Homework 4

Edge Detection and Region Segmentation

CSE473/573: Computer Vision and Image Processing

(Fall 2016)

Instructor: Chang Wen Chen

Teaching Assistants: Radhakrishna Dasari, Shuang Ma

Sugosh Nagavara Ravindra

sugoshna@buffalo.edu

Person#: 50207357

PROBLEM-1

We initially Gaussian Blur (kernel size=3) the input image to remove sharp edges.

a) After applying the DOG mask on UBCampus.jpg we obtain the following image

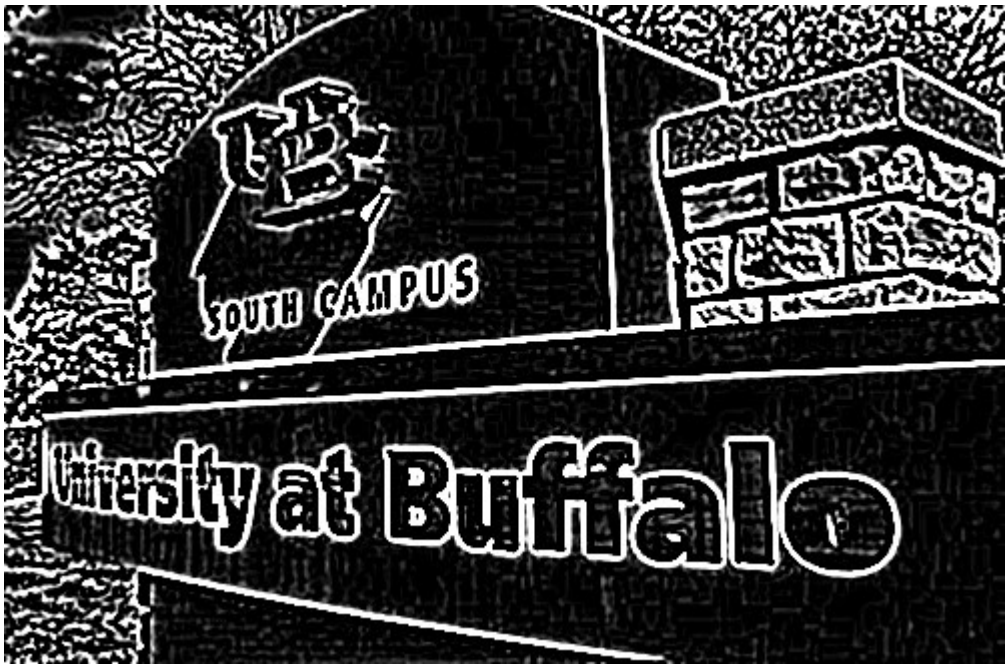
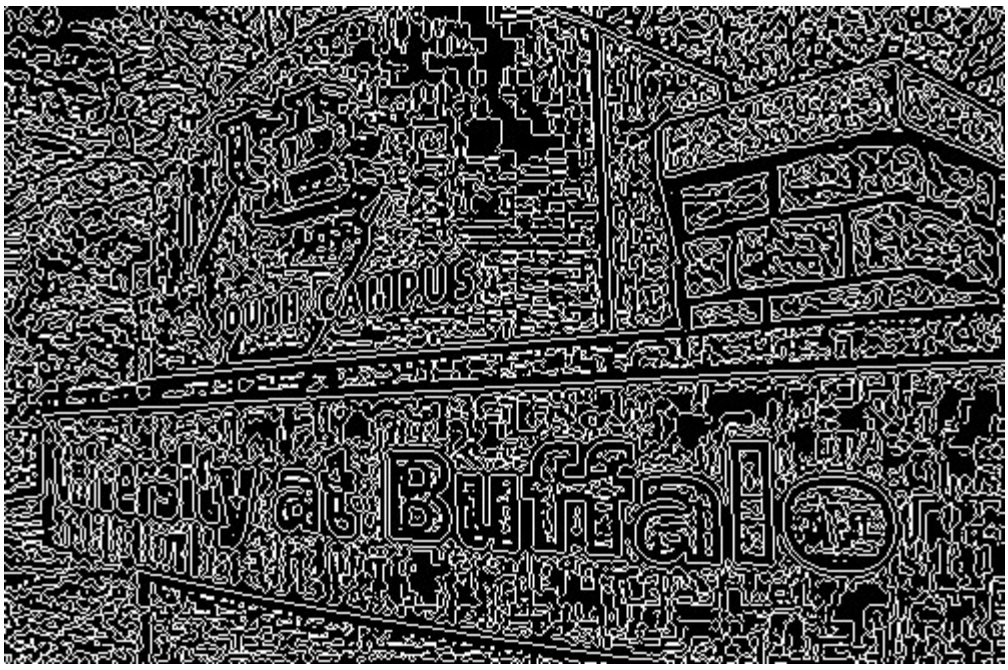


Fig 1.1

b) Zero Crossing of the DOG image is Fig1.2



c) To obtain the strong edges we need the first derivative support, that is obtained by Sobel magnitude. Following is the image of Sobel magnitude over x and y axis

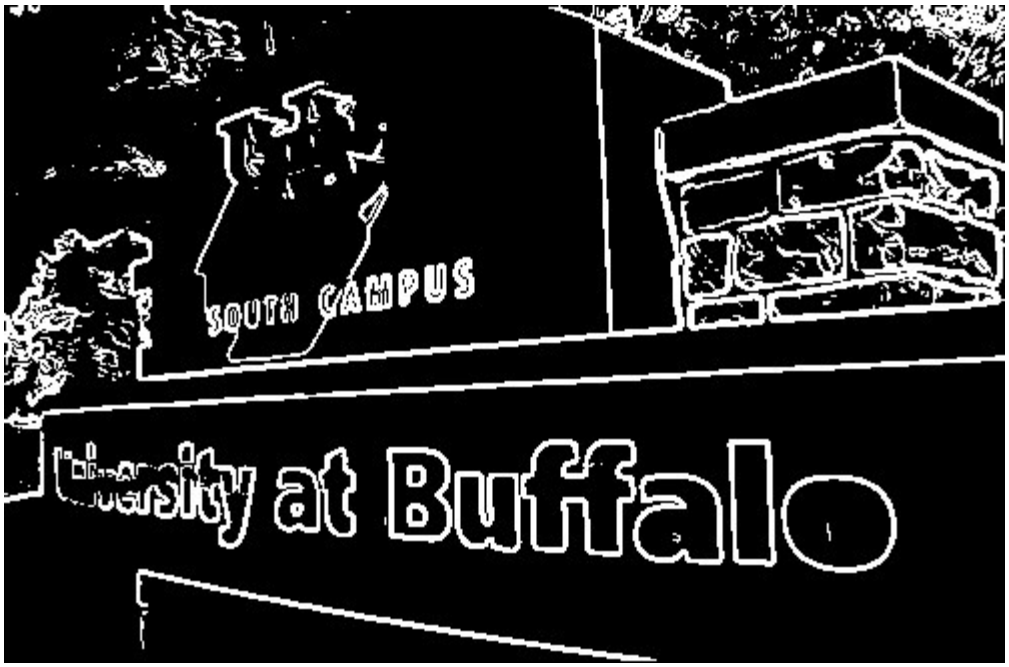
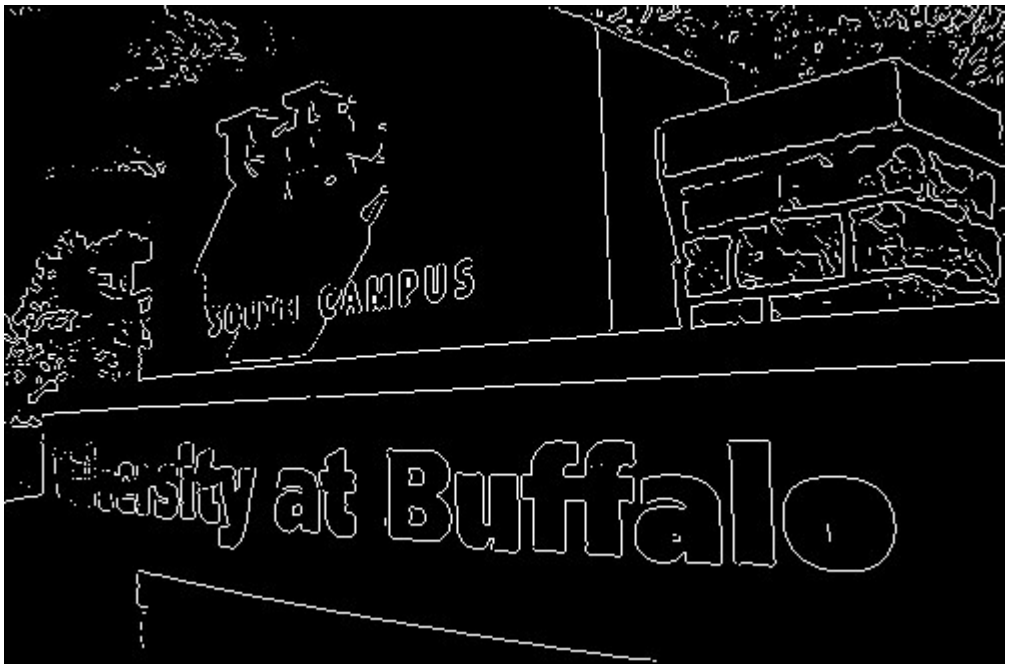


Fig 1.3

The Sobel image was threshold at a value of 128.

Zero-Crossing of DOG after removing weak edges is Fig1.4



d) LoG on the input image using the provided kernel gives the following image

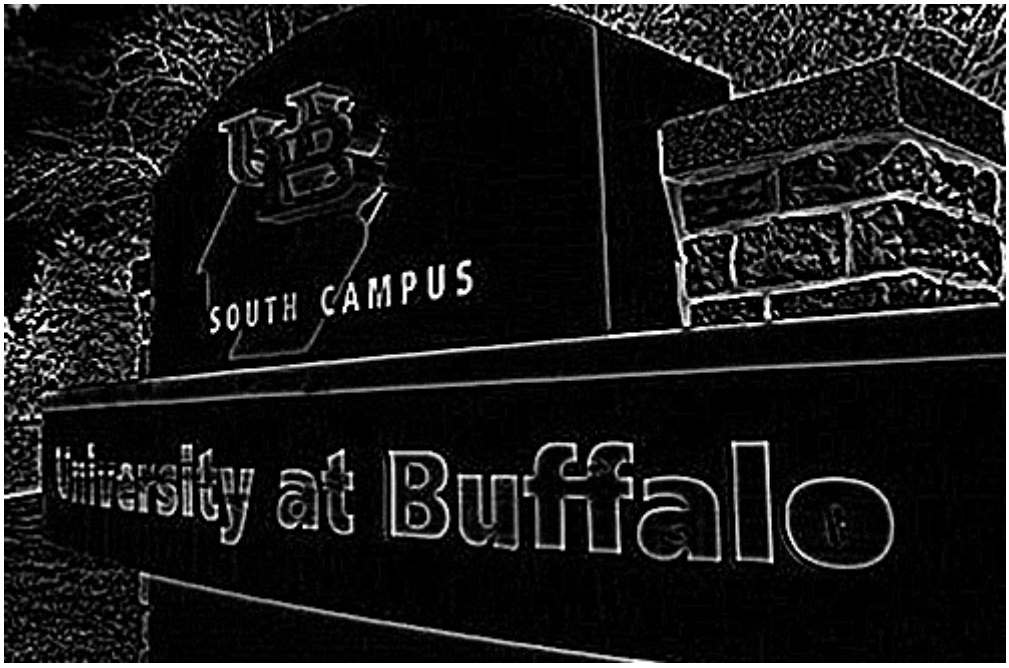


Fig 1.5

We can see that lot of noise has been removed in LoG when compared to DoG

Zero crossing of LoG gives

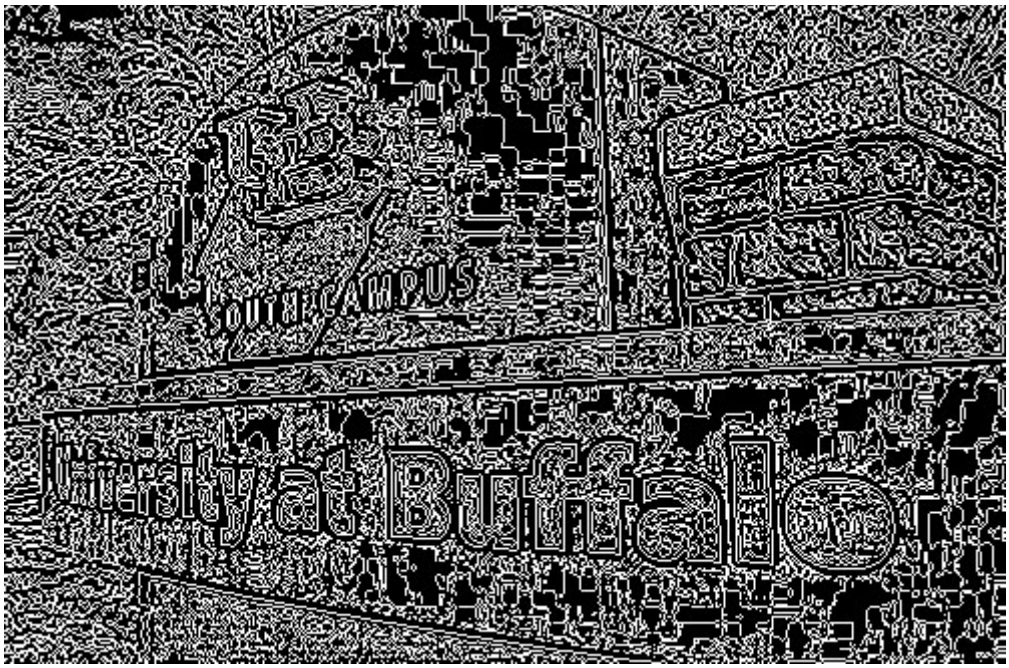


Fig 1.6

After removing weak edges from LoG zero crossing we get

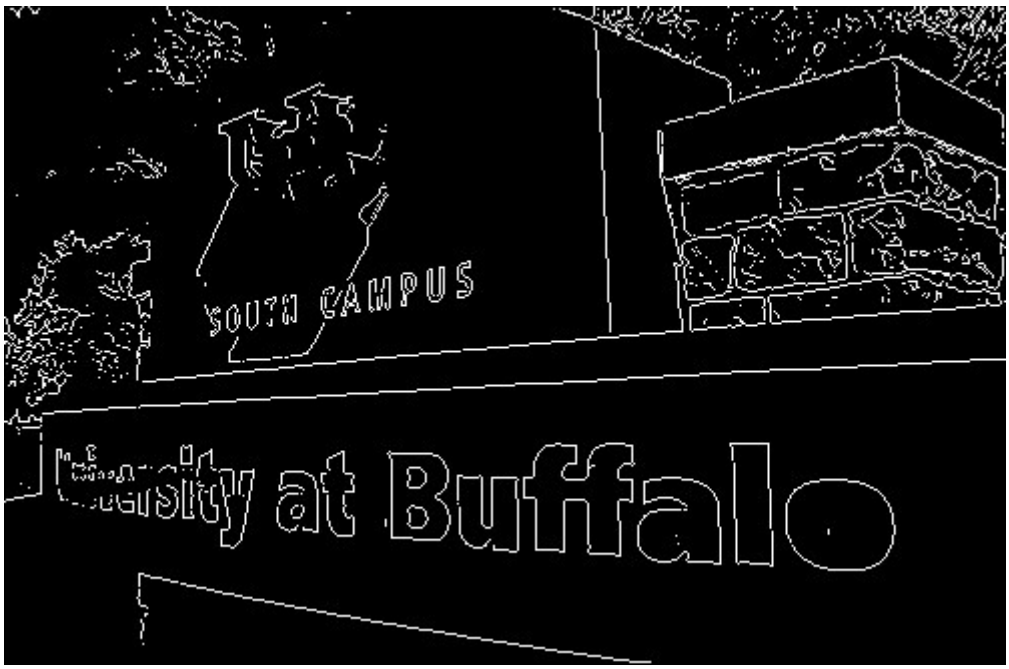


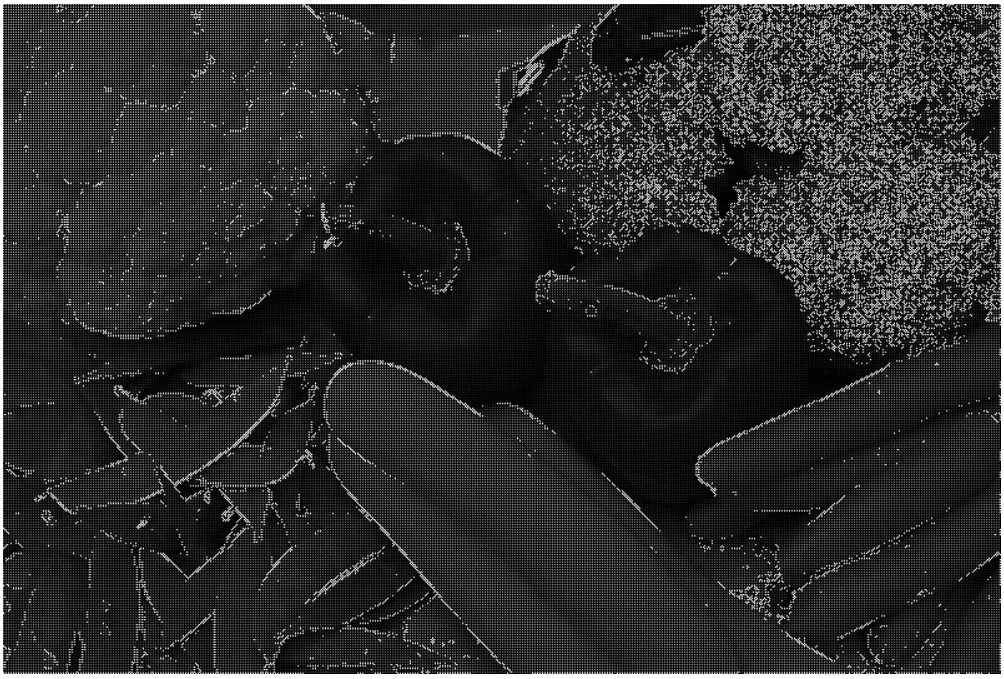
Fig 1.7

e) If we compare Fig 1.4 and Fig 1.5 we see that LoG has more noise in it. The noise in DoG is removed because of the first derivative support from Sobel to remove weak edges. If we use the same first derivative support of Sobel on the LoG zero crossing we obtain an edge image very similar to the DoG Edge image. This is the effect of First derivative support.

PROBLEM-2

Following is the image obtained after we implement the crack edge data structure in between the pixel values

After setting a threshold value of **30** on the crack edges we get the following image. 30 was the optimal value to be set as it neither selected too many crack edges nor too less.



The steps taken to get the best possible segmentation is as follows:

1) Obtain the Crack Edges
2) Set a threshold T1 and threshold the Crack Edges (30)
3) Create a perimeter dictionary which holds <region><endpoint_X><endpoint_y>. This is used to traverse the boundary of the region and check the Crack Edges at the boundry.
4) The implementation 6.18 algorithm is incomplete in this assignment as I was able to implement only the weaker criterion (6.34). The implementation is commented in the source code.

Following is the image obtained from basic boundary melting, where we consider each pixel to be individual region and merge them if their crack edge is equal to 0

Following is the image obtained by applying the weaker criterion which will merge regions together if they have their perimeter points less than some value T3(here 20)

We observe that some of the regions are merged together.