

Computer Vision (CSE/ECE 578)  
Spring-2020  
Assignment-3 GrabCut and MRF  
Posted on: 4th March 2020  
Due on: 14th March 2020

---

## Instructions:

- The assignment can be implemented in Python or C++.
- Make sure that the assignment that you submit is your own work. Any breach of this rule could result in serious actions including an F grade in the course.
- This is a fairly large assignment. The experiments and report writing will take time. Start your work early and do not wait till the deadline.
- Write modular code, with comments clearly outlining the function of each module
- Demonstrate the results on the given images and additional images from your camera
- Make sure your files can be opened. Corrupted files will not be entertained.
- The code must be robust and scalable and it should work for images of any size.
- The submission must have a directory called code/ which contains all the code use to generate the results. The report and the output images should be in the report/ directory.
- Please ensure that the report contains your roll number. Zip the assignment folder. The zip file should be named Roll Number assignment3.zip
- Please post your doubts on the Moodle forum instead of asking the TA's personally.
- There are some sample images included in the assignment, these images are NOT the exhaustive test set. Your code may be tested on a different set.
- Add the procedure and the images of intermediate steps in the report. Add the code to the report as well and explain it wherever necessary.
- Make sure you provide the code separately also and not just in the report.

## Tasks:

### GrabCut

1. Implement GrabCut. See Rother et al. for a detailed description of the algorithm.
2. Refer to Sec.2 of the Rother paper and Boykov Jolly for details of energy minimization based image segmentation.
3. **What you need to implement:** You only need to implement the iterative procedure described in Sec.3.1 and Sec.3.2 of the Rother paper to get a binary segmentation of the image. The user interaction interface in Sec.3.3 is optional. You do not need to implement border matting or foreground estimation in Sec.4.
4. You can use GMM of standard libraries (sklearn etc) that would be required for implementation of Sec.3.1
5. We would be providing you with the test images and bounding boxes. Each image contains only one foreground object and has a corresponding bounding box, which is a rectangle around that object. Please find the images and bounding boxes here.
6. The final program must have a GUI component that allows the user to draw the bounding box which specifies the foreground object in an image and the program should generate the segmentation accordingly.
7. Your report must include a study of how changes in the various parameters affects the segmentation. Some parameters you could experiment with:
  - The number of iterations of GMM updating and energy minimization.
  - The number of mixture components in your GMM.
  - Whether to use soft GMM labels and do EM, and how that affects other parameters/results.
  - Different ways to represent probabilities other than GMMs. 4-neighborhood or 8-neighborhood in your pairwise term.
  - The choice of gamma.
  - Alternative ways of setting beta.
  - A tight initial bounding box or a loose bounding box.
  - Better ways of using the bounding box for segmentation.
  - Different color spaces or ways of representing pixels. Co-segmentation.

### Stereo correspondence using MRFs Bonus

1. Use the Markov Random Fields framework to generate the stereo matching for the pairs of images given in Assignment 2.

2. The report must include a comparison between this method and the methods you used in Assignment 2, with a qualitative explanation for your observation.
3. Please include the code used in a new sub-directory `mrf/` in the `code/` directory. The report for this may be included in the previous question's report.
4. Bonus marks will be given based on the code and quality of results.