Manara university

Engineering college

Faculty of Informatics engineering

Computer vision

2025 - 2026

Lab session 3:
log function and histogram of the image

Tasks:

1. Read images P1 or P2, in the gray scale and store it as (img) array.
2. Apply Logarithmic function to the (img) and store the result in (log_img) array using the two formulas:

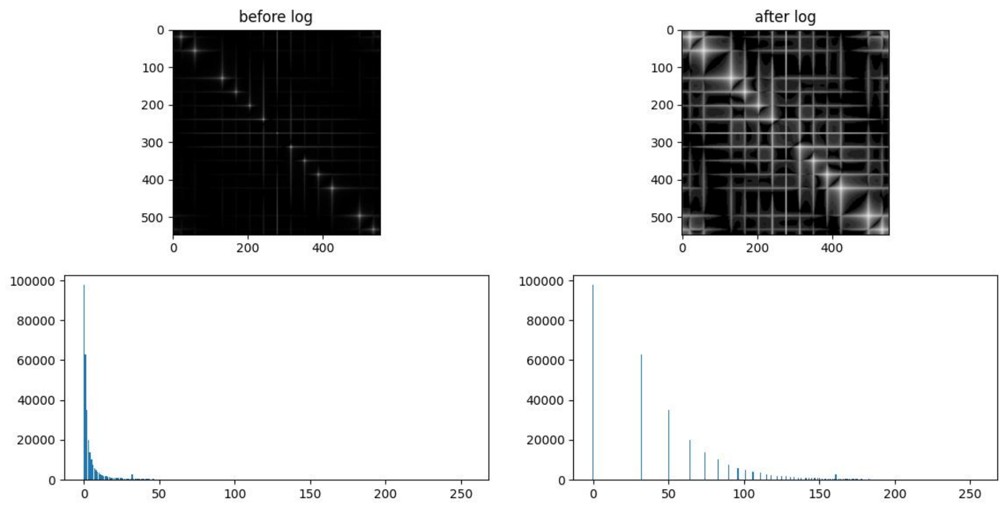
log_image[i, j] = c * (np.log(img[i, j] + 1)),

where: c = 255 / np.log(np.max(img)+1)

3. Calculate the histogram of (img) and store it in a [256, 1] vector called (h1).
4. Calculate the histogram of (log_img) and store it in a [256, 1] vector called (h2).
5. Display (img) and (log_img) with corresponded histograms using (plt.subplots)

Needed syntaxes:

```
# casting log_image array into uint8 one:

log_image = np.array(log_image, dtype = np.uint8)


# creation of multiple sub-figures and sub-plots to display the images and histograms:

fig, axis = plt.subplots(2, 2)

axis[0][0].set_title('before log')

axis[0][0].imshow(img, cmap='gray')

…

axis[1][1].bar(idx[0], h2[0])

plt.show()
```



Code:

```
import cv2

import numpy as np

import matplotlib.pyplot as plt


# Read an image

path = r" \\"

img = cv2.imread(path+"P2.png", 0)


# Apply log transformation method

c = 255/ np.log(1 + np.max(img))

log_image = c * (np.log(img + 1))


# Specify the data type so that

# float value will be converted to int

log_image = np.array(log_image, dtype = np.uint8)


r, c = img.shape
```

```
h1 = np.zeros((1, 256), dtype = int)

h2 = np.zeros((1, 256), dtype = int)

idx = np.zeros((1, 256), dtype = int)


for i in range(r):

    for j in range(c):

        h1[0][img[i, j]] = h1[0][img[i, j]]+1

        h2[0][log_image[i, j]] = h2[0][log_image[i, j]]+1


for i in range(256):

  idx[0][i] = i

# Display both images

fig, axis = plt.subplots(2, 2)

axis[0][0].set_title('before log')

axis[0][0].imshow(img, cmap='gray')

axis[0][1].set_title('after log')

axis[0][1].imshow(log_image, cmap='gray')

axis[1][0].bar(idx[0], h1[0])

axis[1][1].bar(idx[0], h2[0])

plt.show()
```