Big Image Data Processing

with Hand-on Projects

Day-1-Lab Part (03-11-2022)

Practice Set (using Python)	
S.No.	Question
1	Take a color image/photo/picture: i. Convert color image to gray image (without using inbuilt function) and display them both using subplot. ii. Convert the result of (i) into binary image (without using inbuilt function) and display all three images (Color, Gray, Binary) using subplot.
2	***Image Negatives***: Figure 3.4 imageprocessinagplace.com
3	***Histogram Equalization***: Figure 3.20 imageprocessinagplace.com

Projects (using Python)	
1	Content Based Image Retrieval using Texture Features (Python)
2	Content Based Image Retrieval using Color Features (Python)

1. Content Based Image Retrieval using Texture Features (Python)

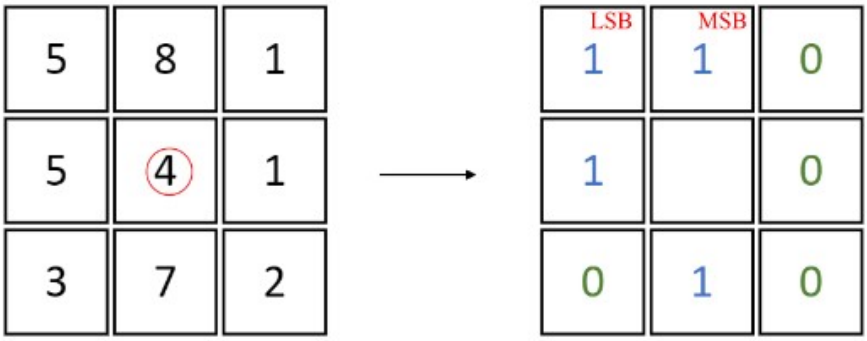
Description about Local Binary Pattern (LBP) Feature of an image

The basic idea of Local Binary Pattern (LBP) is to take the gray value of the central pixel as the threshold and compare the gray value of the pixels in its neighborhood with the threshold value, to obtain the binary code to express the local texture features. Algorithm for same is given below-:

Step-1: Perform windowing operation on image by taking 3×3 window for entire image without any padding.

Step-2: For computing the result of a window, take the gray value of the central pixel as the threshold value, compare the eight gray values of its neighborhood with the threshold value, and the pixel greater than the central gray value is represented by 1, otherwise, it is represented by 0.

Step-3: Then read out 8 binary values according to the clockwise direction and convert it to decimal. (Start value in matrix is the least significant bit).



Binary number is 10001011
Decimal equivalent is 139

Step-4: After getting *(n-2)×(n-2)* matrix values for the *n×n* image represent it in vector form representing histogram. The vector is of size 256 where each entry represents the number of pixels for that intensity value.

Step-5: This final vector represents the *feature vector* of that image.

Task A: You are given a dataset consisting of 250 Gray level images. The task is to find the feature vector for all those images. Then given a query image, find all the top 5 images which are closest to the given query image. The closeness is defined by calculating *the Euclidean distance between the feature vectors* of size 256. The one with least Euclidean distance is considered closest to the given query image.

Task B: Convert all 250 Gray level images to their corresponding Image Negatives. The results obtained in Task A and Task B on a particular query image should match. Accordingly, you need to find out the procedure for calculating the binary pattern for the negative images so as to get the same result as in Task-A.

The 250-image data set is shared to you via Google Drive.

Example Output screenshot:

Query image and the next Five are the retrieved images

2. Content Based Image Retrieval using Color Features (Python)

Color features for CBIR

Step-1: Any color image will generally have 3 channels, namely, Red, Green, and Blue. We are supposed to get the Pixel values of the individual channels and add this frequency of pixels into their respective feature vector (of length 8 here) as shown below.

2	3	4	4
6	7	1	2
3	4	0	1
1	2	3	4

1	2	3	6
0	1	4	5
2	1	6	7
1	2	4	5

1	6	7	2
1	2	4	5
0	1	0	6
7	1	2	4

RED GREEN BLUE

	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
	1	3	3	3	4	0	1	1	1	4	3	1	2	2	2	1	2	4	3	0	2	1	2	2

Step-2: For our color image the vector is of size [3 X 256] where each entry represents the number of pixels for that channel and the respective intensity value.

Step-3: This final vector of size [3 X 256] represents the *feature vector* of that image.

Task : You are given a dataset consisting of 250 Color images. The task is to find the feature vector for all those images. Then given a query image, find all the top 5 images which are closest to the given query image. The closeness is defined by calculating *the Euclidean distance between the feature vectors* of size [3 X 256]. The one with least Euclidean distance is considered closest to the given query image.

Example Output screenshot:

First Image is the Query image and the next Five are the retrieved images with their respective Euclidean distances.