ECE 5554 SU22 – Dr. Jones – HW1

**Part 2:**

Animals img size: (768, 1024, 3)

Stonehenge img size: (768, 1024, 3)

**Part 3:**

Animals red avg: 92.87458928426106

Animals green avg: 100.50979868570964

Animals blue avg: 128.48477045694986

Stonehenge red avg: 92.87458928426106

Stonehenge green avg: 100.50979868570964

Stonehenge blue avg: 128.48477045694986

**Part 5:**

****

**Part 6:**

****

**Part 7:**

****

Code

Homework1.py:

import cv2

import numpy as np

#%%

# Part 1

animals = cv2.imread("animals.png")

# Part 2

height = len(animals)

width = len(animals[0])

depth = len(animals[0, 0])

print(f"Animals img size: {animals.shape}")

# Part 3

animals\_red = animals[:,:,2]

animals\_green = animals[:,:,1]

animals\_blue = animals[:,:,0]

print(f"Animals red avg: {np.mean(animals\_red)}")

print(f"Animals green avg: {np.mean(animals\_green)}")

print(f"Animals blue avg: {np.mean(animals\_blue)}")

#%%

# Part 1

stonehenge = cv2.imread("Stonehenge\_1024x768.png")

# Part 2

height = len(stonehenge)

width = len(stonehenge[0])

depth = len(stonehenge[0, 0])

print(f"Stonehenge img size: {stonehenge.shape}")

# Part 3

stonehenge\_red = stonehenge[:,:,2]

stonehenge\_green = stonehenge[:,:,1]

stonehenge\_blue = stonehenge[:,:,0]

print(f"Stonehenge red avg: {np.mean(animals\_red)}")

print(f"Stonehenge green avg: {np.mean(animals\_green)}")

print(f"Stonehenge blue avg: {np.mean(animals\_blue)}")

#%% Pixel by Pixel Avg

def avg\_pixel\_by\_pixel(img1, img2):

# Assuming both images are the same size

height = len(img1)

width = len(img1[0])

new\_image = np.zeros((height, width), dtype = "uint8")

for y\_pix in range(height):

for x\_pix in range(width):

img1\_value = img1[y\_pix][x\_pix]

img2\_value = img2[y\_pix][x\_pix]

avg\_value = np.mean([img1\_value, img2\_value])

new\_image[y\_pix, x\_pix] = avg\_value

return new\_image

# Part 4

animals\_grayscale = cv2.cvtColor(animals, cv2.COLOR\_BGR2GRAY)

stonehenge\_grayscale = cv2.cvtColor(stonehenge, cv2.COLOR\_BGR2GRAY)

# Part 5

avg\_image = avg\_pixel\_by\_pixel(animals\_grayscale, stonehenge\_grayscale)

# show images

cv2.namedWindow('INPUT', flags=cv2.WINDOW\_NORMAL)

cv2.imshow('INPUT',avg\_image)

cv2.waitKey(0)

cv2.destroyAllWindows()

cv2.imwrite("part5\_img.png", avg\_image)

#%% Pixel by Pixel Max

def max\_pixel\_by\_pixel(img1, img2):

# Assuming both images are the same size

height = len(img1)

width = len(img1[0])

new\_image = np.zeros((height, width), dtype = "uint8")

for y\_pix in range(height):

for x\_pix in range(width):

img1\_value = img1[y\_pix][x\_pix]

img2\_value = img2[y\_pix][x\_pix]

max\_value = np.max([img1\_value, img2\_value])

new\_image[y\_pix, x\_pix] = max\_value

return new\_image

# Part 6

max\_image = max\_pixel\_by\_pixel(animals\_grayscale, stonehenge\_grayscale)

# show images

cv2.namedWindow('INPUT', flags=cv2.WINDOW\_NORMAL)

cv2.imshow('INPUT',max\_image)

cv2.waitKey(0)

cv2.destroyAllWindows()

cv2.imwrite("part6\_img.png", max\_image)

#%% Pixel by Pixel Difference

def diff\_pixel\_by\_pixel(img1, img2):

# Assuming both images are the same size

height = len(img1)

width = len(img1[0])

new\_image = np.zeros((height, width), dtype = "uint8")

for y\_pix in range(height):

for x\_pix in range(width):

img1\_value = img1[y\_pix][x\_pix]

img2\_value = img2[y\_pix][x\_pix]

diff\_value = cv2.absdiff(np.array(img1\_value), np.array(img2\_value))

new\_image[y\_pix, x\_pix] = diff\_value

return new\_image

# Part 7

diff\_image = diff\_pixel\_by\_pixel(animals\_grayscale, stonehenge\_grayscale)

# show images

cv2.namedWindow('INPUT', flags=cv2.WINDOW\_NORMAL)

cv2.imshow('INPUT', diff\_image)

cv2.waitKey(0)

cv2.destroyAllWindows()

cv2.imwrite("part7\_img.png", diff\_image)