

# A short report on 'Grabcut Automated'

## Introduction -

GrabCut algorithm was designed by Carsten Rother, Vladimir Kolmogorov & Andrew Blake from Microsoft Research Cambridge, UK. in their paper, "GrabCut": interactive foreground extraction using iterated graph cuts. An algorithm was needed for foreground extraction with minimal user interaction, and the result was GrabCut.

How it works from user point of view ? Initially user draws a rectangle around the foreground region (foreground region should be completely inside the rectangle). Then algorithm segments it iteratively to get the best result. This is done by learning a Gaussian Mixutre Model for the foreground and background pixels.

[http://docs.opencv.org/3.1.0/d8/d83/tutorial\\_py\\_grabcut.html](http://docs.opencv.org/3.1.0/d8/d83/tutorial_py_grabcut.html) gives a practical overview of Grabcut in OpenCV.

In this mini project, I set out to automate the whole process, so that there is no need to draw a bounding rectangle in order to specify the foreground region.

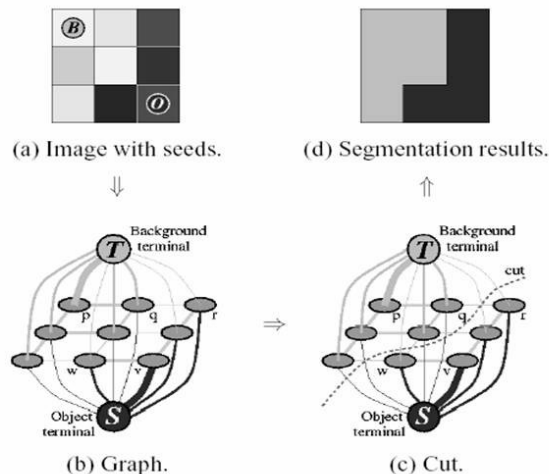


Fig 1 : Illustration of Grabcut.

## Problem Description -

Given an image of a lab scene, the task is to segment out the tables, floor and rest of the image.

## Approach -

Create images with textures of table and floor and rest of the content of image.

Learn a GMM of each label.

Build a graph from the learnt GMM. It is a 4 connected graph with each pixel as a node along with three other nodes for each label. Every foreground pixel is connected to Source node and every background pixel is connected to Sink node.

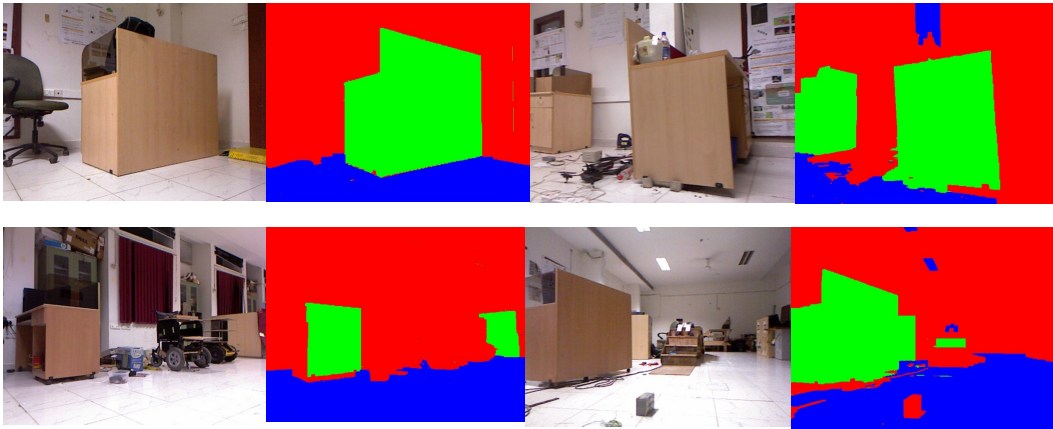
## Methodology -

I modified grabcut.cpp from OpenCV to learn the GMM model and output the node weights for each label to a .txt file. For performing alpha expansion algorithm for multi label inference, I used MRF 2.2 library. I input the node weights which act as the data term and use grabcut's formulation of the smoothness term and subsequently perform the

optimization.

## Results -

I present results in the form of predicted segmentation mask obtained using my method.



**Fig 1** : Pairs of images and predicted segmentation masks.

## References -

1. Fast Approximate Energy Minimization via Graph Cuts.  
Y. Boykov, O. Veksler, and R. Zabih.  
In *IEEE Transactions on Pattern Analysis and Machine Intelligence* (PAMI), vol. 23, no. 11, pages 1222-1239, November 2001.
2. Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. 2004. "GrabCut": Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.* 23, August 2004), 309-314.
3. [http://docs.opencv.org/3.1.0/d8/d83/tutorial\\_py\\_grabcut.html](http://docs.opencv.org/3.1.0/d8/d83/tutorial_py_grabcut.html)