**Stevens Institute of Technology**

**Department of Electrical and Computer Engineering**

**CpE 462 Introduction to Image Processing and Coding**

**Spring Semester 2022 Final Exam, May 12, 6:00 – 10:00 PM**

**Instructions:**

* Please provide necessary **intermediate steps** in your work. You will get zero credit if you only provide the final result without necessary steps.

* **All** calculations are to be done by **hand**, with the help of a calculator. Computer is only allowed for viewing lecture notes and course materials.

* Sign the following statement

**I pledge on my honor that I have abided by the Stevens honor code**



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



**Name (print): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**



**Last four digits of your Student ID: \_\_\_\_\_\_\_\_\_\_\_**



**Problem 1: (20 points)** A slightly distorted 256 256 LENA image is shown below.

1. Sketch the histogram of this distorted image
2. Explain whetherthe nth power or the nth root function can enhance this image.
3. Sketch a contrast stretching function that can enhance this image

original image distorted image



1. **The image is brighter, so the lower amplitudes should be mapped. Thus, the nth power function should be used to enhance visibility**

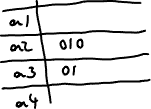
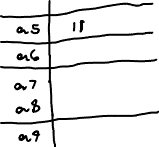


**Problem 2 (20 points)** Given an alphabet A={a1, a2, a3, a4, a5, a6, a7, a8, a9} with probabilities P(a1)=0.04, P(a2)=0.07, P(a3)=0.12, P(a4)=0.21, P(a5)=0.20, P(a6)=0.15, P(a7)=0.10, P(a8)=0.09, P(a9)=0.02.

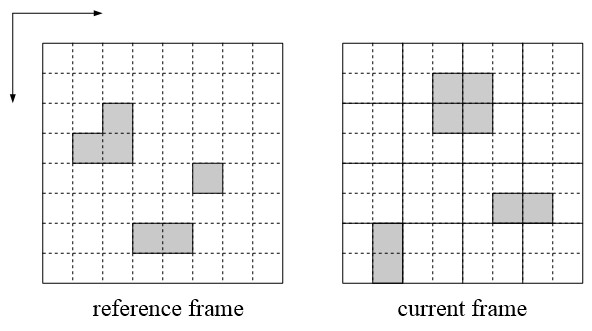
1. Compute the entropy of this data source (**in Log Base 2**);



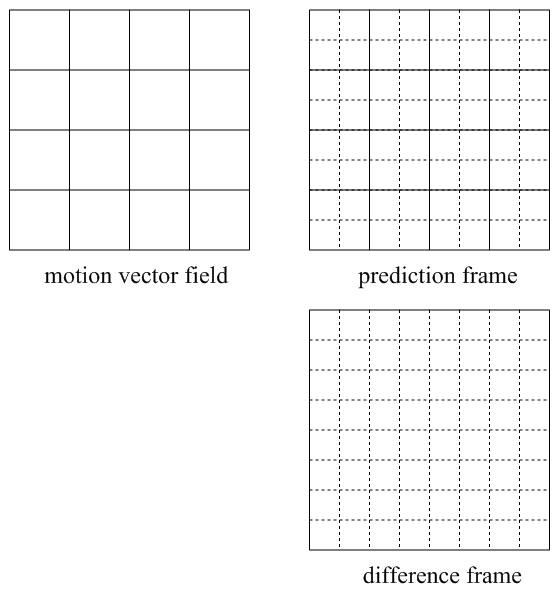
1. Design a Huffman code for this data source. **(Show your steps)**



**Problem 3 (20 points)** Based on the motion compensated prediction used in MPEG, generate the prediction frame, the motion vectors, and the difference frame for the current frame as shown. Assume each box represents a pixel, each macro-block is of 22 pixels, the white boxes have value of zero (**0**), and the gray boxes have value of one (**1**). (Note: motion vectors should point from the location of the best match block in the reference frame to the location of the current macro-block in the current frame.)







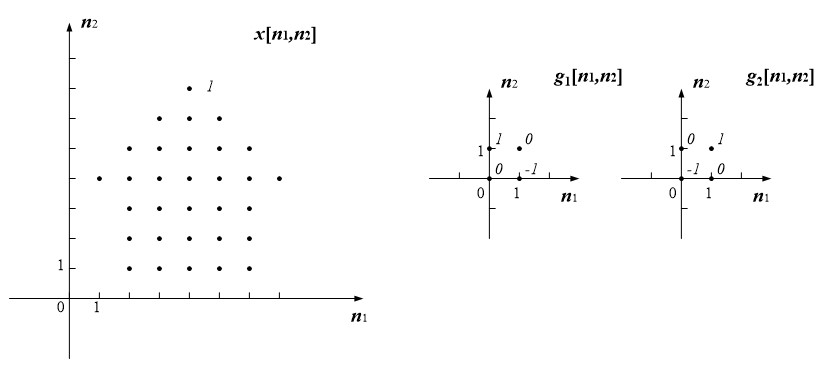


**Problem 4 (20 points)** Given a 2-D signal ***x*[*n*1, *n*2]** and the Roberts edge detector ***g*1[*n*1, *n*2]** and ***g*2[*n*1, *n*2]** as shown, all dark pixels shown in ***x*[*n*1, *n*2]** have value of **1.**

**1)** Calculate the 2-D convolution of ***y*1[*n*1, *n*2] = *x*[*n*1, *n*2]** **\*\* *g*1[*n*1, *n*2],**

**2)** Calculate the 2-D convolution of ***y*2[*n*1, *n*2] = *x*[*n*1, *n*2]** **\*\* *g*2[*n*1, *n*2],**

**3)** Sum up the absolution values of these two outputs as ***y*[*n*1, *n*2] =** **|*y*1[*n*1, *n*2]| + |*y*1[*n*1, *n*2]|,** which shouldbecome the detected edge image. **(Important: show your steps in all calculations.)**



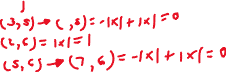






Chart

Description automatically generated







Chart

Description automatically generated Diagram

Description automatically generated



Chart

Description automatically generated



**Problem 5 (20 points)** Write a segment of C/C++ routine to calculate the 22 2D DCT of an input image. The 22 2D DCT is calculated on every non-overlapping 22 image block inside the image. Assume you are using the “**imageproc.cpp**” code structure, and your work should be placed between the comments “**image processing begins**” and “**image processing ends**”. Assume the input image is stored in the 2-D array **image\_in[][]**, the output image should be saved in the 2-D array **image\_out[][]**.

**Hint:** 2D DCT transform coefficients can be pre-calculated, referring to Homework 4.

**Notes:**

1. Try to write your code as efficient as possible, which will be graded accordingly;
2. If you can not complete this code in C/C++, at least you may provide a pseudocode to outline your procedure, and receive some partial credits.

Text

Description automatically generated