

# INTERACTIVE LIVER TUMOR SEGMENTATION USING GRAPH CUT AND GRAB CUT

Uma Maheswari R<sup>1</sup>, Vishnu Priya Dalai<sup>2</sup>

<sup>1</sup>Asst Professor, Department of Electronics & Communications Systems, Vignan's Institute Of Information Technology, AP, India, [maheswari.ramisetty@gmail.com](mailto:maheswari.ramisetty@gmail.com)

<sup>2</sup>Student, Department of Digital Electronics & Communications Systems, Vignan's Institute Of Information Technology, AP, India, [priyaatvizag@yahoo.co.in](mailto:priyaatvizag@yahoo.co.in)

## Abstract

This literature review attempts to provide a brief overview of the most common segmentation techniques, and a comparison between them. It discusses the "Grab-Cut" technique, and "Graph Cut" techniques. GrabCut is a way to perform 2D segmentation in an image that is very user friendly. The user only need to input a very rough segmentation between foreground and background. The Graph Cut approaches to segmentation can be extended to 3-D data and can be used for segmenting 3-D volumes. Other segmentation techniques use either contour or edge segmentation to perform segmentation. The Graph Cut techniques use both contour and edge detection. Typically this is done by drawing a rectangle around the object of interest. The way that this is accomplished technically is by using a combination of Graph Cuts and statistical models of the foreground and background structure in the colour space. Grab Cut Technique use very minimum energy to separate Foreground and Background Images.

**Keywords** - Interactive Image Segmentation, Object Selection, Foreground extraction, Graph Cut, Grab cut

\*\*\*

## 1. INTRODUCTION

This paper will start by describing general ways to categorize segmentation techniques, continues with an extensive survey of current contour-based interactive image segmentation techniques. Image segmentation is the problem of extracting (segmenting, cutting-out) foreground objects from background in an image. The work for the user goes as follows: The user selects an image which should be used to perform the segmentation. The user draws a rectangle around the object of interest that should be segmented. That's all input that is needed by the GrabCut algorithm to perform segmentation. Afterwards the user can do a touch up of the image if needed or use the image again by selecting a new rectangle introducing a new combined editing and segmentation tool. "Grab-Cut" uses a graph to represent an image, and then segments this graph by using a Min-Cut/Max-Flow algorithm. The inclusion of colour information in the Graph Cut algorithm and the iterative learning approach increases its robustness. Thus, GrabCut is a very promising image editing tool for foreground extraction.

As basic edge detection has a clause as it detects only on the edges of the ROI and the intensity of the result image is very low. In Cropping you can only specify the size and position of the crop rectangle as parameters when you call `imcrop`. Specify the crop rectangle as a four-element position vector. In Morphological Segmentation applications there are Thresholding, Black and White Labels and ROI. In Thresholding is a very simple form of segmentation. A

threshold is defined, and then every pixel in an image is compared with this threshold. If the pixel lies above the threshold it will be marked as foreground, and if it is below the threshold as background. The threshold will most often be intensity or colour value. Watershed usually suffers from over-segmentation due to noise or irrelevant local minima in the gradient image. Instead of training or predicting on voxels using classification method, watershed transform is employed as a pre-segmentation step to improve efficiency. GrabCut is a very user-friendly image segmentation tool. For images where the foreground and background are cleanly separated, GrabCut can robustly segment the image given only a rectangular region as input.

### 1.1 Medical Image Segmentation

The use of digital images in medicine has become very popular these days. All diagnostic imaging devices are computerized and can manipulate digital data. This includes Magnetic Resonance Imaging (MRI)[1], Computed Tomography (CT), Ultrasound imaging and more. Medical image processing is used as an important tool in computer-aided diagnosis for assisting doctors in evaluating medical imagery or in recognizing abnormal findings in a medical image. Structures of interest include organs or parts thereof, such as cardiac ventricles or kidneys, abnormalities such as tumors and cysts, as well as other structures such as liver structures, bones, vessels, brain structures etc. Magnetic Resonance Imaging (MRI) and Computed Tomography CT are vital diagnostic

imaging techniques. MRI scan uses magnetic fields to generate pictures of inside of the body as compared to the X-ray beams used in CT scans. Advantage of MRI is that the soft tissue contrast is much better than with X-rays leading to higher-quality images, especially in brain and spinal cord scans. Also, MRI technique [1] utilizes strong magnetic fields and non-ionizing radiation in the radio frequency range, and according to current medical knowledge, is harmless to patients. But, MRI scans are used only in a few situations, like diagnosing brain tumors and primary bone tumors. During a MRI/ CT scan, the patient may be advised to take a contrast (dye) so that resultant images of a specific part of the body have greater clarity. Medical image segmentation refers to the segmentation of known anatomic structures from medical images. Medical image segmentation is often application-specific. This has led to the development of a wide range of segmentation methods addressing specific problems in medical applications. The choice of segmentation algorithms vary between manual, semiautomatic or fully automatic methods. The segmentation process is usually based on gray level intensity, texture, color or shape.

This is the exact motivation for the interactive segmentation method presented here.

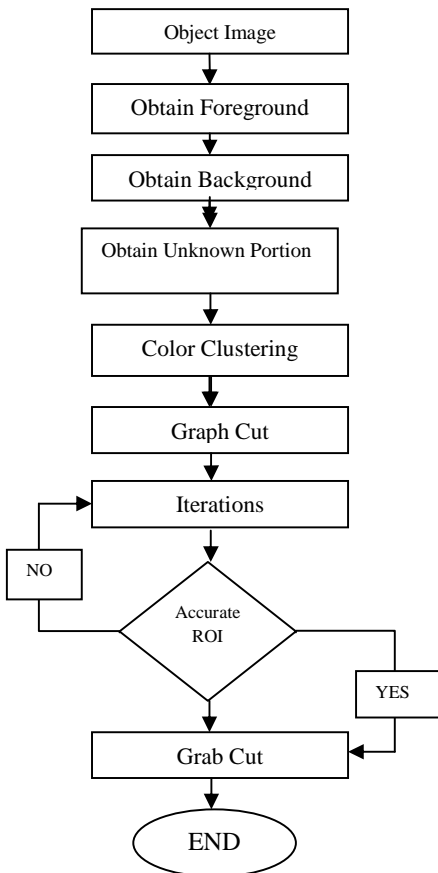


Fig.1. Operational flow chart for the proposed system

## 2. GRAPH CUT

Graph cut is a technique which is used to extract the required foreground and background from an image.

Graph Cut is a combinatorial optimization technique, which was applied by [Boykov and Jolly 2001] to the task of image segmentation. The image is treated as a graph - each pixel is a graph node. For the case of two labels (i.e. object and background) the globally optimal pixel labelling (with respect to defined cost function) can be efficiently computed by maxflow/min-cut algorithms. This technique can be applied to N-dimensional images [2]. Given user-specified object and background seed pixels, the rest of the pixels are labelled automatically.

Graph Cut methods treat the image segmentation problem as a min-cut problem on graphs. Originally suggested by Boykov and Jolly [3], the method is now widely used in many variants.

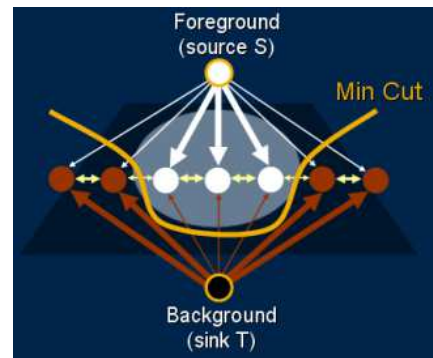


Figure 2: Foreground/Background pixel nodes are marked in white/brown (resp.). The source and sink terminal nodes are drawn outside the image. N-links are drawn in yellow and t-links are drawn in white/brown. Links with high weight are drawn thick. The minimal Cut best-separates the source, with all the foreground pixels, from the sink, with all the background pixels.

As we see in the Fig.2 every pixels denote a node itself. There are two major nodes Background represents sink and Foreground represents source. The lines represent the weights of the edges of foreground and Background. The pixels are connected to the edges between them; similar pixels will always try to form a group. This is called graph theory. Now we partition the graph with a min-cut i.e; a cut with minimal effort. It cuts from Foreground node with weak edge weights of pixels and it tries to go weak edges between Background and Foreground with minimal energy. In this way it separates the Foreground from the Background.

Boykov and Jolly (2001a) introduce a technique for segmentation using a graph to represent the image, and then using a Min-Cut/Max-Flow algorithm to segment the graph.

Pixels in the image are represented by nodes on the graph. The edge weights on the graph are defined by a cost function, which is defined by region and boundary information from the image. A Min-Cut/Max-Flow algorithm is then used to segment the image by minimizing the cost function. This technique has a well defined cost function and so provides a globally optimal solution.

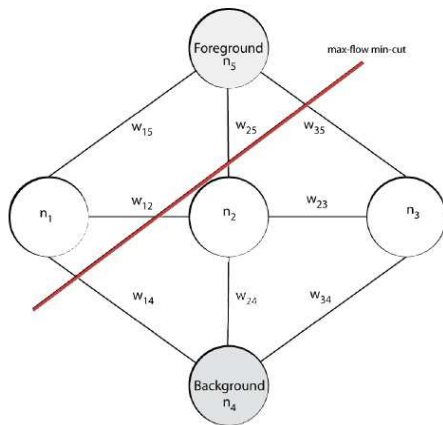


Fig.3 Example of graph to segment to foreground and background

The graph consists of nodes  $n_i$  that has links between them with weights,  $w_{ij}$ . The weights that describes how hard they are connected or related to each other. The min-cut between two nodes are a way to separate the graph in two distinct parts with minimal effort by minimizing  $\sum_{i \in I} w_i$  where  $I$  is the set of links between the nodes that were cut off. In the example in figure 3,  $n_1$  would be classified as foreground and all the other nodes as background.

### 3. GRAB CUT

The GrabCut algorithm introduced in [5] extends the graph cut technique proposed in [4] through an iterative energy minimization process. GrabCut still uses the MRF formulation of the segmentation problem and employs the min-cut/max-flow algorithm of [4]. The main difference lies in the user interface, which only requires a single bounding box around the foreground region. Because this reduces the quality of the initial pixel labels used to learn the unary potentials, they introduce an iterative method which still yields final segmentations comparable to the original graph cut results of [4]. Another important difference between the GrabCut algorithm and [4] is that it uses color information rather than grayscale intensity. Thus, the unary potentials must to model distributions within colour space. The iterative component of GrabCut is key because unlike the precise brushing in [4] the bounding box will incorrectly label some background pixels as foreground. To overcome this problem, [5] repeatedly updates the unary potentials and then the segmentation, at each

iteration, re-learning the appearance models from the current segmentation.

GrabCut [Rother et al. 2004] extends graph-cut by introducing iterative segmentation scheme that uses graph-cut for intermediate steps. The user draws rectangle around the object of interest - this gives the first approximation of the final object/ background labeling. Then, each iteration step gathers colour statistics according to current segmentation, re-weights the image graph and applies graph-cut to compute new refined segmentation. After the iterations stop the segmentation results can be refined by specifying additional seeds, similar to original graph-cut.

The basic steps for the GrabCut algorithm is as follows

1. The user input three things: The foreground, background, and the unknown part of the image that can be either foreground or background. This is normally done by selecting the rectangle around the object of interest and mark the region inside that rectangle as unknown. Pixel outside this rectangle will then be marked as known background.
2. The computer creates an initial image segmentation, where the unknown pixels are placed in the foreground class and all known background pixels are classified as background.
3. The foreground and background are modeled as Gaussian Mixture Models (GMMs) using the Orchard-Bauman clustering algorithm.
4. Every pixel in the foreground assigned most probable Gaussian component in the foreground GMMs. The same process is done with the pixels in the background but with components of the background GMMs
5. New GMMs are learned from the pixel sets that were created in the previous step.
6. A graph is built and Graph Cut is used to and a new classification of foreground and background pixels.
7. Repeat step 4-6 until the classification converges

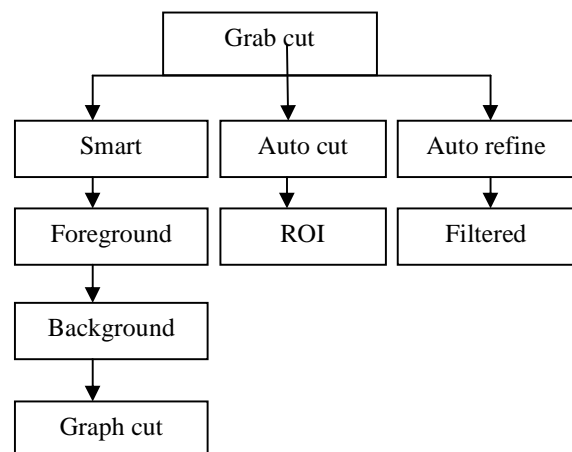


Fig 4: Flow chart for Grab cut.

### 3.1 How "GrabCut" Works

"GrabCut" is a segmentation technique that uses graph cuts to perform segmentation. Like most segmentation techniques "GrabCut" uses information encapsulated in the image. Most segmentation techniques make use of either edge information or region information in the image. "GrabCut" makes use of both edge and region information. This information is used to create an energy function which, when minimized, produces the best segmentation. In order to perform segmentation a graph is built, where nodes in the graph represent pixels in the image. In addition two special nodes are also created. These are the Sink and Source nodes. Every pixel node in the graph is connected to the Source and Sink node. The Source node represents the foreground of the image, and the Sink node the background. In order to segment the image the Source and Sink nodes must be separated.

The energy function is incorporated into the graph as weights between pixel nodes and weights between pixel and Source or Sink nodes. Weights between pixel nodes are determined by edge information in the image. Thus a strong indication of an edge between two pixels (a large difference in pixel colour) results in a very small weight between two pixel nodes. The region information determines the weights between pixels nodes and the Source and Sink nodes. These weights are calculated by determining the probability of the pixel node being part of the background or foreground region. In order for foreground and background regions to be created, some pixels in the image need to be labeled before segmentation as either foreground or background. This is referred to as the clue marking stage. Any pixels that are labeled during this stage are set as hard constraints. This means that during the segmentation process, hard labeled pixels cannot change their labeling.

A Min-cut/Max-Flow algorithm[5] is used to segment the graph. This algorithm determines the minimum cost cut that will separate the Source and Sink nodes. The cost of the cut is determined by the sum of all the weights of the links that are cut. Once the Source and Sink nodes are separated, all pixel nodes connected to the Source node become part of the foreground, and the rest become part of the background.

## 4. RESULTS

Grab Cut has three processes

1. Smart Selection
2. Auto Cut
3. Auto Refine.

In Smart Select there are the steps to get the required Segmented Image. Select foreground of the image, then select background of the image and finally apply Graph Cut to get the required Segmented Image. After multiple iterations in Graph Cut we will get the Final Output. But the final result was not appreciable.

So we can opt for Auto Cut Technique, here we need not give foreground and background as in Smart select its just enough to give ROI in a rectangular plane to get the required Segmented Image

Auto Cut Refine Technique is a system defined, Advanced processes of Auto Cut technique with an additional feature of Filtering of Image

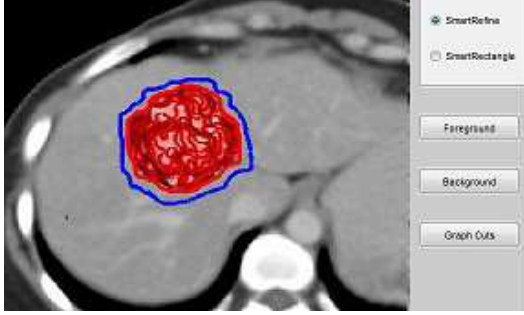

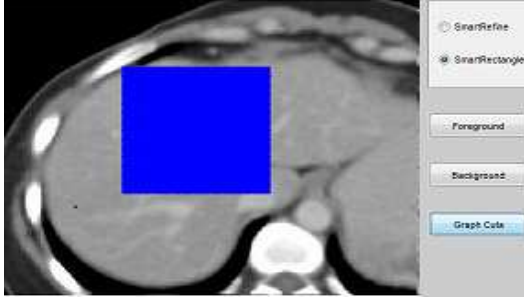

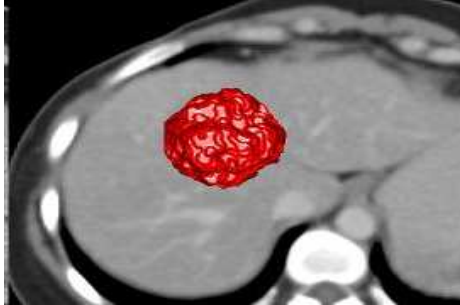

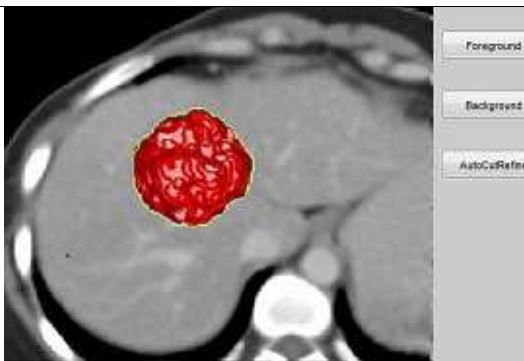
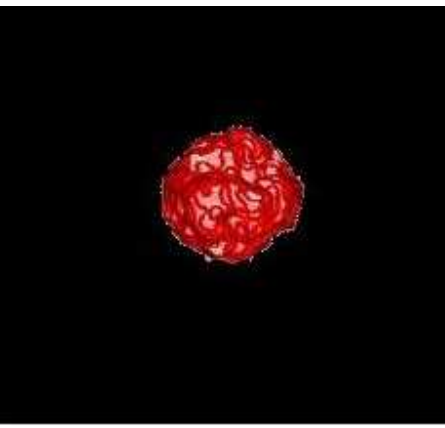
I have implemented the "GrabCut" segmentation technique as a plug-in for the GIMP, and here follow a few examples of the type of results that have been obtained:

**Smart Select:** - In this segmentation tool we can observe the foreground is selected with RED and background is selected with BLUE. After completion of Foreground and Background we select Graph Cut option on the window to Get the required Tumor as shown in Smart Select Technique.

**Smart Rectangle:** - In this segmentation tool we can observe the foreground is selected with RED rectangle and background is selected with BLUE rectangle over RED. After completion of Foreground and Background we select Graph Cut option on the window to Get the required Tumor as shown in Smart Rectangle Technique.

**Auto Cut:** - In this technique we use rectangle cropping where the tumor is detected quickly without many iterations.

**Auto Cut Refine:** - This technique is same as Auto Cut but we receive Filtered image of a detected Tumor. As we can observe GREEN line were generated automatically around the Tumor.

S.No	Technique	Original Image	Segmented Image
1.	Smart select		
2.	Smart rectangle		
3.	Auto-cut		
4.	Auto-refine		

## CONCLUSIONS

Our method exhibits promising results for the aimed application. However some open problems still remain. First, the segmentation of multiple tumors in the same liver often requires additional user markers to correctly separate the tumors. The developed method merges the tumors in a single object when different tumors are too close. This problem requires that the user adds markers between the merged tumors. However the used methods (minimal surfaces and Markov random fields on a region adjacency graph) are fast and can be used interactively. Secondly we did not develop any preprocessing step such as filtering of the images. There is thus still some possible improvements of our methodology. Grab Cut works well when the object of interest has an another colour distribution compared to the background. The result from the algorithm could be adjusted by a touch up by the user that may improve the result.



**Vishnu Priya D** M.Tech Student,  
Department of Digital Electronics and  
Communication Systems Vignan's  
Institute Of Information Technology, AP,  
India, priyaatvizag@yahoo.co.in

## REFERENCES

- [1] D. Adalsteinsson, and J.A. Sethian, A Fast Level Set Method for Propagating Interfaces , Journal of Comp. Physics, 118, pp.269-277, 1995.
- [2] Y. Boykov and G. Funka-Lea, Graph Cuts and Efficient N-D Image Segmentation, In International Journal of Computer Vision (IJCV), vol. 70, no. 2, pp. 109-131, 2006
- [3]Y. Boykov and M.P. Jolly, Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D images. In International Conference on Computer Vision, (ICCV), vol I, pp. 105-112, 2001
- [4]Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in ND images .In Computer Vision, 2001. ICCV 2001 Proceedings Eighth IEEE International Conference on, volume 1, pages 105–112.IEEE, 2001
- [5]C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactiven foreground extraction using iterated graph cuts. In ACM Transactions on Graphics (TOG), volume 23, pages 309–314.ACM, 2004

## BIOGRAPHIES



**Uma Maheswari R** M.Tech Asst  
Professor , Department of Electronics and  
Communications Systems, Vignan's  
Institute Of Information Technology, AP,  
India, maheswariu.ramisetty@gmail.com