Assignment - Week 2

June 21, 2025

Dear Learners,
You've done a fantastic job navigating through Week 1, and now it's time to take it a step further!.
Wishing you all the very best for Week 2 — you've got this!

To submit the Week 2 assignment, make a separate folder in the GitHub repository that you created, and store the assignment solution in that folder.

Problem 1

Read an image and apply a custom convolution kernel to sharpen the image.

1. Define a 3×3 sharpening kernel using **NumPy** and apply it using **cv2.filter2D**.
   Use the given Laplacian kernel as a sharpening filter:

0	-1	0
-1	5	-1
0	-1	0

2. Display the original and filtered images side by side using np.hstack()
3. Similarly, use 2 **Sobel filters of size 3x3 kernel** for horizontal and vertical edge detection and display them in 2 different images.
   Click here to Know more about Sobel filters.

Problem 2

Work with histogram equalization to enhance image contrast.

1. Load a grayscale image and display its histogram using **matplotlib**.
2. Apply histogram equalization using cv2.equalizeHist() and display the result.
3. Plot the histogram of the equalized image and compare it to the original.

Problem 3

Create a basic image processing pipeline using OpenCV.

1. Load a color image and convert it to grayscale.
2. Apply a Gaussian blur (kernel size 5×5) to reduce noise.
3. Run Canny edge detection and display all three: original, blurred, and edge image side by side using matplotlib

Problem 4

1. Read an image and convert it to grayscale format. Count how many pixels have a brightness greater than a certain threshold (>200). Highlight these bright pixels by drawing white dots at their positions on the image.
2. Read an image in colored format and
   (a) Given a list of coordinates like [(50, 60, 150, 120), (200, 180, 300, 250)] where each tuple is (x1, y1, x2, y2), draw bounding boxes on an image.
   (b) Label each box with a name like "Object 1", "Object 2", etc.
   (c) Display the final image with all annotations.
3. Create a blank image of size 500x500 pixels and perform the following operations:
   (a) draw a red rectangle (Without infill), a green circle (with infill), and a blue diagonal line on it.
   (b) Add your name as a watermark in the bottom-right corner.
   (c) Save the image and display it.

Problem 5

1. Read a colored image and convert it to **HSV Color Channel** image.
   (a) Split the HSV image into its three channels: Hue, Saturation, Value.
   (b) Display all three channels separately, i.e, 3 separate images displaying 1 channel at a time.
2. Load a color image and convert it from BGR to HSV. Convert the HSV image back to BGR and display all three images: Original BGR, HSV, and Reconverted BGR. (Observe whether the reconverted BGR image looks similar to the original image or not.)