UNIVERSITY OF CALIFORNIA, SANTA BARBARA
Department of Electrical and Computer Engineering

ECE 178 **Image Processing** Fall 2020

Homework Assignment #3

(Due on Wednesday 10/28/2020 by 6 pm)

Problem # 1. Linear and shift invariant operators: Determine for the following operators whether they are linear and whether they are shift invariant. Justify your answers. (Where your answer is "no", it is enough to show a counter example).

a) Mean: $f[m,n] \to g[m,n] = \mu$, where constant $\mu = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f[m,n]$.

b) Gain: $f[m,n] \to g[m,n] = \alpha f[m,n]$, where α is a non-zero constant.

c) Gamma correction: $f[m,n] \to g[m,n] = f[m,n]^{\gamma}$, where γ is a non-zero constant.

d) Affine: $f[m,n] \to g[m,n] = \alpha f[m,n] + \beta$, where α and β are non-zero constants.

e) Thresholding or binarization: $f[m,n] \to g[m,n] = u(f[m,n] - T)$, where $u(\cdot)$ is the (continuous) unit step function, and T is a constant.

f) Coordinate flip: $f[m,n] \to g[m,n] = f[n,m]$

Problem # 2. Convolution with various boundary conditions, and some non-linear filtering: This problem is to be solved on paper (not computer). Consider the following image of size $M = 4, N = 3$. (The boxed pixel represents the $[0,0]$ position.)

$$f = \begin{array}{cccc} \boxed{3} & 2 & 3 & 5 \\ 4 & 1 & 3 & 4 \\ 3 & 2 & 4 & 4 \end{array}$$

which we would like to convolve with a filter having the familiar impulse response

$$h = \begin{array}{ccc} 1 & 2 & 1 \\ 0 & \boxed{0} & 0 \\ -1 & -2 & -1 \end{array}$$

and obtain an output image **of the same size**.

a) First solve while assuming the image is extended with zero padding.

b) Next, assume periodic extension and convolve. (Note: periodic extension means $f[m,n] = f[m+M, n+N]$) Comment on which pixel locations you knew in advance would show the same outcome in (a) and (b)?

c) Next, apply a 3×3 median filter to f under the zero padding extension assumption.

d) Finally, what is the impulse response h of the median filter? Is it surprising?

Problem # 3. *A hands-on problem.* Efficient image blurring using the concept of filter separability:
Consider the following bi-variate Gaussian function (can also be viewed as joint PDF of two independent random variables):

$$f(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\left(\frac{(x-\mu_x)^2}{2\sigma_x^2} + \frac{(y-\mu_y)^2}{2\sigma_y^2}\right)}. \tag{1}$$

It is centered at the mean (μ_x, μ_y) and its spread is determined by the respective standard deviations σ_x and σ_y, along the x and y directions. We will use (1) to construct a $2D$ filter kernel that belongs to the class of "moving weighted average" filters we discussed in class. Convolving an image with this kernel would blur it. In this problem, whenever a convolution is performed on an image, assume zero padding extension.

a) First, use the gaussian of (1) to create a discrete $2D$ matrix (also called a discrete 2D gaussian kernel) denoted $h[m,n]$, of size 9×9, with the origin $[0,0]$ at the top left corner of the matrix, i.e., $m \in \{0, 1, \ldots, 8\}$ and $n \in \{0, 1, \ldots, 8\}$. Set $\mu_x = \mu_y = 4$ so that the gaussian center coincides with the center of the matrix, set $\sigma_x = \sigma_y = 1.5$, and populate the matrix with values $h[m,n] = f(m,n)$ using (1). (*hint:* consider using *meshgrid* or nested for loops). Next, normalize the matrix elements so that they add up to 1.0 (i.e., they represent weights of a moving *average*) to obtain the filter impulse response (gaussian kernel) $h[m,n]$. Visualize $h[m,n]$ (use inbuilt visualization tools).
As an aside, note how many multiplications per pixel are required to convolve an image with $h[m,n]$.

b) Next, take a closer look at the gaussian function of (1) and explain why the filter $h[m,n]$, constructed using $f(x,y)$ in part (a), must in fact be a *separable* filter. This means that $h[m,n]$ can be written as an outer product of two 1D filters denoted $h_x[m]$ and $h_y[n]$. Visualize $h_x[m]$ and $[h_y[n]$ (use inbuilt visualization tools).
As another aside, note how many multiplications per pixel would be needed to blur the image using separable row and column filtering with $h_x[m]$ and $h_y[n]$.

c) Perform convolution on the attached image (*peppers.png*): First using $h[m,n]$, as if we did not realize it was separable. Show the result as well as the runtime (using *tic-toc* in matlab and *time* in python). Convolution with $h[m,n]$ should yield a blurry image. Next, exploit the concept of separable filters to blur the image by convolving with the 1D filters $h_x[m]$ and $h_y[n]$. Show the output and report the runtime. Of course, the output image after blurring with $h[m,n]$, regardless of whether we exploit separability, should be exactly the same - verify this. Also verify that the runtime ratio of the two approaches is consistent with the ratio of number of multiplications per pixel, as observed in parts (a) and (b). (Though not exactly the same, because we neglected additions and general overhead in the programs).