EBME/CSDS 361/461. Homework 1. Spring.
David Wilson, Ph. D.
Department of Biomedical Engineering
Homework: Spatial filtering and edge detection

Please be aware that another assignment will soon follow, so it is best to start on this one immediately. Please upload your answers to canvas in the appropriate place.

All homework assignments must be completed individually (no teams/groups). You must understand the coding which is done. More complex programming will soon be coming, so you need to know about how to display images, etc.

Your responses to each question to each should include:

- A short discussion of each problem. ***Code/images alone will not suffice.*** A brief description/explanation should accompany each image/algorithm.
- Any processed images
- All MATLAB code used to generate results and images. NOTE: MATLAB code may be randomly tested by TA for functionality!
- Be sure to comment your MATLAB/Python Scripts thoroughly! We need to understand your approach to each problem.

Please submit the following files in Canvas:

- Your responses to each question in PDF format, titled "`[caseID]_HW01.pdf`"
- All generated images, MATLAB/Python code in one zip file titled "`[caseID]_HW01_Code.zip`"

Please read the question descriptions carefully. Be sure to address every question. We try to give you sufficient information to get you successfully started on this simple, but interesting, project.

Images will be provided by the TA. *(note: the .mat file is simply the MATLAB data representation of the corresponding .tiff image)*

- EDGES.mat, edges.tiff
- EINSTEIN.mat, EINSTEIN.tiff
- EINSTEIN_blur.mat, EINSTEIN_blur.tiff
- TUMOR.mat, tumor.tiff
- disk_img.mat, disk_img.tiff

Questions

1. In MATLAB, the function **conv2** has an argument "shape," why? Explain each possible value of the argument "shape." (Create a small (say 20x20) flat test image of grayscale value 100 and filter it with a 5x5 kernel of constant value 3 under different "shape" values. Explain results.

2. Investigate unsharp mask edge enhancement. Perform edge enhancement on the degraded image, EINSTEIN_blur using unsharp mask processing. Your task is to sharpen the image to look like the original image, EINSTEIN prior to burring. List the parameters at your disposal to optimize. Manually adjust parameters to get what you consider a "good" result. You can start with a 5x5 kernel. (*NOTE: your resulting image from filtering may not exactly match EINSTEIN prior to blurring. It should be visibly improved though*.) Print the output image and describe your filtering process.

   The next step is *optional*, but it will allow you to think of an innovation. Can you think of a way to create an optimal result here with a grid search? Really, there is no need to do it. Just describe it.

3. Investigate "segmentation" of a tumor within a brain image:
   a. Filter the image TUMOR.mat with a Sobel filter using **imfilter**. The output will be a grayscale image.
   b. Interactively determine a threshold value to identify edges of interest (the tumor). Try to create a minimum number of "extra" edges within the brain. The output should be a binary image. (Do not be concerned about the skull and skin surface which could be easily edited out by image cropping.)
   c. Print the filtered grayscale image and the thresholded binary image. What is your recommended threshold value?

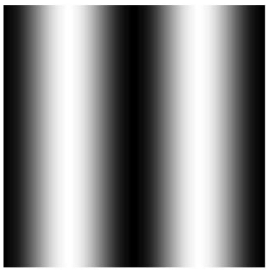
4. Using the function, **edge**, investigate edge detection on the image, EDGES.mat. The edge function creates a binary image with edges marked as foreground. Algorithms include some described in class as well as some others. Algorithms are: Sobel, Prewitt, Roberts, Laplacian of Gaussian, zero crossing, and Canny.

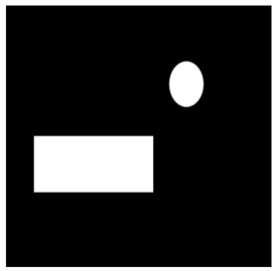
   Include a result from each method above in your report. In addition, answer the following questions.
   a. Which method(s) give a biphasic edge response? How do you know?
   b. Does the Sobel filter give a rotationally invariant response? In addition, look at the result of filtering the disk (disk_img.mat).
   c. Which method gives the "best" edge detection? Defend your answer.

5. Very briefly describe the Canny edge detection algorithm. Especially remember to describe hysteresis thresholding.

6. **EBME 461 students.** Write a MATLAB program to show that a "7x7 averaging filter" obeys linearity. Create two artificial test images, I_1 and I_2. One should be a sinusoid. The other should be an image containing a rectangle and a disk. If f is the filtering operation, show numerically that $f(a.I_1 + b.I_2) = a.f(I_1) + b.f(I_2)$. *(The period is a common way to indicate multiplication.)* How small is the difference between the images from the left and right sides of the equation? Do you think that your results are consistent with the filter being linear?

It should be understood that there are limitations to this analysis. A proof would require one to show this for all values of a and b, and images I1 and I2. In general, one cannot perform a proof numerically. Below are examples of a sinusoid image (I1; left) and a rectangle and disk image (I2; right). Please create test images that look something like these.





Functions of particular interest

Required MATLAB functions:
imfilter, **conv2**, **edge**

Useful MATLAB functions:
imread, load, fspecial, imshow, imagesc

Python Functions:
scipy.ndimage.correlate, **scipy.ndimage.convolve**, **cv2.filter2D**

For further understanding: Read about image display in Chapter 2 of *Digital Image Processing in MATLAB.*