Assignment5

Color Histogram Equalization

2017113547 이정근



#Color Histogram Equalization.py

#2017113547 이정근

import cv2

import numpy as np

NUM\_INTENSITY = 256

def HistogramEqualize(img):

height, width = img.shape

num\_pixels = height\*width

# making histogram

histogram = np.zeros((NUM\_INTENSITY,))

for y in range(height):

for x in range(width):

histogram[img[y, x]] = histogram[img[y, x]] + 1

# normalizing histogram

normalized\_histogram = np.divide(histogram, num\_pixels)

# making cdf

cdf = np.zeros((NUM\_INTENSITY,))

cdf[0] = normalized\_histogram[0]

for k in range(1, NUM\_INTENSITY, 1):

cdf[k] = cdf[k-1] + normalized\_histogram[k]

# finding output gray level

output\_gray\_level = np.multiply(NUM\_INTENSITY-1, cdf)

output\_gray\_level = np.round(output\_gray\_level)

# convert image to equalized image

result = np.zeros((height, width), np.uint8) # result image

for y in range(height):

for x in range(width):

result[y, x] = output\_gray\_level[img[y, x]]

return result

img = cv2.imread('dgu\_night\_color.png', cv2.IMREAD\_COLOR) # img2numpy

imgYCC = cv2.cvtColor(img, cv2.COLOR\_BGR2YCR\_CB)

imgYCC\_out = np.zeros(imgYCC.shape, np.uint8)

imgYCC\_out[:,:,0] = HistogramEqualize(imgYCC[:,:,0])

row, col = imgYCC\_out[:,:,0].shape

# divide by zero error preventing

for i in range(row):

for j in range(col):

if imgYCC[i, j, 0] == 0:

imgYCC[i, j, 0] = 1

s = 0.095

img\_out = np.zeros(img.shape, np.uint8)

img\_out[:,:,0] = np.multiply(imgYCC\_out[:,:,0],np.divide(img[:,:,0],imgYCC[:,:,0])\*\*s)

img\_out[:,:,1] = np.multiply(imgYCC\_out[:,:,0],np.divide(img[:,:,1],imgYCC[:,:,0])\*\*s)

img\_out[:,:,2] = np.multiply(imgYCC\_out[:,:,0],np.divide(img[:,:,2],imgYCC[:,:,0])\*\*s)

cv2.imshow("Color Histogram Equalized Image", img\_out)

cv2.waitKey()