CS555/455 Fall 2018

Assignment 2

(Due on October 4 by 11:59pm)

I. Questions (46%):

(1) (10%) Given following four masks, order them in that generating smooth images from light smoothness to heavy smoothness, Explain why.

M1:

1	1	1
1	1	1
1	1	1

M2:

1	2	1
2	4	2
1	2	1

M3:

0	1	0
1	4	1
0	1	0

M4:

0	0	0
0	1	0
0	0	0

M4,M3,M2,M1

This is ordered from least smooth to most smooth. M4 considers only the primary pixel when calculating the new value. Similar M3 only considers the north, south, east, and west pixels and puts priority on the primary pixel. While M2 considers all pixels it prioritizes the primary pixel again and the considers the north, south, east, and west pixels the second most. And finally M1 considers all pixels equally making it the most smooth.

(2) (8%) The filter for image enhancement can be designed by first-order derivatives and second-order derivatives. Compare the first-order derivatives and the second-order derivatives, which one is better for image enhancement. Explain why.

First order derivatives are very sensitive to noise and create thicker edges

Second order are even more sensitive to noise than the first order derivatives. It also produces a double edge.

It depends on the image. If the image is a light circle on a dark background you should use a first order derivative, but if the image is of a spiderweb against a light background it is better to use a second order derivative.

(3) (8%) An edge image (E) is generated by filtering a gray scale image (I) by a Laplacian mask (M). The sharpening image can be obtained by simply adding the original image (I) and the edge image (E). Show a single mask (S) based on the mask M such that the image sharpening can be implemented with one pass of the single mask.

$$M = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

S=[1 1 1]

[1 -7 1]

[1 1 1]

(4) (8%) To extract edge information of an image, people can either

(a) Blur the image first, then apply edge detector
or

(b) Apply edge detector first, then do the image blurring.

Which way is better, explain why.

It is better to blur the image first to reduce the noise before detecting edges. You get a similar effect by detecting edges first then blurring but you run the risk of losing some of the edges you find.

(5) (12%) Apply the median filter to remove the noises in the following image I:

4	4	4	4	4	4	4	4
4	4	4	48	4	4	4	4
4	4	64	64	64	64	4	4
4	17	64	64	96	64	4	4
4	4	64	85	64	64	8	4
4	4	64	64	64	64	4	4
4	56	4	4	23	4	4	4
4	4	4	4	4	4	4	4

(note: assume that all the pixels outside the image have value: 4)

(a) Use 3*3 square-shape median filter to filter image I, obtain image M1;

4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4
4	4	48	64	64	4	4	4
4	4	64	64	64	64	4	4
4	4	64	64	64	64	4	4
4	4	56	64	64	23	4	4
4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4

(b) Use 5*5 cross-shape median filter to filter image I, obtain image M2;

4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4
4	4	64	64	64	64	4	4
4	4	64	64	64	64	4	4
4	4	64	64	64	64	4	4
4	4	64	64	64	64	4	4
4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4

(c) Compare M1 and M2, indicate which filter is better, and explain why.
It depends on how much fine detail the image has. For example if the "noise" in the picture above is actually fine detail then the square filter would be better for filtering the image. Otherwise the cross filter would be better because the cross formation considers the main pixel most and then the neighbors in the primary directions. Because it accounts for fewer pixels it has a less likely chance it will consider noise.

II. (44%) Programming: Scale Space using Laplacian of Gaussian operator; Unsharp Masking, and Sobel Operator;

Description: Unsharp masking, Sobel operator, Laplacian of Gaussian operator, and Scale Space filter can be used for edge detection and enhancement. Apply these operators to a gray scale image to generate the edge image. In this assignment, you will implement these algorithms, and get various details of edge information in a given gray scale image.

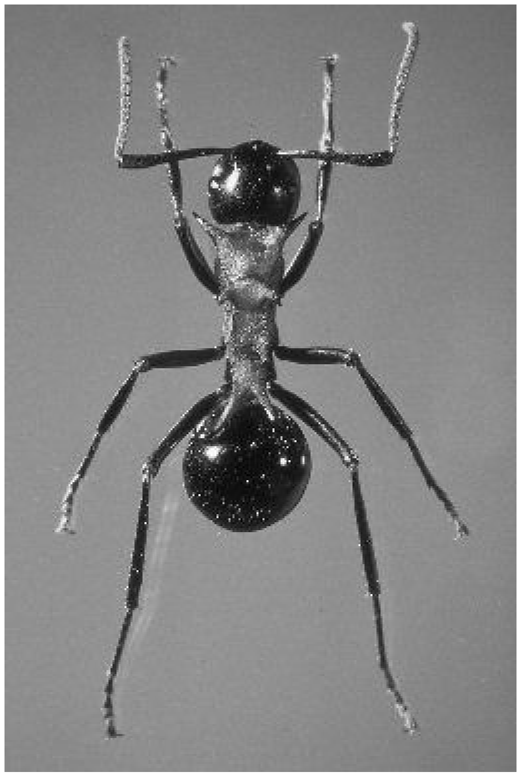
Your implementation:

(1) Obtain image f1 ("basel_gray.bmp) and f2 ("ant_gray.bmp), and display;
(2) (10%) Apply the Unsharp Masking method to enhance the edges of images(f1, f2) and obtain the enhanced image E1 and E2;
(3) (10%) Apply the Sobel operator to the two images (f1, f2) and generate the edge image Es1 and Es2;
(4) (10%) Generate LOG Mask1 7*7 (sigma = 1.4) and Mask2 11*11 (sigma=5.0).
(5) (14%) Apply Mask1 and Mask2 to intensity images (f1, f2), obtain two sets of edge images (E1_1, E1_2) for f1, and (E2_1, E2_2) for f2; Explain which images have more detailed edge information, and explain why.
(6) (Extra points 10%) Based on two different scales defined by the above Mask1 and Mask2, design an algorithm to find the zero-crossing points for fine edge detection in a scale space.
(7) (Extra points 10%): Apply Canny Edge detection approach to extract the edges in the image f1 and f2.

II. Hand-in

- **Code package:** Your program package includes all the files that can be re-compiled, executed, and demoed.
- **Write-up (10%)**: Report the experimental results along with figures and your description of the algorithms you designed and implemented, so that the reader can understand the method you used. Note: in the report, you should include following sections:
  (1) Author's Name and Email.
  (2) Purpose of the project
  (3) Method
  (4) Results
  (5) Bug report (if any), which includes "PARTS THAT ARE NOT COMPLETE" indicating any uncompleted parts of the project and any BUGS in your program that you are aware but have no idea why they exist.
  (6) Report extra work for extra credit (if any)
  (7) References (if any): Document here any sources, books, internet resources you have benefited from.

- Submit your code and report (in a .ZIP file) through blackboard on/before the due date.



Ant Image


Basel Image