ECE 5273
Test 2

Friday, May 9, 2025
10:30 AM - 12:30 PM

Spring 2025 Name: SOLUTION

Dr. Havlicek Student Num:

Directions: This is an open notes test. You may use a clean copy of the course notes as published on the course web site and a calculator. Other materials are not allowed. You have 120 minutes to complete the test. All work must be your own.

SHOW ALL OF YOUR WORK for maximum partial credit!

GOOD LUCK!

SCORE:

1. (20) ______

2. (20) ______

3. (20) ______

4. (20) ______

5. (20) ______

TOTAL (100):

On my honor, I affirm that I have neither given nor received inappropriate aid in the completion of this test.

Name: Date:

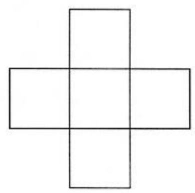
1. **20 pts**. True or False. Mark *True* only if the statement is **always** true.

TRUE	FALSE	
X		(a) **2 pts**. If $\mathbf{I}_1$ and $\mathbf{I}_2$ are two 512×512 digital images, then their linear convolution $\mathbf{J} = \mathbf{I}_1 * \mathbf{I}_2$ has a size of 1023×1023. Notes p. 5.44
X		(b) **2 pts**. If two digital images are both periodic with the same period, then their wraparound convolution equals their linear convolution. Notes p. 5.21
	X	(c) **2 pts**. Any grayscale morphological filter can be implemented by pointwise multiplication of DFT's if sufficient zero padding is applied. Morphological filters are nonlinear
	X	(d) **2 pts**. The bilateral filter is based on a product of two kernels (weighting functions): one that weights neighbors based on spatial distance and one that weights neighbors based on neighborhood similarity. Notes p. 6.79
X		(e) **2 pts**. Homomorphic filtering can be used to transform a multiplicative noise problem into an additive noise problem. Notes p. 6.118
X		(f) **2 pts**. The JPEG standard allows for arithmetic coding to be used as the entropy coding method; however, arithmetic coding is seldom used in JPEG compression of general consumer-type images. Notes p. 7.70 + Lectures
X		(g) **2 pts**. Edge thinning and edge linking are not required after application of the basic LoG edge detector (*i.e.*, the LoG *without* thresholding). Notes p. 8.85
X		(h) **2 pts**. In grayscale template matching, the normalized cross-correlation $\hat{C}_{\mathbf{B_T} \circ I(i,j), \mathbf{T}}$ always takes a value between 0 and 1 (inclusive). Notes p. 8.42
X		(i) **2 pts**. In anisotropic diffusion, the diffusion coefficients control the inhibition of smoothing over edges and encourage smoothing in non-edge image regions. Notes p. 8.94
OH MY!!		(j) **2 pts**. Trump will be elected President again in 2028.

2. **20 pts**. The 5×5 image $\mathbf{I}$ shown below has 3-bit pixels in the range $0 \leq I(m,n) \leq 7$:

$$\mathbf{I} = \begin{array}{|c|c|c|c|c|} \hline 4 & 7 & 2 & 5 & 3 \\ \hline 1 & 5 & 4 & 7 & 4 \\ \hline 0 & 5 & \mathbf{7} & 2 & 3 \\ \hline 3 & 7 & 1 & 2 & 4 \\ \hline 2 & 4 & 4 & 4 & 5 \\ \hline \end{array}$$

Denoising is to be performed by applying a median filter with the window $\mathbf{B} =$ CROSS(5) shown here:

$\mathbf{B}$ = 

Use Delman's fast bit-serial median filtering algorithm to compute the filtered output value for the middle pixel $\mathbf{I}(2,2)$ shown above as a large boldface "7."

Note: you do **not** have to apply the filter at any other pixels! Just apply it at the middle pixel $\mathbf{I}(2,2)$ *only* to find the filtered output value for that pixel *only.*

More workspace is provided on the following page in case you need it.

The windowed set is $B \circ I(2,2) = \{4, 5, 7, 2, 1\}$.

Following Notes pp. 6.56–6.58:

ORIGINAL DATA	FIRST CHANGE	SECOND CHANGE
1 0 0	1 0 0	1 0 0
1 0 1	1 0 1	1 0 1
1 1 1	1 1 1	1 1 1
0 1 0	0 0 0	0 0 0
0 0 1	0 0 0	0 0 0
→ 1	→ 0	→ 0

$$\boxed{\text{MEDIAN} = 100_b = 4_d}$$

3. **20 pts**. Pixels in the 6×6 image **I** shown below have 6 bits per pixel and take values in the range $\{0, 1, 2, \ldots, 63\}$. The image is sent through a communication channel where it is corrupted by salt & pepper noise. The received image **J** is shown below at right.

I =

16	16	16	48	48	48
16	16	16	48	48	48
16	16	16	50	48	48
16	16	16	50	48	48
16	16	16	50	48	48
16	16	16	48	48	48

J =

16	0	16	48	0	48
16	16	63	48	48	48
63	16	16	50	0	48
16	16	16	63	48	63
16	0	16	0	48	48
16	16	16	48	48	48

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 **K** below and compute the ISNR. There is workspace on the following two pages.

* MEDIAN is best against salt & pepper noise.
* Use the smallest possible window / structuring element.
  - ROW(3), COL(3), DIAG(3), and ANTI-DIAG(3) all perform poorly because of the noise hits that are close together.
  - SQUARE(9) can't handle pixel (4,4) ... gets 16 for the median.

⇒ USE CROSS(5).

$\boxed{\text{MEDIAN with } B = \text{CROSS}(5)}$

Show the restored image here:

K =

16	16	16	48	48	48
16	16	16	48	48	48
16	16	16	48	48	48
16	16	16	48	48	48
16	16	16	48	48	48
16	16	16	48	48	48

ISNR= 30.1536 dB

More Workspace for Problem 3...

$$K = \begin{bmatrix} 16 & 16 & 16 & 48 & 48 & 48 \\ 16 & 16 & 16 & 48 & 48 & 48 \\ 16 & 16 & 16 & 48 & 48 & 48 \\ 16 & 16 & 16 & 48 & 48 & 48 \\ 16 & 16 & 16 & 48 & 48 & 48 \\ 16 & 16 & 16 & 48 & 48 & 48 \end{bmatrix}$$

$$I-J = \begin{bmatrix} 0 & 16 & 0 & 0 & 48 & 0 \\ 0 & 0 & -47 & 0 & 0 & 0 \\ -47 & 0 & 0 & 0 & 48 & 0 \\ 0 & 0 & 0 & -13 & 0 & -15 \\ 0 & 16 & 0 & 50 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$I-K = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$(I-J)^2 = \begin{bmatrix} 0 & 256 & 0 & 0 & 2304 & 0 \\ 0 & 0 & 2209 & 0 & 0 & 0 \\ 2209 & 0 & 0 & 0 & 2304 & 0 \\ 0 & 0 & 0 & 169 & 0 & 225 \\ 0 & 256 & 0 & 2500 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$(I-k)^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$MSE(k) = \frac{4+4+4}{36} = \frac{12}{36} = \frac{1}{3}$$

$$MSE(J) = (256 + 2304 + 2209 + 2209 + 2304 + 169 + 225 + 256 + 2500)/36$$

$$= \frac{12{,}432}{36} = 345\tfrac{1}{3}$$

$$ISNR = 10 \log_{10}\left(\frac{MSE(J)}{MSE(k)}\right) = 10 \log_{10}\left[\frac{12{,}432}{36} \cdot \frac{36}{12}\right]$$

$$= 10 \log_{10}\left[\frac{12{,}432}{12}\right] = 10 \log_{10} 1{,}036 = \underline{\underline{30.1536 \text{ dB}}}$$

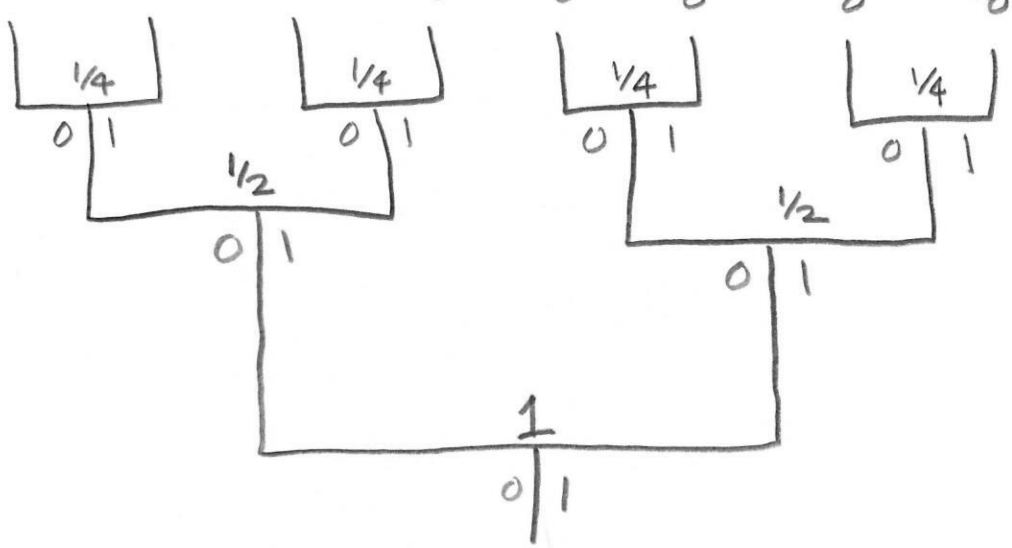
4. **20 pts.** Gray scale digital images **I** with 3 bits per pixel and gray levels in the range $\{0, 1, \ldots, 7\}$ are modeled as coming from an information source with the following source symbol probabilities (normalized histogram):

Note; flat histogram means max entropy... there won't be any compression!!

k	0	1	2	3	4	5	6	7
$p_{\mathbf{I}}(k)$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$

(a) **12 pts**. Design a Huffman code to encode these images.

K	0	1	2	3	4	5	6	7
$p_I(K)$	1/8	1/8	1/8	1/8	1/8	1/8	1/8	1/8



K	0	1	2	3	4	5	6	7
C(K)	000	001	010	011	100	101	110	111
L[c(K)]	3	3	3	3	3	3	3	3

Problem 4 cont...

(b) **4 pts.** Find the expected BPP (bits per pixel) and CR (compression ratio) for the coded images $C(\mathbf{I})$.

$$BPP = \underbrace{3\cdot\tfrac{1}{8}}_{0} + \underbrace{3\cdot\tfrac{1}{8}}_{1} + \underbrace{3\cdot\tfrac{1}{8}}_{2} + \underbrace{3\cdot\tfrac{1}{8}}_{3} + \underbrace{3\cdot\tfrac{1}{8}}_{4} + \underbrace{3\cdot\tfrac{1}{8}}_{5} + \underbrace{3\cdot\tfrac{1}{8}}_{6} + \underbrace{3\cdot\tfrac{1}{8}}_{7} = 3 ///$$

$$CR = \frac{BPP(I)}{BPP[C(I)]} = \frac{3}{3} = 1 ///$$

$$\boxed{\begin{array}{c} BPP\ C(I) = 3 \\ CR = 1.0 \end{array}}$$

(c) **4 pts.** Does your code achieve the theoretical lower bound on BPP for the coded images? Explain why or why not.

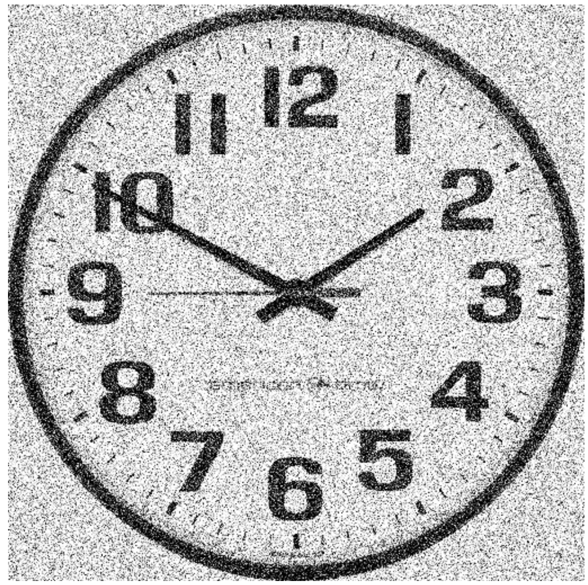
YES. All of the symbol probabilities are integer powers of 2. //

→ However, in this case the entropy is 3 so the shannon entropic lower bound is equal to the BPP of the naïve representation.

5. **20 pts.** The 512×512 image $\mathbf{I}$ shown at left below has 8-bit pixels in the range $0 \leq I(m,n) \leq 255$. This image is transmitted through a communication channel where it is corrupted by IID additive white Gaussian noise with standard deviation $\sigma_N = 100$. The received image $\mathbf{J}$ is at right below.

I

J



Three edge detectors are applied:

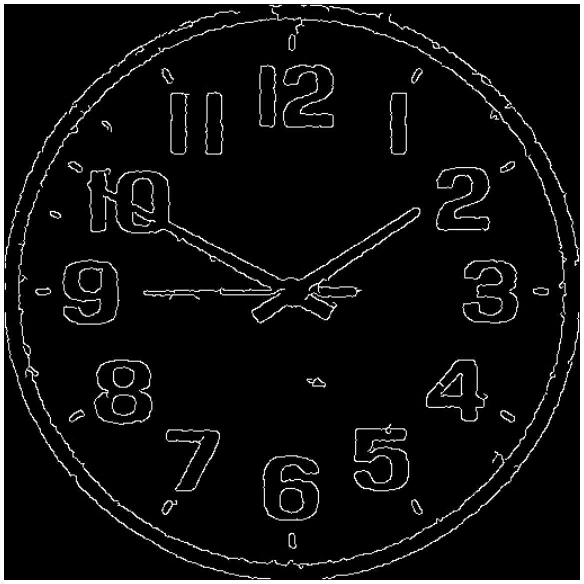
- Sobel, where the threshold shown on page 8.54 of the notes is set to 42.
- LoG with a space constant of $\sigma = 7.00$. ZC thresholding is applied with a threshold of 0.10.
- Canny with a space constant of $\sigma = 1.41$. ZC thresholding is applied with a threshold of 0.20.

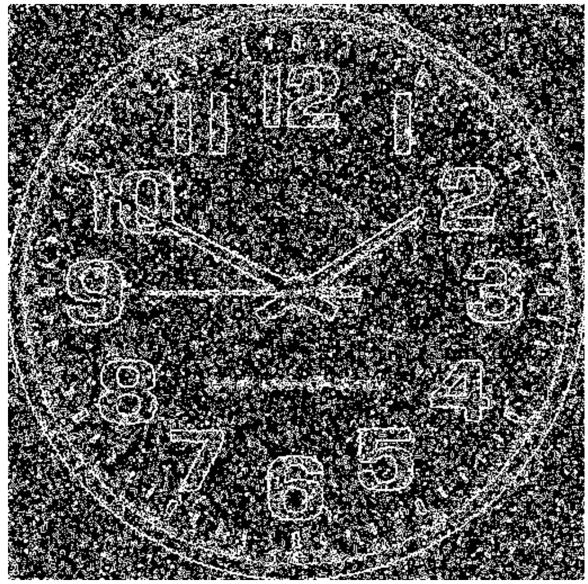
Identify and label the three resulting edge maps shown on the following page.

Problem 5 cont...

Identify and label the three resulting edge maps shown below:

Canny



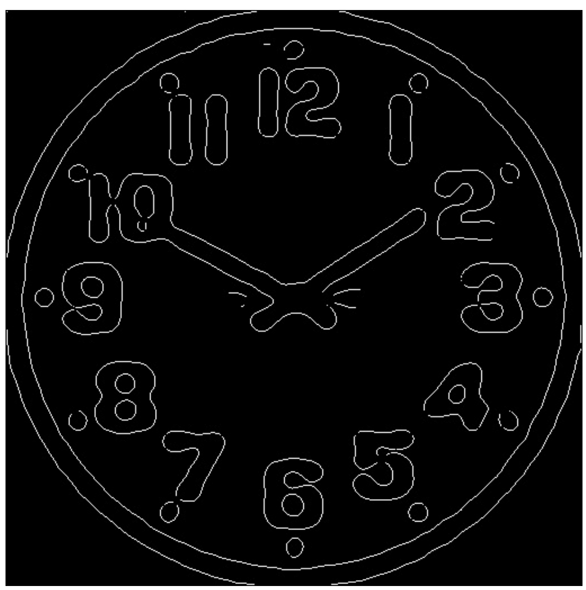


Sobel

- Noise reducing
- Detects medium & large scale edges
- Less rounding of corners
- Some broken edges due to thresholding

- Detects all edges
- High sensitivity to noise

LoG



- Noise reducing
- Detects only larger scale edges
- More rounding of corners
- Some broken edges due to thresholding