UNIVERSITY OF PITTSBURGH

Department of Electrical and Computer Engineering

ECE 1390 Introduction to Image Processing

Computer Project 3 (Fall 2017)

Assigned: Nov 27

Due: Dec 10

References: Sections: 5.1-5.10, 9.1-9.6, and 10.1-10.6 (Gonzalez and Woods 3rd edition)

Problem 3.1: Spatial Filtering – in the following, g is a noisy input image, and $m \times n$ is the size of the neighborhood that defines the filter size.

- **a)** Write a function f_hat = aMean(g, m, n) that implements the arithmetic mean filter defined in Eq. (5.3-3).
- **b)** Write a function f_hat = geoMean(g, m, n) that implements the geometric mean filter defined in Eq. (5.3-4).
- **c)** Write a function f_hat=ctharMean(g, m, n, q) that implements the contraharmonic mean filter defined in Eq. (5.3-6). Here q is the order of the filter.
- **d)** Write a function f_hat = medianFilter(g, m, n) that implements a median filter of size $m \times n$
- **e)** Read the image circuitboard_gaussian.tif and use the functions aMean and geoMean to filter the Gaussian noise
- **f)** Read the images circuitboard_salt.tif and circuitboard_pepper.tif, and use the function ctharMean to filter the salt noise and pepper noise.
- **g)** Read the image circuitboard_saltandpepper.tif and use function medianFilter to filter the salt and pepper noise.

Problem 3.2: Erosion and Dilation – In the following, I is an input image of size $M \times N$, B is a structuring element (SE) of size $m \times n$. Hint: Similar to convolution and correlation, to accommodate all possible excursions of B, pad I with m rows above and below and n columns to the left and right.

- **a)** Write a function E = morphoErode (I, B) for performing morphological erosion of binary image I with SE B.
- **b)** Write a function D = morphoDilate (I, B) for performing morphological dilation of binary image I with SE B.
- **c)** Read the image UTK.tif. Using a 3×3 SE of 1s, erode this image successively three times and display your result.
- **d)** Read the image UTK.tif. Using a 3×3 SE of 1s dilate this image successively three times and display your result.

Problem 3.3: Thinning and Pruning- In the following, I is an input image of size $M \times N$, B is a a sequence of K structuring elements of size $m \times n \times K$. Hint: Similar to convolution and correlation, to accommodate all possible excursions of B, pad I with m rows above and below, and n columns to the left and right.

a) Write a function, T= morphoThin(I, B) for thinning the foreground of binary image I using a sequence of K, 3×3 structuring elements contained in $3 \times 3 \times K$ array, B. Hint: Use the SEs from Fig. 9.21. Implement hit or miss transformation as an erosion and you can use your function morphoErode from 3.2.a

b) Write a function P = morphoPrun(I, numthin, nnumdil) to prun the foreground of binary image I. The structuring elements and methods used are described in Fig. 9.25. Parameter numthin is the number of times the thinning algorithm is applied to delete end points and numdil is the number of dilations done at the end. Generally you will want numdil to be less than or equal to numthin. Hint: you can use your function morphoDilate from 3.2.b for dilation, and your function morphoThin from 3.3.a.

c) Read image UTK.tif and apply morphoThin to it. Present your results.

d) Apply morphoPrun to the thinned image from (c) and present your results. Hint: numdil is much smaller than numthin and start with a small number for numthin.

Problem 3.4: Edge Detection – In the following, f is the input image; type represents the type of kernel used to compute the gradient (prewitt or sobel corresponding to the kernels presented in Fig. 10.14) ; and T is the threshold which is in the range [0,1].

a) Write a function g = edgeMag(f, type, T) for computing the magnitude of the gradient of image f.

b) Write a function ang = edgeAngle (f, type, T) for computing the edge angle image.

c) Read the image building.tif, compute edge magnitude image using the Sobel kernels, and threshold the image with T=1. Present your result.

d) Read the image building.tif, compute the angle image, threshold the image with T=0. Present your result.

Problem 3.5: Otsu global thresholding

a) Write a function [g, sep, kstar] = otsuThresh(f) that implements the Otsu thresholding algorithm discussed in Section 10.3.

b) Read the image polymercell.tif and threshold it using function otsuThresh. Display your result to confirm that you get the same result as Figure 10.39 (d)