{k+1} = L^k &\oplus \{ \text{x'} : \text{x'} \in N_4(\text{x}) \text{ and } \phi(\text{x'}) < 0,\ \psi(\text{x'}) > 0 \} \\
&\ominus \{ \text{x} : \forall \text{x'} \in N_4(\text{x}) \text{ such that } \psi(\text{x}) > 0,\ \psi(\text{x'}) > 0 \} \\
&\oplus \{ \text{x'} : \text{x'} \in N_4(\text{x}) \text{ and } \phi(\text{x'}) > 0,\ \psi(\text{x'}) < 0 \} \\
&\ominus \{ \text{x} : \forall \text{x'} \in N_4(\text{x}) \text{ such that } \psi(\text{x}) < 0,\ \psi(\text{x'}) < 0 \}
\end{aligned}
$$

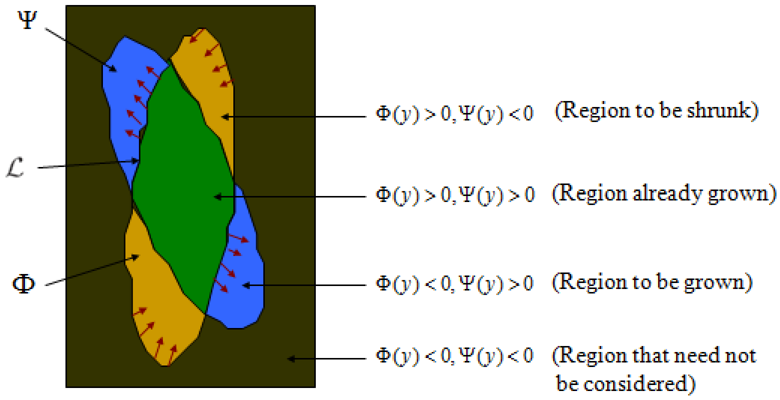

Figure 5: Demonstration of the region growing algorithm where the frontier L evolves from its current position using the implicit representation ϕ and the likelihood of the region ψ.Four region types are possible using ϕ and ψ which are indicated by different colors. This figure is taken from [2]

The $\oplus$ and $\ominus$ operator define the addition and removal of elements from the set. The first term deals with the expansion of the region. A neighboring pixel of the frontier that is not part of the current region is added to the frontier if it has a positive likelihood value. During such an expansion step, some frontier pixels may become interior and need to be removed. The second term removes such interior pixels. The third term correspond to the contraction of the region. A neighboring pixel of the frontier that is part of the current region with a negative likelihood is added to the frontier to contract the region. During this step, existing frontier pixels may become exterior and the fourth term removes such exterior pixels from the frontier.

Since ϕ is an implicit representation of the region, there is no necessity of removing any interior or exterior pixels. For every iteration,ϕ is updated as: $\forall$ x $\in$ L

$$\phi^{k+1}(\text{x'}) = \{ +1, \text{x'} \in N_4(\text{x}) \text{ and } \phi^k(\text{x'}) < 0, \psi^k(\text{x'}) > 0 \} \text{ (expansion)}$$
$$\{ -1, \text{x'} \in N_4(\text{x}) \text{ and } \phi^k(\text{x'}) > 0, \psi^k(\text{x'}) < 0 \} \text{ (contraction)}$$

The region evolution is stopped when no more points are removed or added to the frontier.

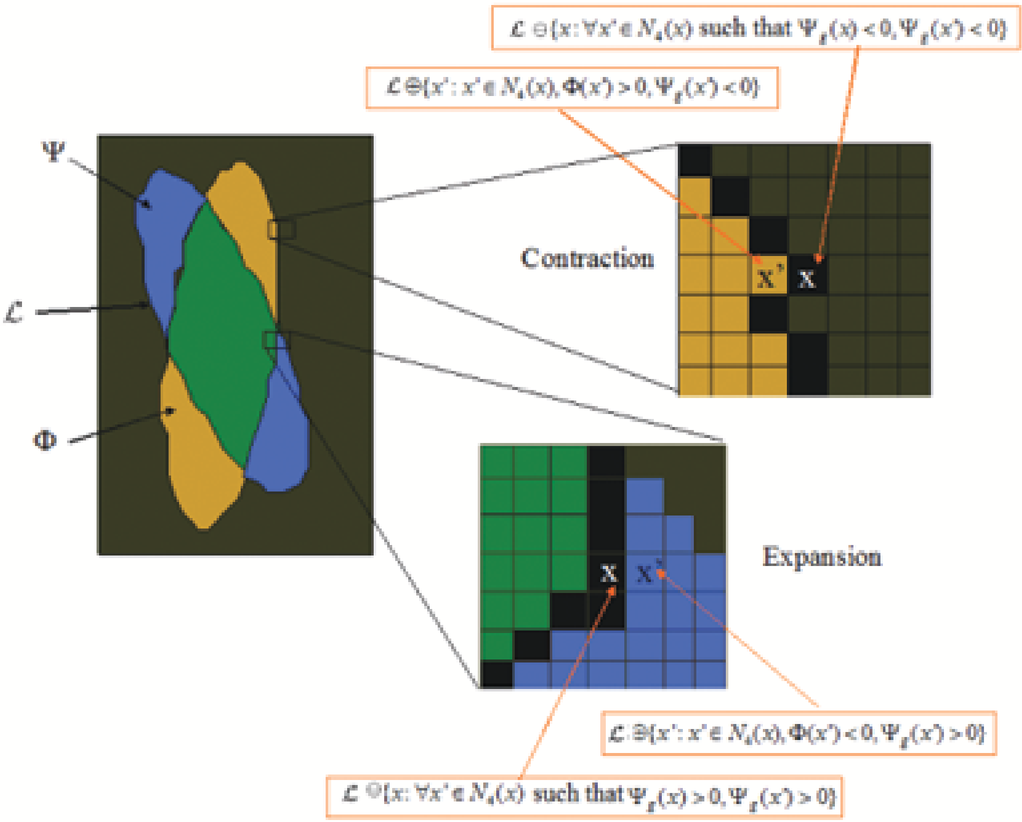

Figure 6: Cases of expansion and contraction are shown for the pixels where one of the four conditions in equation are satisfied. This figure is taken from [2]

Region growing algorithm

Require: The likelihood term ψ, ϕ and the list L

ϕ represents the entire region and L represents the boundary of the region

repeat
  for all $x \in \mathcal{L}$ **do**
    { Expansion Step: }
    for all $x' \in N_4(x)$ **do**
      if $\Phi(x') < 0$ AND $\Psi_g(x') > 0$ **then**
        $\mathcal{L} \leftarrow \mathcal{L} \oplus x'$
        $\Phi(x') \leftarrow +1$
      end if
    end for

Expand

    if $\Psi_g(x) > 0$ AND $(\Psi_g(x') > 0, \forall x' : x' \in N_4(x))$ **then**
      $\mathcal{L} \leftarrow \mathcal{L} \ominus x$
    end if

Remove interior points

    { Contraction Step: }
    for all $x' \in N_4(x)$ **do**
      if $\Phi(x') > 0$ AND $\Psi_g(x') < 0$ **then**
        $\mathcal{L} \leftarrow \mathcal{L} \oplus x'$
        $\Phi(x') \leftarrow -1$
      end if
    end for

Contract

    if $\Psi_g(x) < 0$ AND $(\Psi_g(x') < 0, \forall x' : x' \in N_4(x))$ **then**
      $\mathcal{L} \leftarrow \mathcal{L} \ominus x$
    end if

Remove exterior points

  end for
until $\mathcal{L}$ is not modified

Figure 7: The Region Growing Algorithm, Image source is [2]

3.4 The final segmented images:

Figure 8: The final segmented images. (I) Original image, (II) Segmented image, (III) Manual marking of segments on original image. (Original images taken from internet)

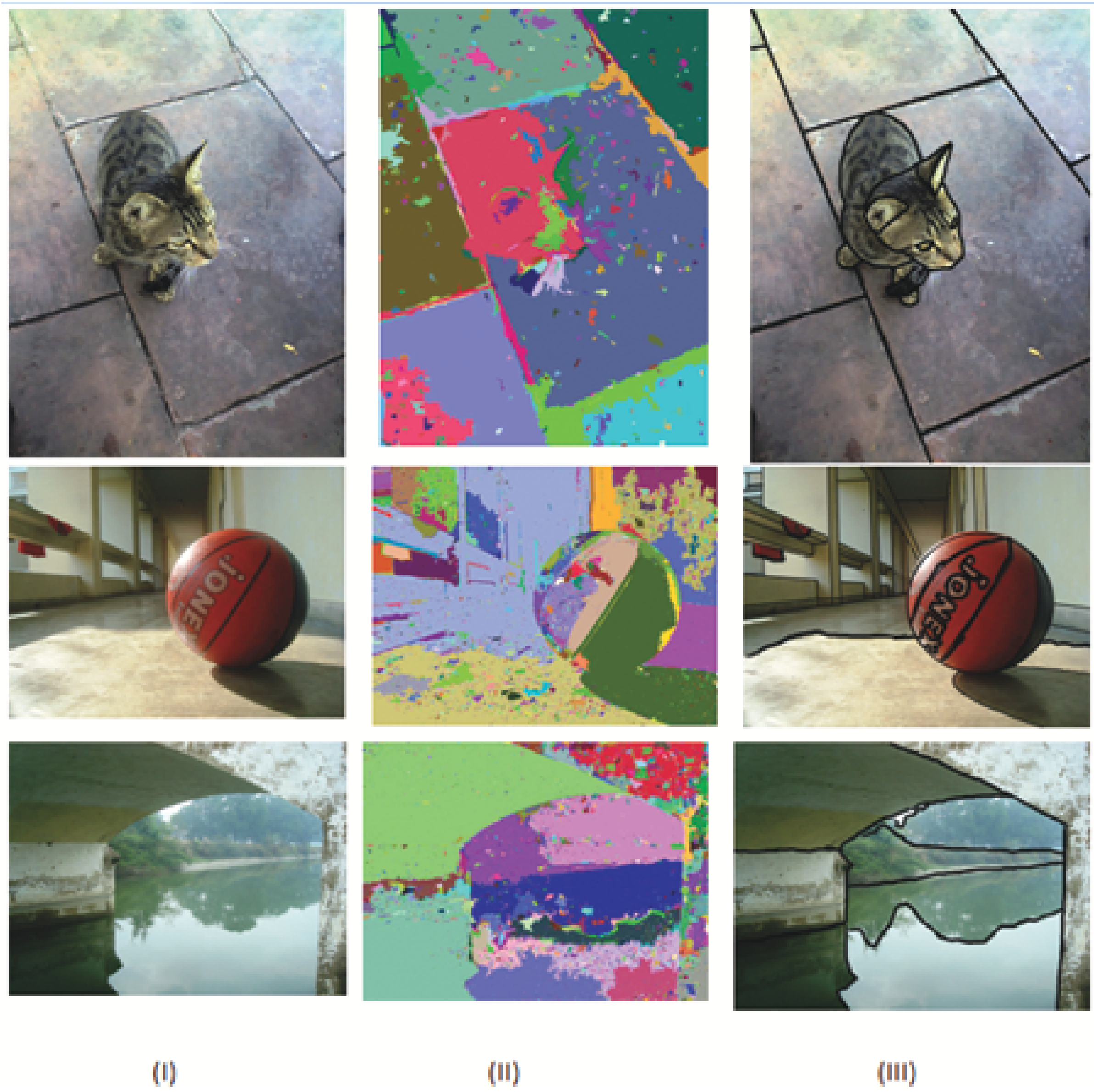

Figure 9: The final segmented images. (I) Original image, (II) Segmented image, (III) Manual marking of segments on original image. (Original images clicked by me)

4 Conclusion

Region growing is one of the classic approaches for solving computer vision problems like segmentation, region matching, and stereo matching. Some algorithms, like [1] and [8], use a best first strategy based on the cost where precedence is given to one neighbor over the other while growing the region. Using such a strategy will assist in speeding up the algorithm only in greedy cases where a fixed number of neighbors need to be accepted into the region. In most of the region growing algorithms, all neighbors need to be evaluated for the region to be grown. Hence, we prefer to store the costs and their associated pixels using an unordered linked list rather than a heap or some ordered sequence thereby reducing the running time.

Also the expansion and contraction is achieved using only a singly linked list instead of two separate frontiers. This also reduces some running time.

The major drawback of the algorithm is that it segments the image based only on color/intensity, hence it gives poor results for textured image. Also, it depends largely on the value of T in likelihood term computation which needs to be taken different value for different images.

In all, the segmentation algorithm is fast but it was found to be slightly sensitive to pre-defined thresholds.

Some of these measures can be incorporated to improve the result:

- Use texture features in place of color features in order to incorporate proper segmentation of textured images and shadow regions of an image.
- Add global information into the region segmentation process to reduce the sensitivity of the algorithm to pre-defined thresholds.
- Input images can be smoothen before segmentation to reduce noise.

References

[1] Stefano Soatto Andrea Vedaldi. Local features, all grown up. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 1753–1760, 2006.

[2] Prakash Chockalingam. *Non-rigid multi modal object tracking using Gaussian mixture model.* PhD thesis, Department of Computer Engineering, The Graduate School of Clemson University, 2009.

[3] D. Comaniciu and P. Meer. Meanshift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24', year =:603619.

[4] Maxime Lhuillier. Efficient dense matching for textured scenes using region growing. *British Machine Vision Conference*, pages 700–709, 1998.

[5] Shankar K Pal Nikhil R Pal. A review on image segmentation techniques. *Pattern Recognition*, 26:1277–1294, 1993.

[6] Nalin Pradeep Prakash Chockalingam. Adaptive fragments based tracking of non-rigid objects using level sets. *International Conference on Computer Vision (ICCV)*, 2009.

[7] W. C. Karl Y. Shi. Real-time tracking using levelsets. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2:34–41, 2005.

[8] Long Quan Yichen Wei. Region-based progressive stereo matching. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 106–113, 2004.