Problem 1

Application of Sobel filter in x axis y axis and taking squared average of both

(a) Obtain and display the DoG image by applying the following DoG mask to the test image
(b)Compute and display the zero-crossing of the DoG image obtained in (a)
(c) Compute and display the zero-crossing strong edges by removing weak edges that do not have first derivative support in (b)

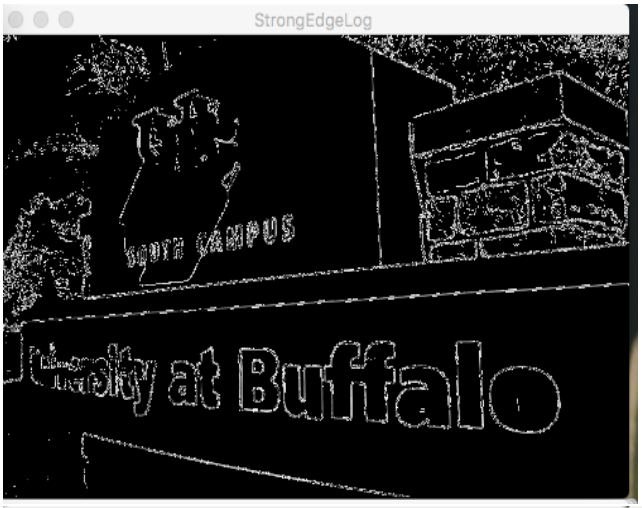
Threshold value set to 70

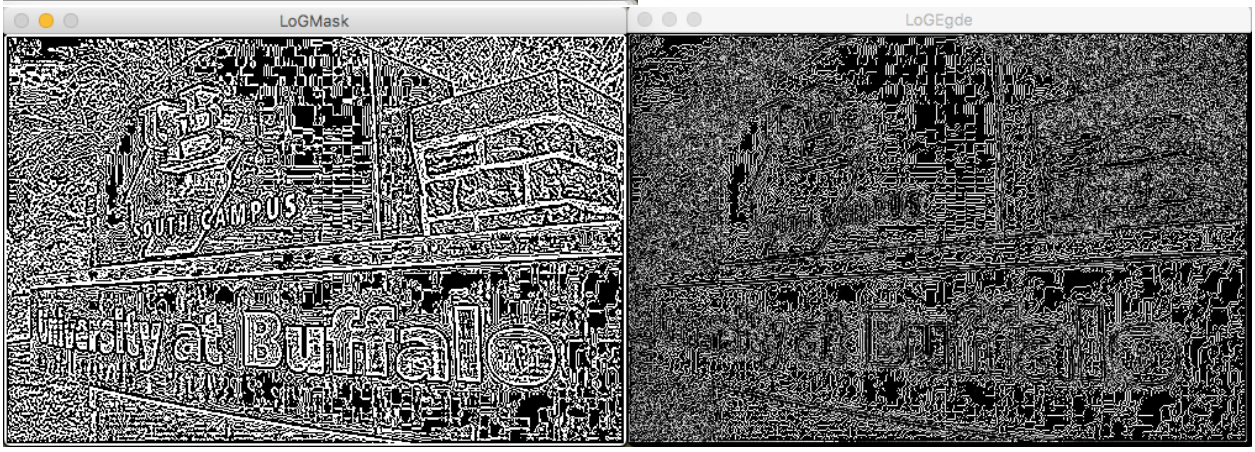
C) Removing weak edges

(A) DoG Mask

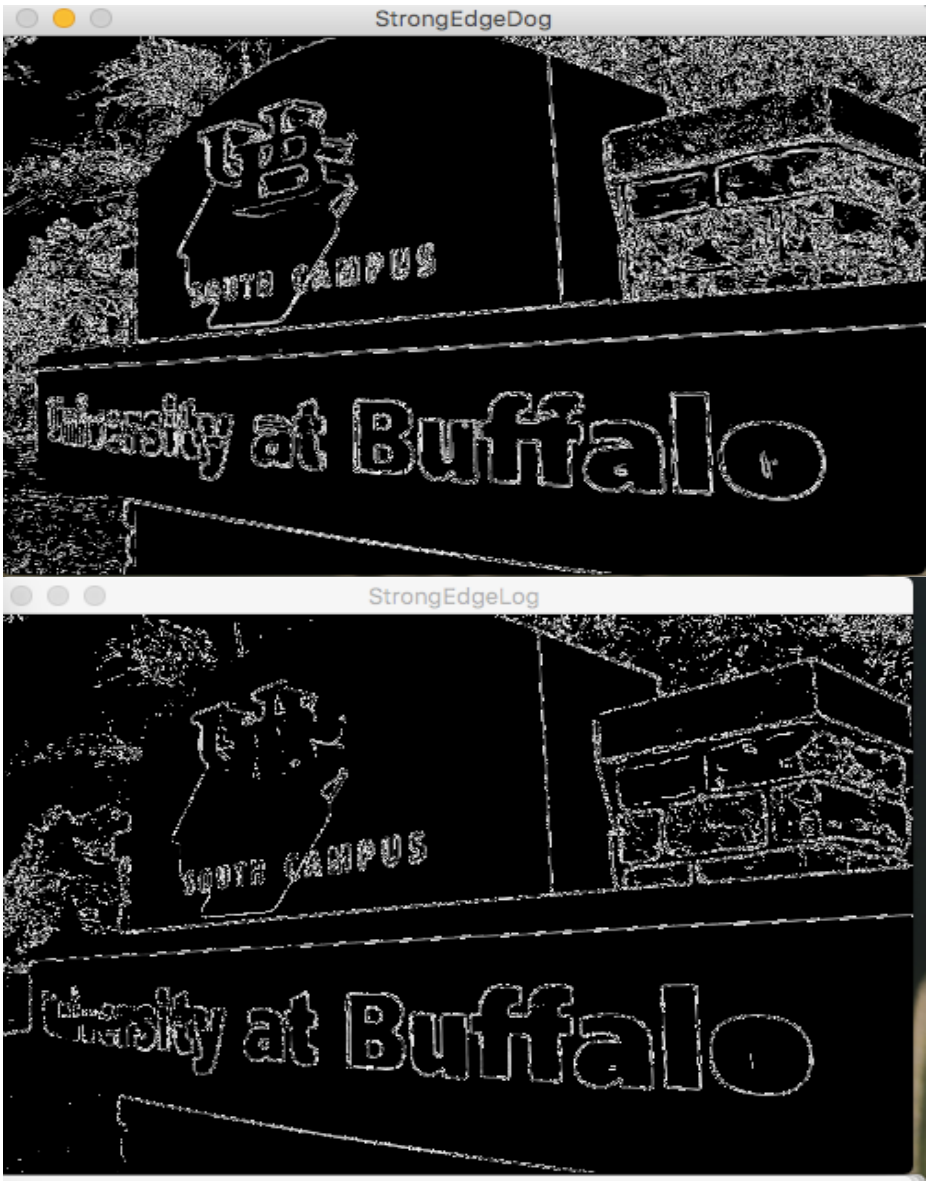
(B) Edge Detection

(d) Compute and display the LoG zero-crossing edges by applying the following LoG mask to the test image





(e) Compare the results in (c) and (d) and explain why the edges obtained in (c) and (d) are different



(e) Compare the results in (c) and (d) and explain why the edges obtained in (c) and (d) are different

LoG gives enhancement sharpens the edges but also increases noise.
DoG is a low pass filter which gives a blurring effect hence smoothening the image. In this process we loose the high intensity pixels. The edges are not so sharp
In DoG to get a result just as LoG we need to apply a lower threshold inorder to obtain similar result as LoG

Problem 2:

Region merging is an effective scheme for region growing based segmentation. Region growing may begin with each pixel within an image in which a pixel represents a single region initially. Regions will be merged to satisfy the region segmentation conditions as defined by Equations (6.30) and (6.31). Perform the following algorithm to the “Mixed Vegetables” test image through Python programming: (a) Perform Algorithm 6.18 – Region merging via boundary melting and display the results of the final region merging. Each student is required to set their own thresholds and also explain why such thresholds are selected.

Threshold t1= 12
Firstly, we consider every pixel to be an individual segment. If they are having weak edges between them and dissolve these into same region label enlarging the area from single pixel following the 4 neighbor principle.
In the crack image, we consider the crack edges having value < t1 as weak edges and dissolve them to form one region

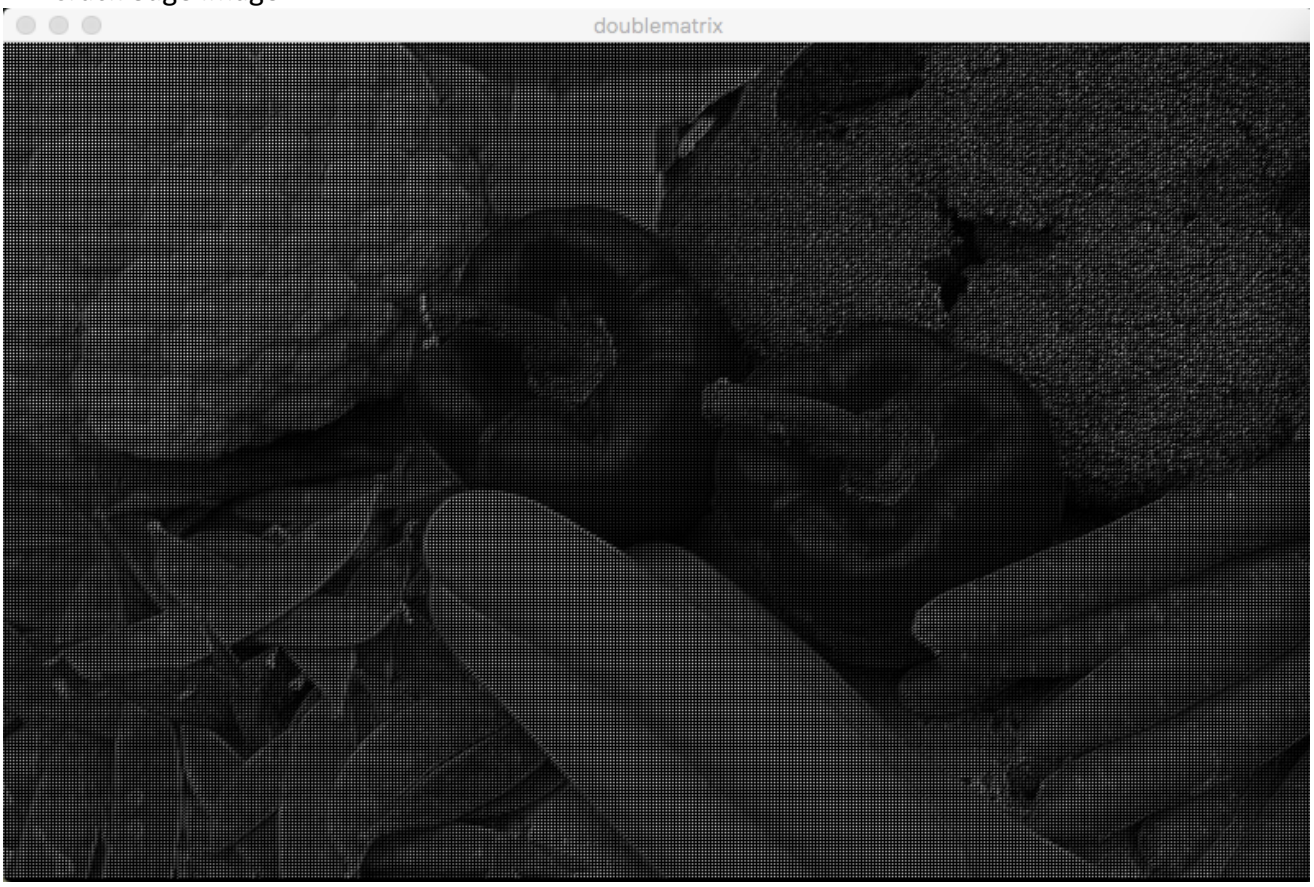
Threshold t2= 0.5
Here, we consider W/ min(l1,l2), such that ,
W is the count of weak edges at boundary of r1 and r2 regions respectively.
And l1 and l2 is the count of weak edges at boundary of r1 and r2 regions respectively.

Setting a threshold low as 0.5 keeps in check that there is optimal ration maintained between the perimeter of smaller region and the weak edges it shares with the region with larger perimeter

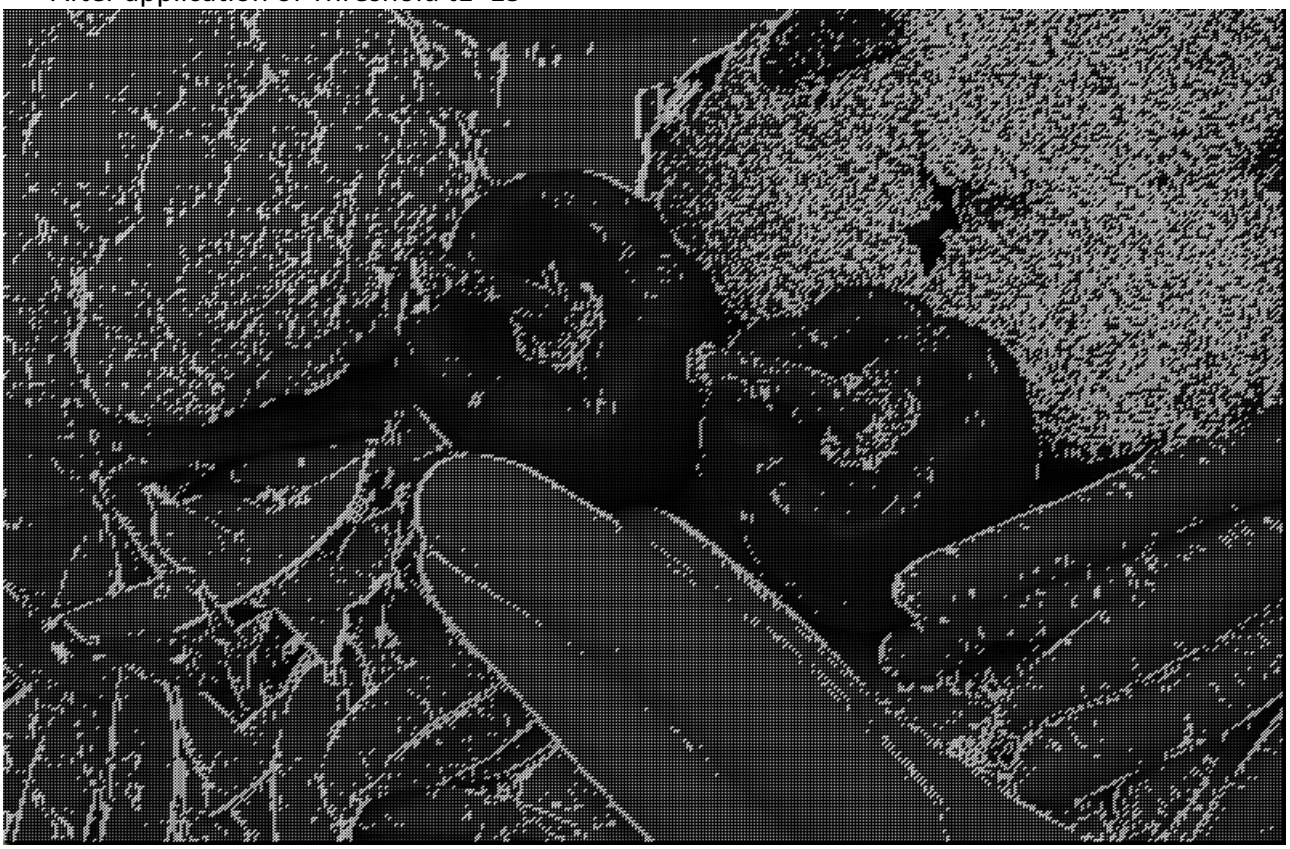
Threshold t3= 15
Here we are considering only the region that have significant amount of common edges in the boundary and merging regions accordingly
W> t3

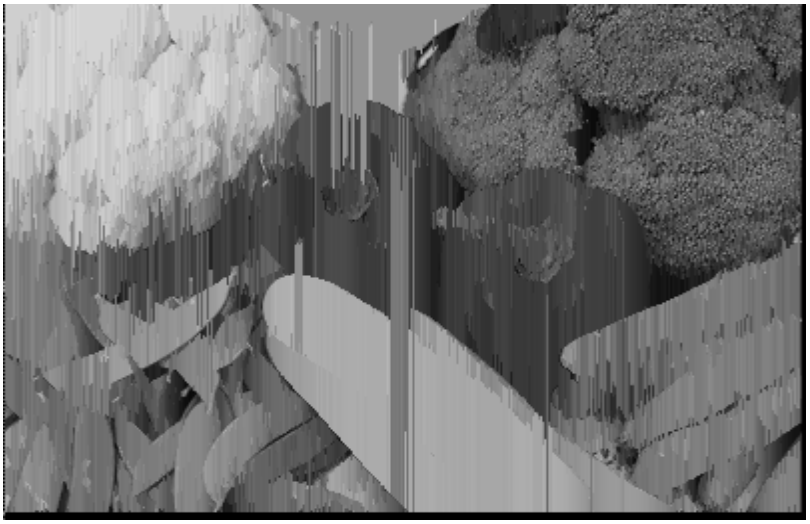
Original Image

- Crack edge image



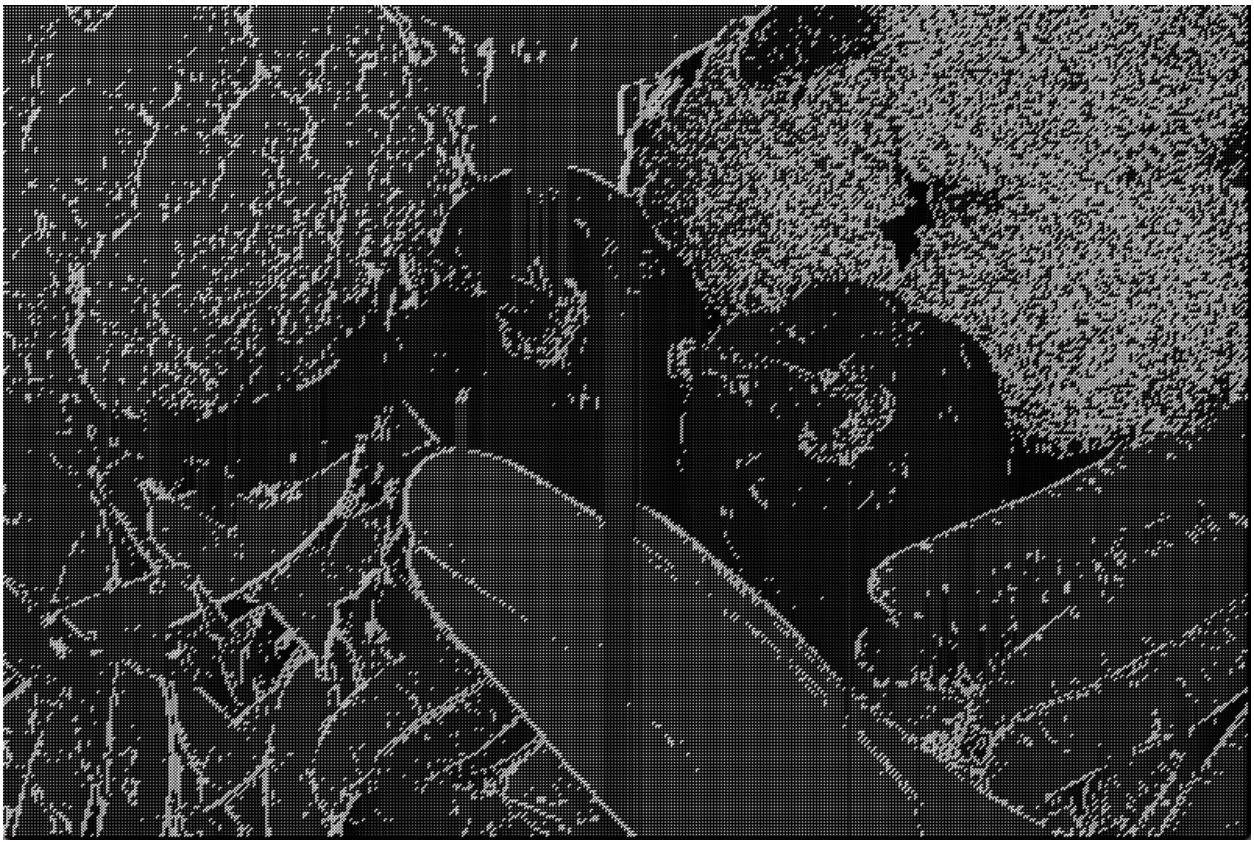
- After application of Threshold t1=15

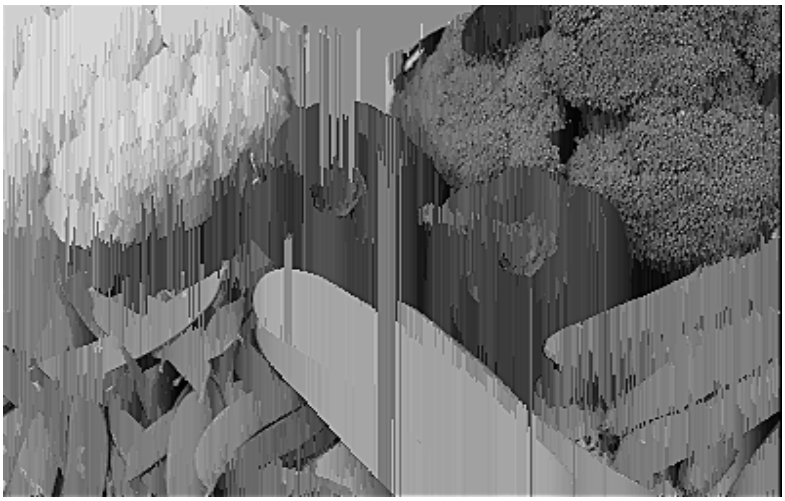




Obtained regions by combining the pixels of same region using average of the pixel values

- After application of Threshold t3=20





Obtained regions by combining the pixels of same region using average of the pixel values