Assignment#1

Wang Ruchen

April 11, 2015

In this assignment, I used c++ and OpenCV to complete the assignment.

1. Strategy

The original image is shown in Figure 1:

图 1: image

1.1 Extract image information

I choose color information to complete this step. First, I quantize each color channel from 0-255 to 0-32. Then, every (R,G,B) is convert to a number. The result is shown in Figure 2. These numbers are saved in a $m \times m$ matrix.

1.2 Superpixel segmentation

OpenCV_Contrib is an extensive library of OpenCV. The library has a function about superpixel algorithm. I used the function to get superpixel segmentation picture, Figure 3. The number of superpixel areas is 192.

(a) (b)

图 2: quantize image

```
Ptr< SuperpixelSEEDS >    cv::ximgproc::createSuperpixelSEEDS (int image_width,
    int image_height, int image_channels, int num_superpixels, int num_levels,
    int prior=2, int histogram_bins=5, bool double_step=false)
```

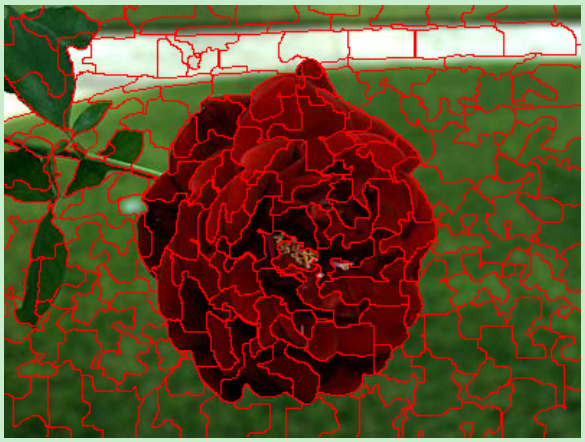

图 3: superpixel

1.3 Compute features of each superpixel

The color information of every superpixel area is saved in a histogram. So the total of histograms is 192.

1.4 Compute super pixel feature contrast

I used a 192×192 histogram to save the superpixel feature contrast. Every element in the histogram is a contrast between every two histograms.Then, compute the average contrast of the histogram's every row. The average contrast is the saliency value of every superpixel area.

1.5 Convert superpixel saliency to pixel saliency

The pixels in same superpixel area is assigned the same value. The initial saliency map is shown in Figure 4.

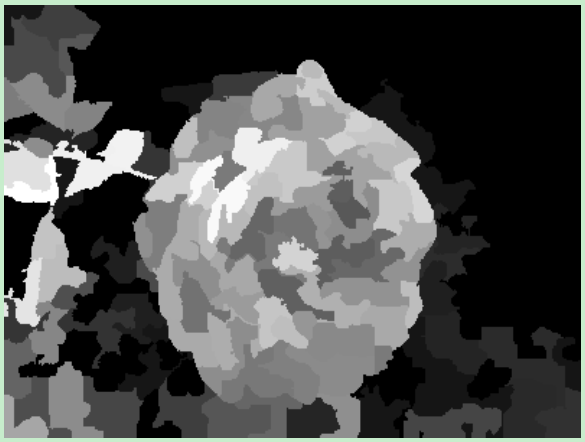

图 4: initial saliency map

1.6 Use prior

I used center prior to enhance the result. The center prior is shown in Figure 5.

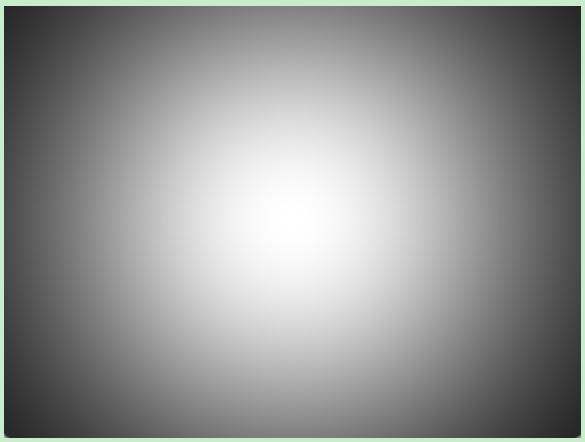

图 5: center prior

2. Result

Finally, the result of salient object detection is shown in Figure 6:

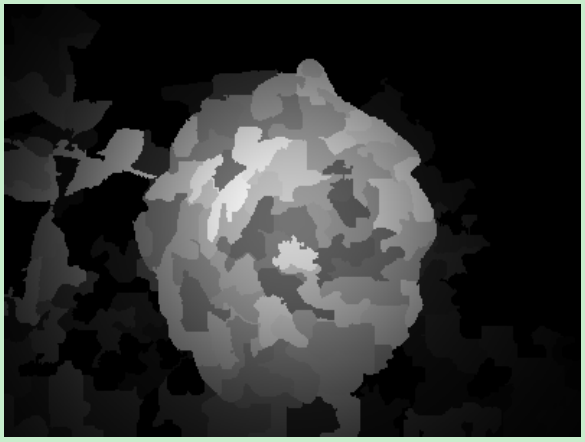

图 6: saliency map