Part 3

import cv2

import numpy as np

import numpy.linalg as la

import math

def determine\_values(current\_h\_val, c\_val, x\_val):

r\_val = g\_val = b\_val = 0.0

if current\_h\_val >= 0 and current\_h\_val < 60:

r\_val = c\_val

g\_val = x\_val

elif current\_h\_val >= 60 and current\_h\_val < 120:

r\_val = x\_val

g\_val = c\_val

elif current\_h\_val >= 120 and current\_h\_val < 180:

g\_val = c\_val

b\_val = x\_val

elif current\_h\_val >= 180 and current\_h\_val < 240:

g\_val = x\_val

b\_val = c\_val

elif current\_h\_val >= 240 and current\_h\_val < 300:

r\_val = x\_val

b\_val = c\_val

elif current\_h\_val >= 300 and current\_h\_val < 360:

r\_val = c\_val

b\_val = x\_val

return r\_val, g\_val, b\_val

brightness\_file\_name = raw\_input('Brightness file to be histogram equalized: ')

brightness\_image = cv2.imread(brightness\_file\_name)

rows\_for\_image = len(brightness\_image[:,0])

columns\_for\_image = len(brightness\_image[0,:])

total\_pixels = rows\_for\_image\*columns\_for\_image

histogram\_equalized\_array = np.zeros(256)

for i in range(rows\_for\_image):

for j in range(columns\_for\_image):

v\_value = brightness\_image[i][j][0]

histogram\_equalized\_array[int(v\_value)] += 1

cumulative\_sum = 0.0

for k in range(len(histogram\_equalized\_array)):

current\_pmf = float(histogram\_equalized\_array[k])/float(total\_pixels)

cumulative\_sum = cumulative\_sum + current\_pmf

new\_value = cumulative\_sum\*255

histogram\_equalized\_array[k] = math.floor(new\_value)

new\_brightness\_image = np.zeros([rows\_for\_image, columns\_for\_image])

for l in range(rows\_for\_image):

for m in range(columns\_for\_image):

v\_value = brightness\_image[l][m][0]

new\_brightness\_image[l][m] = histogram\_equalized\_array[int(v\_value)]

if 'concert' in brightness\_file\_name:

hue\_file\_name = 'concert\_hue.jpg'

saturation\_file\_name = 'concert\_saturation.jpg'

elif 'sea1' in brightness\_file\_name:

hue\_file\_name = 'sea1\_hue.jpg'

saturation\_file\_name = 'sea1\_saturation.jpg'

elif 'sea2' in brightness\_file\_name:

hue\_file\_name = 'sea2\_hue.jpg'

saturation\_file\_name = 'sea2\_saturation.jpg'

hue\_image = cv2.imread(hue\_file\_name)

saturation\_image = cv2.imread(saturation\_file\_name)

rgb\_array = np.zeros([rows\_for\_image, columns\_for\_image, 3])

for i in range(rows\_for\_image):

for j in range(columns\_for\_image):

current\_h\_val = hue\_image[i][j][0]

current\_s\_val = saturation\_image[i][j][0]

current\_v\_val = new\_brightness\_image[i][j]

current\_h\_val = (current\_h\_val/255.0)\*360.0

current\_s\_val = current\_s\_val/255.0

current\_v\_val = current\_v\_val/255.0

c\_val = current\_s\_val\*current\_v\_val

x\_val = c\_val\*(1 - abs(((current\_h\_val/60)%2)-1))

m\_val = current\_v\_val - c\_val

r\_val, g\_val, b\_val = determine\_values(current\_h\_val, c\_val, x\_val)

red = (r\_val + m\_val)\*255

green = (g\_val + m\_val)\*255

blue = (b\_val + m\_val)\*255

rgb\_array[i][j][0] = red

rgb\_array[i][j][1] = green

rgb\_array[i][j][2] = blue

if 'concert' in brightness\_file\_name:

cv2.imwrite('concert\_histeq.jpg', rgb\_array)

elif 'sea1' in brightness\_file\_name:

cv2.imwrite('sea1\_histeq.jpg', rgb\_array)

elif 'sea2' in brightness\_file\_name:

cv2.imwrite('sea2\_histeq.jpg', rgb\_array)