ECE 5273/CS 5273
Test 2

Tuesday, November 21, 2000
1:30 PM - 2:45 PM

Fall 2000

Dr. Havlicek

Name: SOLUTION

Student Num: ____________

Directions: This is an open notes, open book test. You have 75 minutes to complete the test. All work must be your own.

SHOW ALL OF YOUR WORK for maximum partial credit!

GOOD LUCK!

SCORE:

1. (25) ______

2. (25) ______

3. (25) ______

4. (25) ______

TOTAL (100):

1. **25 pts.** True or False. Mark the correct answer. Mark *True* only if the statement is **always** true.

TRUE	FALSE	
	X	(a) **2 pts.** For convolving two N × N digital images, multiplying DFT's instead of directly implementing the convolution reduces the computational complexity from N^4 to $N \log N$.
X		(b) **2 pts.** To implement the **linear** convolution of two 256 × 256 digital images by multiplying DFT's, it is generally necessary to zero pad both images to a size of 512 × 512.
	X	(c) **2 pts.** Median filtering of a 512 × 512 digital image can be efficiently implemented by multiplying DFT's.
X		(d) **2 pts.** In the linear image restoration problem, the Wiener filter reduces to the inverse filter if no additive noise is present.
X		(e) **2 pts.** In practical (real-world) restoration problems, the Wiener filter often fails to produce a visually appealing result.
	X	(f) **3 pts.** If regularization is applied to an image restoration problem where there is **no** additive noise, then the Lagrange multiplier should be set to the value $\lambda = 1$.
	X	(g) **3 pts.** For edge detection, the Laplacian operator is much less sensitive to noise than the gradient operator.
	X	(h) **3 pts.** The main problem with watershed algorithms is that they usually under-segment an image.
	X	(i) **3 pts.** Gray scale morphological filters can be implemented by applying binary morphological filters directly to gray scale images; the only difference between the two is the input image type.
	X	(j) **3 pts.** The image *girl2.bin* is a picture of the instructor before his sex change operation.

2. **25 pts.** For the window (structuring element) SQUARE(9), give the filter weights (coefficients) $\mathbf{A}^T$ for

(a) **5 pts.** A median filter:

$$\mathbf{A}^T = [0,0,0,0,\ 1,\ 0,0,0,0]$$

(b) **5 pts.** An average filter:

$$\mathbf{A}^T = [\tfrac{1}{9},\tfrac{1}{9},\tfrac{1}{9},\tfrac{1}{9},\tfrac{1}{9},\tfrac{1}{9},\tfrac{1}{9},\tfrac{1}{9},\tfrac{1}{9}]$$

(c) **5 pts.** An OS filter to perform morphological erosion:

$$\mathbf{A}^T = [1,\ 0,0,0,0,\ 0,0,0,0]$$

(d) **5 pts.** An OS filter to perform morphological dilation:

$$\mathbf{A}^T = [0,0,0,0,\ 0,0,0,0,\ 1]$$

(e) **5 pts.** The inner average OS filter INNER_AVE$_2$:

$P = 2$
$2P+1 = 5$

$$\mathbf{A}^T = [0,0,\ \tfrac{1}{5},\tfrac{1}{5},\tfrac{1}{5},\tfrac{1}{5},\tfrac{1}{5},\ 0,0]$$

3. **25 pts**. Pixels in the 6×6 image $\mathbb{I}$ shown below take values in the range $\{0, 1, 2, \ldots, 99\}$. The image is sent through a communication channel where it is corrupted by additive noise. The received image $\mathbb{J}$ is also shown below.

$\mathbb{I}$ =

11	**11**	**12**	**12**	**13**	**13**
11	**12**	**12**	**13**	**13**	**13**
12	**12**	**13**	**13**	**13**	**14**
12	**13**	**13**	**13**	**14**	**14**
13	**13**	**13**	**14**	**14**	**14**
13	**13**	**13**	**14**	**14**	**14**

$\mathbb{J}$ =

11	**11**	**33**	**12**	**13**	**13**
11	**12**	**0**	**13**	**13**	**13**
12	**12**	**99**	**13**	**13**	**14**
12	**13**	**13**	**89**	**14**	**14**
13	**13**	**13**	**45**	**14**	**14**
13	**13**	**13**	**1**	**14**	**14**

Design a nonlinear filter to restore the received image by attenuating the noise. Handle edge effects by replication. Explain your solution. Show the restored image $\mathbb{K}$ below. There is workspace on the following page.

- The noise has positive and negative values and makes the corrupted pixels much larger or smaller than their original values and than their neighbors ⟹ use MEDIAN. OPEN-CLOSE & CLOSE-OPEN would also work, but MEDIAN is much simpler.
- The noise is spatially correlated along columns of the image. ⟹ Use the window $\mathbb{B}$ = ROW(3). In worst case there will be two uncorrupted pixels and one corrupted pixel in the window. → So the median will always be an uncorrupted pixel.

Show the restored image here:

$\mathbb{K}$ =

11	11	12	13	13	13
11	11	12	13	13	13
12	12	13	13	13	14
12	13	13	14	14	14
13	13	13	14	14	14
13	13	13	13	14	14

- Why not ROW(5)? Because a larger window smooths more and will tend to change pixels that were not corrupted.
- Why not COL(3), SQUARE(9), CROSS(5)? Because all of these will have multiple corrupted pixels in the window. Also, SQUARE(9) will perform more smoothing than is needed.

⟹ So $\mathbb{K}$ = MED[$\mathbb{J}$, ROW(3)]

More Workspace for Problem 3...

4. **25 pts.** Apply local estimator-based edge detection to the 10×10 image $\mathbb{I}$ shown below to find edges within the dark boundary **only**. You do **not** have to find edges outside the dark boundary.

$\mathbb{I}$ =

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	15	0	0	0	45	45	45	45
0	0	15	0	0	0	45	45	45	45
0	0	15	0	0	0	45	45	45	45
0	0	15	0	0	0	45	45	45	45
0	0	15	0	0	0	45	45	45	45
0	0	15	0	0	0	45	45	45	45
0	0	15	0	0	0	45	45	45	45

For your windows, use $\mathbb{B}_U = \mathbb{B}_D = \mathbb{B}_L = \mathbb{B}_R$ = ROW(3). Use a median filter for the local estimator EST. Handle edge effects by replication. For the edge distance measure, use d=1. For the point operation, use $M(i,j) = \max\{|\Delta_x(i,j)|, |\Delta_y(i,j)|\}$. Choose an appropriate threshold τ.

You may use the blank images on the following page for work space.

$M(i,j)$ has only two values: 0 and 45.

Need $0 < \tau < 45$.

We'll take $\tau = 30$.

Show the binary edge map $\mathbb{E}$ below:

$\mathbb{E}$ =

0	0	0	0	1	1
0	0	0	1	1	1
0	0	0	1	1	0
0	0	0	1	1	0
0	0	0	1	1	0
0	0	0	1	1	0

More Workspace for Problem 4...

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	45	45	45
0	0	0	0	0	45	45	45
0	0	0	0	0	45	45	45
0	0	0	0	0	45	45	45
0	0	0	0	0	45	45	45
0	0	0	0	0	45	45	45

$J = MED[I, ROW(3)]$

0	0	0	0	45	45
0	0	0	0	45	45
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

$\Delta_y(i,j) = J(i+1,j) - J(i-1,j)$

$[\ (i,j) = (row, col)\ ]$

0	0	0	0	0	0
0	0	0	45	45	0
0	0	0	45	45	0
0	0	0	45	45	0
0	0	0	45	45	0
0	0	0	45	45	0

$\Delta_x(i,j) = J(i,j+1) - J(i,j-1)$

0	0	0	0	45	45
0	0	0	45	45	45
0	0	0	45	45	0
0	0	0	45	45	0
0	0	0	45	45	0
0	0	0	45	45	0

$M(i,j) = \max\{|\Delta_x(i,j)|, |\Delta_y(i,j)|\}$