## CSE463 Lab Assignment 2

Deadline: March 8, 2025, 11:59PM

**Submission instructions:**

You will require a total of 7 different images in this assignment. The numbers are mentioned in square brackets.

* Open a folder in Google Drive and name it as- ID\_Name\_Lab2
  1. Upload the images you used (number them the way [..] is numbered, eg. Image 1, Image 2, etc.)
  2. Upload a PDF consisting of the codes & screenshots of the outputs of each code [Optional]
  3. Ensure that you submit an ipynb file containing all the outputs of your results.
* Share the link of the folder in the submission form. (Make sure the folder is accessible(**Anyone with the link**)
* Submission Link: <https://forms.gle/mq7xB2XUoT6iwp5b8>

#### 

#### **Step 1: Exploring Basic Convolution and Custom Kernels**

1. **Apply Convolution with a Simple Kernel**
   * **Task:** Given a **grayscale** image[1], apply a basic 3x3 **identity kernel** to the image using a 2D convolution function (e.g., cv2.filter2D). Observe and describe the output in 1-2 sentences.
2. **Custom Kernel Design**
   * **Task:** Design a 3x3 custom kernel to create a **sharpening** effect on an image[2]. Apply the kernel to the image and describe how this kernel affects the overall clarity and contrast of the image. [Opposite of blurring]

#### **Step 2: Understanding Padding and Its Effects on Convolution**

1. **Experiment with Different Padding Techniques**
   * **Task:** Apply the sharpening kernel created in Step 1 to an image[3] with three different padding types: **constant (zero-padding)**, **reflect padding**, and **same padding**.

#### **Step 3: Filtering for Noise Reduction and Smoothing**

**4. Adding Noise and Applying an Average Filter**

* + **Task:** Add **Gaussian noise** to an image[4]. Apply a 5x5 average filter using cv2.blur and observe the changes in noise level and overall appearance of the image. Explain in 1-2 sentences.

**5. Gaussian Blur for Smoothing**

* + **Task:** Apply a Gaussian blur to the noisy image created in Part 4 using a 5x5 Gaussian kernel (cv2.GaussianBlur). Experiment with different standard deviations (sigma values) and observe how the level of smoothing changes. Explain in 1-2 sentences.

#### 

#### **Step 4: Edge Detection and Gradient Calculation**

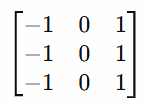
**6. Applying Laplacian Filter for Edge Detection**

**Task**: Use the Laplacian filter to detect edges in a clear image[5]. Apply the Laplacian filter on the grayscale image, then visualize the result.

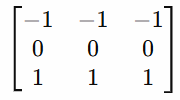
**7. Estimating Horizontal and Vertical Gradients**

**Task**: To estimate the edges in horizontal and vertical directions, apply two simple gradient kernels:

* For vertical edges, use:



* For horizontal edges, use:



Apply each kernel separately to an image[6] and visualize the results to observe the types of edges each captures. Explain in 1-2 sentences. (Just one more 😐)

#### **Step 5: Image Enhancement through Histogram Equalization**

**8. Histogram Equalization for Contrast Enhancement**

* + **Task 1:** Load a low-contrast grayscale image[7] and apply histogram equalization. Observe and describe in 1-2 sentences how the contrast improves.
  + **Task 2:** Apply histogram equalization three times on the original image[7] and describe any diminishing effects or artifacts that appear. Explain in 1-2 sentences (One last time 🙂)