CV2015Spring—Assignment #3

Due: June 18, 2015 (12:00AM)

Dai Jialun, Wang Ruchen

June 3, 2015

1. Assignment requirement

For this assignment, you will implement segmentation through grab cut and meanshift. Then, evaluate each method by adjusting parameters in segmentation. See Figure 1 and Figure 2 for examples.

(a) Input image (b) Segmentation result (c) Evaluation

Figure 1: Grab cut

(a) Input image (b) Segmentation result (c) Evaluation

Figure 2: Mean shift

2. Grab Cut (60 points)

The whole framework of the implementation for evaluation of grab cut is shown in Figure 3, it may serve as a reference for your assignment. In this part, what you need to do:

1. Implement image signature, grab cut and evaluation (20 points).
2. Adjust one parameter (see 2.6) to obtain different segmentation results, then evaluate (20 points).
3. Adjust another parameter (see 2.6) to obtain different segmentation results and evaluate (20 points).

Next, I will introduce the implementation and requirement of each part of the framework.

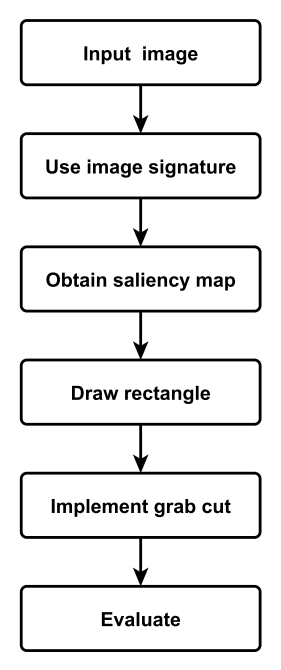


Figure 3: Framework of the implementation for evaluation of grab cut.

2.1 Step 1: Input image

A simple dataset (PASCAL) can be downloaded from the website[1]. The dataset contains input images and groundtruth. Use all images (850 images totally) from the dataset for segmentation, and evaluate grab cut by adjusting different parameters.

[1] http://vision.ouc.edu.cn/~zhenghaiyong/courses/cv/2015spring/assignments/PASCAL.zip

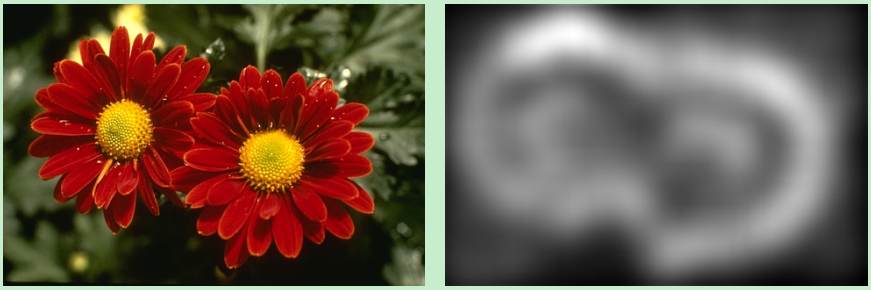
Here, I just pick one image for segmentation and evaluation as an example and the parameter I choose to adjust is threshold in step 3.

2.2 Step 2: Use image signature

Input The input color image ($m \times n \times 3$ matrix).

Output The saliency map ($m \times n$ matrix).

Hint The saliency map is obtained by image signature and the matlab code of "signatureSal" can be downloaded from website[2], where the files of SIG_single, signatureSal and default_signature_param are used.



(a) Input image (b) Saliency map

Figure 4: Image signature

2.3 Step 3: Draw the rectangle

Input The input saliency map image ($m \times n$ matrix).

Output A rectangle that is used to initialize the grab cut.

Hint

- The rectangle is used to locate the most probable position of object and initialize mask in grab cut.
- You can use thresholding to transform saliency map to binary image and draw a rectangle according to the binary image. The size of rectangle is also adjustable.

[2] http://www.vision.caltech.edu/~harel/share/gbvs.php

Figure 5: Rectangle in input image

2.4 Step 4: Implement grab cut

Input The input image ($m \times n \times 3$ matrix) from step 1, and the rectangle that you computed from step 3.

Output The image after segmentation ($m \times n \times 3$ matrix).

Implementation Use grab cut with the rectangle to initialize and iterate k times for segmentation.

Hint The C++ code for drawing rectangle and implementing grab cut can be downloaded from the website[3].

Figure 6: Segmentation result

2.5 Step 5: Evaluate segmentation result

Input Segmentation result ($m \times n \times 3$ matrix) and groundtruth ($m \times n$ matrix) from dataset.

Output A figure that indicates evaluation results (The horizontal axis represents the parameter, the vertical axis represents the evaluation results.).

[3] http://vision.ouc.edu.cn/~zhenghaiyong/courses/cv/2015spring/assignments/GrabcutCode.zip

Implement Adjust threshold to obtain different segmentation results and draw a bar graph to evaluate grab cut.

Hint

- You can draw PRF (Precision Recall F-measure) bar graph to evaluate.
- The evaluation is shown is Figure 7.
- A matlab code for computing PRF (Precision Recall F-measure) can be downloaded from website[4].

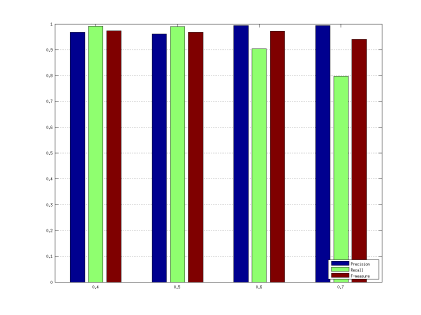


Figure 7: PRF bar graph

2.6 Notes

- You need to compute the mean value of evaluation result from all images when evaluating.
- The parameters you choose to adjust can be the followings (adjust at least two parameters for this assignment):
  - threshold in step 3.
  - size of rectangle in step 3.
  - times of iteration in step 5.

3. Mean shift (40 points)

The whole framework of the implementation for evaluation of mean shift is shown in Figure 8, it may serve as a reference for your assignment. In this part, what you need to do:

1. Input all images by batch processing (10 points).

[4] http://vision.ouc.edu.cn/~zhenghaiyong/courses/cv/2015spring/assignments/GrabcutCode.zip

2. Adjust two parameters of mean shift (see 3.2) to get different segmentation results (10 points).

3. Evaluate these segmentation results via two evaluation methods (see 3.3) (20 points).

Next, I will introduce the implementation and requirement of each step of the framework.

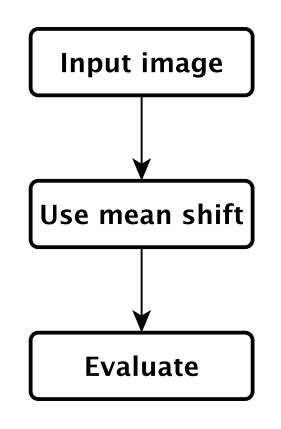


Figure 8: Framework of the implementation for evaluation of mean shift.

3.1 Step 1: Input image

A simple dataset (BSDS500) can be downloaded from the website[5]. The dataset contains input images and groundtruth. Use all images from the dataset for segmentation.

3.2 Step 2: Segment via mean shift

In this step, you need to adjust the parameters of mean shift to get different segmentation results.

Input The color image ($m \times n \times 3$ *matrix*).

Output The segmentation results ($m \times n \times 3$ *matrix*) and label matrixes ($m \times n$ *matrix*) of different parameters.

Implementation The parameters of mean shift, which can be adjusted, include:

- Spatial radius
- Color radius
- The number of iterations

[5] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/

- Iteration accuracy

You should adjust at least two parameters to get different segmentation results.

Hint You can download the source code from the website[6].

(a) (b)

Figure 9: The original image and segmentation result.

3.3 Step 3: Evaluate segmentation result with groundtruth

In this step, you need to evaluate the different segmentation results from step 2.

Input The label matrixes ($m \times n$ *matrix*) from step 2, the groundtruth matrixes ($m \times n$ *matrix*) from dataset.

Output Line chart of the evaluation results. (The horizontal axis represents parameters. The vertical axis represents the evaluation results.)

Implementation The evaluation methods include:

- Probabilistic Rand Index (PRI)
- Variation of Information (VOI)
- Global Consistency Error (GCE)
- Boundary Displacement Error (BDE)

You can choose at least two evaluation methods from them.

[6] http://www.mathworks.com/matlabcentral/fileexchange/40990-mean-shift-pixel-cluster/content/meanShiftPixCluster.m

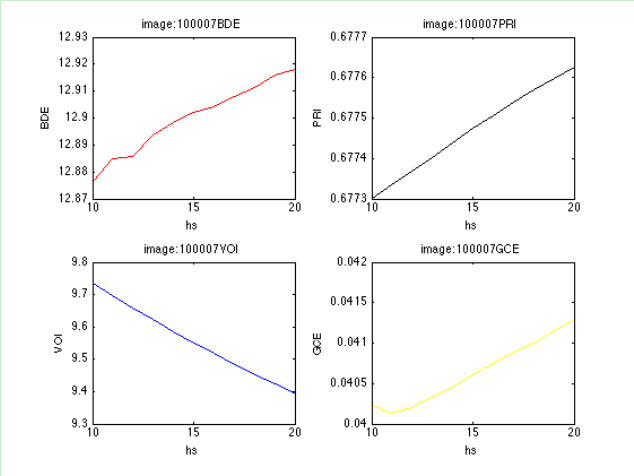


Figure 10: Line chart of BDE PRI VOI GCE

Hint You can download the source code from the website[7].

Example: Figure 10 shows four kinds of evaluation results (BDE PRI VOI GCE) by adjusting a parameter (spatial radius).

PS: The final evaluation result of the dataset is the mean of all images.

4. Submission instructions

4.1 What to hand in?

- Your code (Show your medial result in your code)
- A report containing the following:
  - Your name at the top
  - A brief explanation of your implementation strategy (in English)

4.2 Where to hand in?

Submit to Piazza in form of a followup below my assignment note.

[7] http://www.eecs.berkeley.edu/~yang/software/lossy_segmentation/SegmentationBenchmark.zip