Homework Assignment #3B

Due Date: Wednesday, 02 October, 2013 at 11:59pm (MDT)

<u>What to Submit</u>

Please submit a single <u>pdf</u> file via Learning Suite. For this homework you should turn in:

1. Written Exercises (preferably submit online with programming portion, otherwise give directly to me no later than 5pm Friday)
2. Programming Exercise Write-up (as described below) and all code that you've written to complete the exercises.

<u>WRITTEN EXERCISES</u>

1) Compute by hand the spatial filtering of the following signal by the following mask/kernel:

Signal:

1	3	4	2	1	3	2	1	3	1

Mask/Kernel:

1	1	1

You only need to produce an output signal that is as long as the input signal. How do you handle the boundaries of the finite-length signal?

2) Compute by hand the spatial filtering of the following signal by the following mask:

Signal:

1	3	4	2	1	3	2	1	3	1

Mask/Kernel:

1	2	1

As before, you only need to produce an output signal that is as long as the input signal.

3) Suppose the following signal was put into a system with the following impulse response. What would the result be?

Signal:

1	3	4	2	1	3	2	1	3	1

Impulse response:

1	2	3

Unlike the previous two exercises, treat the signal as an infinite one with samples before/after these as zeroes. Your results should thus have 12 elements. Remember to flip the kernel appropriately depending on which method you use to calculate the convolution.

4) Compute by hand the spatial filtering of the following image and mask:

Image:

1	3	4	4	1
1	2	5	6	3
1	3	2	3	2
2	2	1	2	1
1	3	2	1	1

Mask:

1	1	1
1	2	1
1	1	1

Your result should be 5x5. State how you handle the boundaries of the image.

The following exercise is from the Gonzales and Woods (G&W) textbook:

5) G&W 3.19b

   a) Develop a procedure for computing the median of an *n* x *n* neighborhood [for reference]

   b) Propose a technique for updating the median as the center of the neighborhood is moved from pixel to pixel

   The idea here is to take advantage of the reusable information as you slide the median-filtering window across the image. Can you avoid having to sort/select again from scratch each time?

PROGRAMMING EXERCISES

Please complete the following exercises in your programming environment of choice (nominally MATLAB or Python/NumPy):

PART A: Arithmetic Operations

Implement one of these two projects:

1) The image in file stuff1.png contains some assorted paraphernalia. The image in stuff2.png has one less object. Write a program to locate the missing object and return the (x,y) coordinates of its center of gravity. You will want to use a combination of arithmetic operations, level operations, and your own creativity.

2) The files in (frames.zip, frames.tgz) contain a sequence of images corrupted by noise. Average 2, 5, 10, 20, and 40 of these images together to reduce the noise. Display the result for each. Why does this work? How many is enough? How can you tell?

PART B: Spatial Filtering

1) Implement the following smoothing techniques and apply them to the input image noisygull.png. (You might also want to try it on other images as well, but you only have to turn in results for this one.) Explain the results for each method and mask/kernel size.

   - **Uniform averaging kernels** ("box filters") of size 3x3, 5x5, 7x7, and 9x9
   - **Median filtering** with size 3x3, 5x5, 7x7

You must code the spatial filtering (convolution or correlation) and median filtering routines yourself.

Hint: for different-sized masks, code it just once and pass the size of the mask as a parameter.

2) Using the input image whitebox.png, separately apply the following masks and explain the results for each:
   - **Uniform averaging kernels** of size 3x3, 5x5, 7x7, and 9x9 (reuse the your code above)
   - Each of the **Sobel kernels** (first derivatives in x and y respectively). Note: each of these should be normalized by 8.
   - The **Laplacian kernel**. No normalization required mathematically, but be careful: the results can be in the range [-255*4,+255*4] or larger depending on the kernel used.
3) Use the Sobel kernels to calculate gradient magnitude images for the input images whitebox.png and blocks.png.
4) Again calculate the gradient magnitude image for the input image blocks.png, but blur the image first by a uniform kernel of size 3x3. Repeat this for blurring kernels of 5x5, 7x7, and 9x9. What happens?
5) Use unsharp masking to sharpen the blocks.png image. Try adjusting the sharpening strength and/or radius of the blurring to find a combination you find effective or pleasing. Note: Your results should be sharpened, but there should be no other side-effects (such as darkening or lightening of the overall image). Be careful here to clip values outside the range [0,255] instead of inadvertently rescaling the values (losing contrast) or causing overflow/underflow.

Each of these might produce results that exceed the range [0,255]. For display or printing, you'll need to scale these results to the range [0,255]. If you need to continue working with that image in memory, be sure to scale a *copy* of the image.

CHALLENGE PROBLEM (Extra Credit)

The image tissue.png is a real tissue sample used for cancer screening. Devise a technique (e.g. level operations, windowing, etc.) to separate the different parts of the tissue cells. The image tissueRegions.png has three regions drawn on it. Can you perform image operations that will allow you to accurately count how many cell nuclei (the darkest purple regions) are in each region?

For both parts of the programming exercises ([A] Arithmetic Operations and [B] Spatial Filtering), prepare a brief write-up as described below.

Programming Exercise Write-up

Please prepare a written write-up (preferably submitted as a PDF) which includes the following:

- All code that you wrote for the exercises
- The relevant images that were generated from applying the different arithmetic and spatial filtering operations).
- A brief explanation of what you did, any challenges you encountered, things that were difficult, unclear, etc.
- How long did the different parts (written, programming) of the assignment take you?