DCIT 412 Lab Assignment

10989853

**Task 1**

import cv2

from google.colab.patches import cv2\_imshow

**#Load the image**

image = cv2.imread('cars.jpeg')

**# Check if the image loaded successfully**

if image is None:

    print("Error: Could not load image 'cars.jpeg'. Make sure the file exists in the current directory.")

else:

**#Convert to grayscale**

    gray\_image = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

**#Display original and grayscale images**

    cv2\_imshow(image)

    cv2\_imshow(gray\_image)

**# Save the grayscale image**

    cv2.imwrite('photo\_gray.jpg', gray\_image)

    print("Grayscale image saved as 'photo\_gray.jpg'.")

    # Wait until a key is pressed and close the image windows

    # cv2.waitKey(0) # waitKey is not needed with cv2\_imshow

    # cv2.destroyAllWindows() # destroyAllWindows is not needed with cv2\_imshow

**Task 2**

import cv2

import matplotlib.pyplot as plt

from google.colab.patches import cv2\_imshow

image = cv2.imread('nice house.jpeg')

**#Check if the image loaded successfully**

if image is None:

    print("Error: Could not load image 'nice house.jpeg'. Make sure it's in the same directory.")

    exit()

**#Convert to Grayscale, HSV, and LAB**

gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

hsv = cv2.cvtColor(image, cv2.COLOR\_BGR2HSV)

lab = cv2.cvtColor(image, cv2.COLOR\_BGR2LAB)

**#Display all converted images**

cv2\_imshow(image)

cv2\_imshow(gray)

cv2\_imshow(hsv)

cv2\_imshow(lab)

**#Save converted images**

cv2.imwrite('photo\_grayscale.jpg', gray)

cv2.imwrite('photo\_hsv.jpg', hsv)

cv2.imwrite('photo\_lab.jpg', lab)

print("Images saved: photo\_grayscale.jpg, photo\_hsv.jpg, photo\_lab.jpg")

**#Plot histogram of grayscale image**

plt.figure(figsize=(8, 4))

plt.title('Grayscale Histogram')

plt.xlabel('Pixel Intensity')

plt.ylabel('Frequency')

plt.hist(gray.ravel(), bins=256, range=[0, 256], color='gray')

plt.grid(True)

plt.tight\_layout()

plt.show()

**#Wait for key press and close windows**

# cv2.waitKey(0) # waitKey is not needed when using cv2\_imshow

# cv2.destroyAllWindows() # destroyAllWindows is not needed when using cv2\_imshow

**Task 3: Binary Thresholding Question**

A pixel in a grayscale image has a value of 180. If the threshold for binary thresholding is set to 150, what is the new pixel value after applying binary thresholding? (Assume output values are 0 for pixels below the threshold and 255 for pixels above the threshold.)

**Answer**

The rule in binary thresholding says that

if the pixel > or = the threshold, then set it to 255

And if the pixel value < threshold, set it to 0

Given that, the original pixel value is 180 and the threshold is 150

Since 180 > or = 150, then the new pixel value = **255**