**Digital Image Processing (CSE/ECE 478)**
**Monsoon-2019**
**Assignment-2 (240 points)**
**Posted on: 16/08/19**
**Due on: 23/08/19**

---

Grade Table (for teacher use only)

| Question | Points | Score |
|---|---|---|
| 1 | 50 | |
| 2 | 15 | |
| 3 | 25 | |
| 4 | 35 | |
| 5 | 45 | |
| 6 | 25 | |
| 7 | 30 | |
| 8 | 15 | |
| Total: | 240 | |

1. (50 points) Edge Detection

   1. Apply the Canny edge detector(use `cv2.Canny` or MATLAB's `edge` function) to `bell.jpg` and `cubes.png`. Tweak the values of the arguments(minVal and maxVal), and report the values that give the best results for each image. *Hint:* For the bell image, try to detect as many edges in the bell as possible, while avoiding edges in the background. (20 points)

   2. Consider Roberts, Prewitt, and Sobel filters and Laplacian filters. Apply these filters on `barbara.jpg` and make observations upon comparing their outputs. Compare these with the output of Canny edge detector on the same image. (20 points)

      **Prewitt:** $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$ ; $M_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$

      **Sobel:** $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$ ; $M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$

      **Roberts:** $M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ ; $M_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

      $\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ $\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

   3. Add noise to the input image above using Gaussian sampling. Study the effect of applying the filters on noise-affected inputs. (10 points)

2. (15 points) Prove that subtracting the Laplacian from an image is proportional to un-sharp masking.

3. (25 points)
   1. Create a matrix of size 3x3 which when convolved with `box.png` results in a white line where the white meets the black. (10 points)
   2. Convolve `blur.jpg` with the above matrix and the transpose of the above matrix. Report your observations. (10 points)
   3. Follow the above steps for suitable images of your choice. (5 points)

4. (35 points) High-Boost Filtering
   1. Implement high-boost filtering on the image `bell.jpg` varying the window size and the weight factor and report your observations. (15 points)
   2. Repeat the above exercise on suitable images of your choice. (10 points)
   3. How is bilateral filtering different from high-boost filtering? (10 points)

5. (45 points) The implementation of linear spatial filters requires moving a mask centered at each pixel of the image, computing sum of products of mask coefficients and corresponding pixels.
   1. Implement an algorithm for low-pass filtering a grayscale image by moving a $k \times k$ averaging filter of the form `ones(k)/(k`$^2$`)`. (5 points)
   2. As the filter is moved from one spatial location to the next one, the filter window shares many common pixels in adjacent neighborhoods. Exploit this observation and implement a more efficient version of averaging filter. To appreciate the benefits of doing so, generate a plot of $k$ vs run-time for various sized images. The plot diagram should contain a line plot for each image size you pick. Use different marker types to distinguish the default implementation and improved implementation. Just to give you a rough idea, look at `https://imgur.com/a/0HtYlTE`. (15 points)
   3. Utilize the observation similar to above to implement an efficient version of a $k \times k$ median filter. Generate a plot figure similar to previous question. (25 points)

6. (25 points) Bilateral Filtering
   1. Implement bilateral filter and apply it to sky.png and noir.png. (15 points)
   2. Vary the effect of domain and range components of the bilateral filter and try to minimize the L2 distance from the ground truth images. Display the images and the approximate values for the sigma. (10 points)

7. (30 points) Applications of Bilateral Filter
   1. **Cross Bilateral Filter** : In low light imaging, images tend to be noisy and loose sharp edges. However, with flash, there is an unpleasing direct-lighting effect. Thus a cross bilateral filter can be used with the range and spatial filters acting on two different images. Use a cross bilateral filter on the images pots_flash.jpg and pots_no_flash.jpg to produce the output and report the parameters? (15 points)

2. **Inverse Bilateral Filter** : Does it makes sense to develop an inverse bilateral filter, which blurs an image at edges and preserves the homogeneous regions. If it makes sense, design it and suggest its applications (15 points)

(a) Image with Flash (b) No Flash Image (c) Intended Result

8. (15 points) **Image Restoration**
The image `Degraded.jpg` has been degraded in some way. Try to find out what kind of degradation has been applied on the image and try to restore it. The original image before degradation is `Clean.jpg`. Clearly explain what you did to restore the image.

(a) Degraded.jpg (b) Clean.jpg