CV2015Spring—Lab report of Saliency #1

Due: Thursday, April 12 8:00 AM

Zhao Haiwei

April 12, 2015

1. The arrangement of experiment

The whole framework of the implementation for salient object detection is shown in Figure 1.

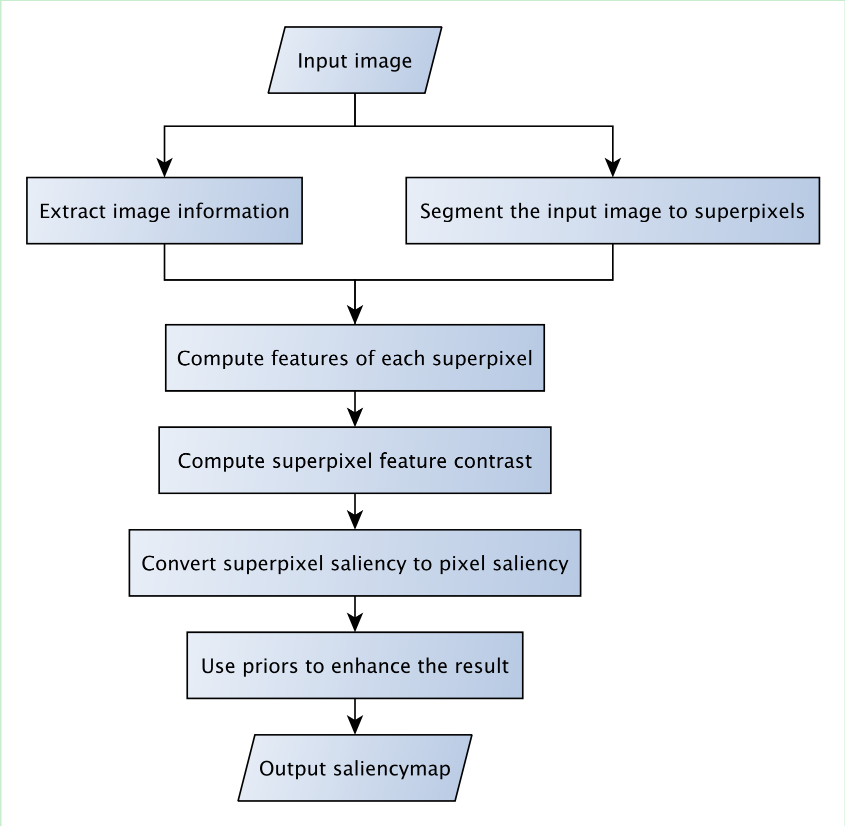

Figure 1: Framework of the implementation for salient object detection.

2. The arrangement of experiment

2.1 Input image

I pick one image from the dataset randomly for test.

Figure 2: Input image.

2.2 Step 1-1: Extract image information

Input The input color image ($m \times n \times 3$ matrix), m is the width of the input image, n is the height of the input image.

Output $m \times n$ matrix.

Implementation This step determines the regional feature I want to use in step 2, and I choose Color feature for this assignment.I quantize each color channel (RGB) to reduce the number of colors (from 256 to 32), then the number of color is reduced to $32 \times 32 \times 32 = 32768$. In order to create a bin for each color, I use a number ($1 \sim 32768$) instead of (R, G, B) values to represent a color uniquely.

Figure 3: Resize the color space.

Figure 4: Quantize the color space.

2.3 Step 1-2: Segment the input image to superpixels

Input The input color image ($m \times n \times 3$ matrix), m is the width of the input image, n is the length of the input image.

Output Superpixel segmentation matrix ($m \times n$ matrix), the value of the pixel in this matrix is just a label to indicate the superpixel it belongs to, and pixels of the same superpixel are labeled the same value.

Implementation There are a lot of superpixel segmentation algorithms, I choose one from OpenCV.

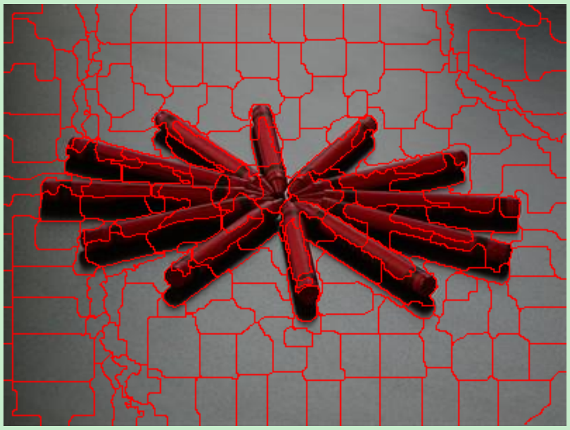

Figure 5: Superpixel segmentation.

2.4 Step 2: Compute features of each superpixel

Input Image information matrix, superpixel segmentation matrix.

Output h histograms, h is the number of superpixels after segmentation.

Implementation I computed color histogram feature of each superpixel.

2.5 Step 3: Compute superpixel feature contrast

Input h histograms.

Output h values, each value indicates the global feature contrast of a superpixel.

Instructions I compute global regional contrast, which means that the saliency of a superpixel is computed as its feature contrast of all the other superpixels in the image.

Theory The formulation for histogram distance is as follows:

$$\chi^2(\mathbf{h}_1, \mathbf{h}_2) = \sum_{i=1}^{b} \frac{2(h_{1i} - h_{2i})^2}{h_{1i} + h_{2i}} \tag{1}$$

where $\mathbf{h}_1$ and $\mathbf{h}_2$ are color histograms of two distinct regions, h_{1i} and h_{2i} are the ith component of $\mathbf{h}_1$ and $\mathbf{h}_2$ respectively, b is the number of histogram bins. Moreover, both histograms are normalized, i.e. their entries sum up to one.

2.6 Step 4: Convert superpixel saliency to pixel saliency

Input h values, superpixel segmentation matrix.

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

Implementation I assign all the pixels of the same superpixel the same saliency value.

2.7 Step 5: Use priors to enhance the result

Input Initial saliency map.

Output Final saliency map.

Implementation I choose Center prior for this assignment.I make a center map and use it mutiplying the initial saliency map to get the final saliency map.

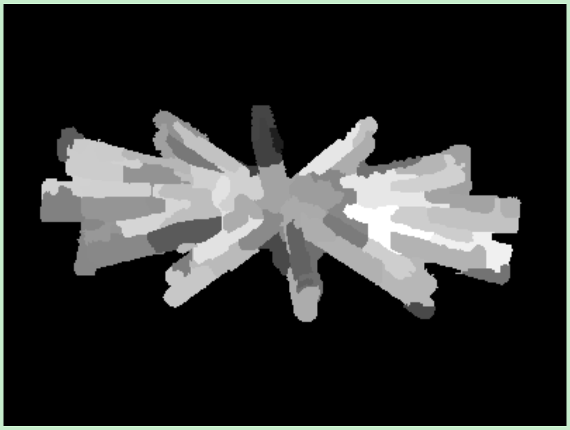

Figure 6: Initial saliency map.

Figure 7: Center map.

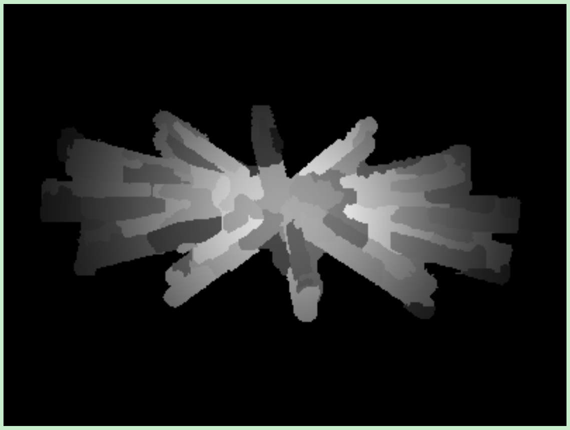

Figure 8: Final saliency map.

2.8 Supplement: I use segmentation to show the final result.

Input Final saliency map.

Output Segmentation map.

Implementation I choose threshold segmentation for this assignment.

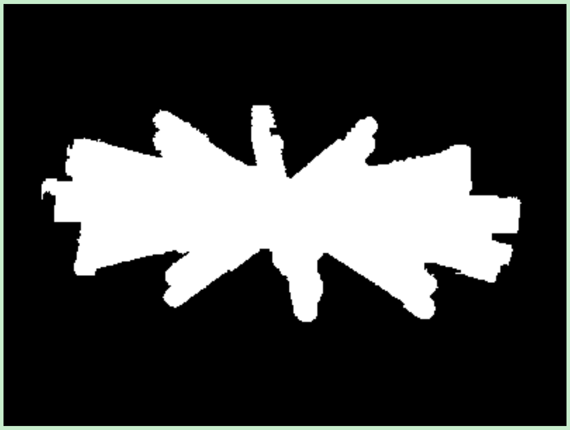

Figure 9: Segmentation map.