Khulna University Of Engineering & Technology

KUET

SESSIONAL REPORT

Department Of CSE Course No. CSE-4128

Experiment No. 03

Name of the Experiment Introduction to histogram equalization and its implementation in Python.

Remarks

Name Md Abu Saeed

Roll No 1907057

Group No. B2

Year 4th

Semester 1st

Date of Performance 06/03/2024

Date of Submission 20/03/2024

Objectives:

(i) To learn about histogram.

(ii) To know about histogram equalization and histogram matching.

(iii) To calculate pdf, cdf of an image.

(iv) To implement histogram equalization in python.

Introduction: The histogram of an image consists of intensity on x-axis and corresponding frequency count on y-axis, denoted by

$$h(r_k) = n_k$$

n_k → frequency count of intensity r_k

r_k → intensity level

Histogram equalization is used to ~~match~~ enhance contrast of an image.

The process of calculating histogram equilization are as follows:

(i) Calculate the pdf of all the intensity level using

$$p(r_k) = \frac{n_k}{MN}, \quad k = 0, 1, 2 \ldots L-1$$

MN = total pixels

(ii) Calculate cdf of those intensity level.

(iii) Multiply cdf with highest intensity to satisfy transformation function.

$$S_k = T(r_k) = (L-1) * cdf(r_k)$$

(iv) Then each intensity is mapped with new intensity to obtain the output image.

PDF (probability density function) represents the probability of occurring a number. From histogram, pdf of an intensity can be calculated by dividing the frequency count by total no. of pixels.

$$P(r_k) = \frac{n_k}{MN}$$

Histogram and pdf of an image are similar, but there is only scaling difference.

CDF (Cumulative destiny function) is the cumulative sum of pdf values. The final value is 1, which is the sum of all probability.

$$cdf(0) = pdf(0)$$
$$cdf(i) = cdf(i-1) + pdf(i)$$

Example:

r_k	n_k	$P(r_k)$	$cdf(r_k)$	$S(r_k)$
0	790	0.19	0.19	1
1	1023	0.25	0.44	3
2	850	0.21	0.65	5
3	656	0.16	0.81	6
4	329	0.08	0.89	6
5	245	0.06	0.95	7
6	122	0.03	0.98	7
7	81	0.02	1.0	7

Here, intensity of '5' will be replaced by new intensity 7.

Classwork pseudocode:

1. import necessary libraries for image processing and plotting

2. function plot(data, figure_no)
   plot data using plot function
   name window using figure_no
   Display the window

3. function show_all(image, counter):
   calculate hist for the image.
   convert the histogram into 1D array.
   calculate total number of pixels of the image.
   Display the original image.
   Calculate the pdf of the image.
   Plot histogram and pdf using plot function.
   Calculate cdf of the image from pdf.
   plot the cdf using plot function.
   creating intensity mapping by multiplying cdf with 255.
   Return the histogram, pdf, cdf and intensity mapping.

4. function start(image):
   Get histogram, pdf, cdf, mapping using show_all function.
   Create output image by mapping with new intensity.
   Get histogram, pdf, cdf, mapping using show_all function of the output image.
   Display the output image
   close all windows.

Read input image in grayscale format
Call the start function with the input image.

Discussion: Histogram equalization is used to increase image contrast. It works by redistributing the intensity distribution to span the full intensity range. It can improve the visibility and detail in an image, specially with an image with low contrast. It works by calculating the cdf and then change the intensity mapping into new intensities.

Conclusion: Histogram equalization is a valuable technique for improving contrast in an image. It is very useful for images with limited intensity range, which enhances visual quality of an image across various application. It should be used carefully as over-enhancement can cause un-natural effect on image.

Reference:

① Documents from lab (Lab_B2).
② www.geeksforgeeks.org/histogram-equalization-in-digital-image-processing.