

VASAVI COLLEGE OF ENGINEERING (Autonomous)
IBRAHIMBAGH, HYDERABAD – 500 031
Department of Computer Science & Engineering

INNOVATION IN TEACHING

Course: Image Processing
Topic: Histogram of an Image

Faculty: C. Gireesh
Semester: VI

Teaching Aid / Tool Used: MATLAB

Description of the Tool:

“MATLAB is a high-level language and interactive environment that enables you to perform computationally intensive tasks faster than with traditional programming languages such as C, C++ and Fortran.”

MATLAB is an interactive, interpreted language that is designed for fast numerical matrix calculations

Tools Usage in Teaching:

Histogram – A histogram is bar graph that shows a distribution of data. In image processing histograms are used to show how many of each pixel value are present in an image. Histograms can be very useful in determining which pixel values are important in an image. From this data you can manipulate an image to meet your specifications. Data from a histogram can aid you in contrast enhancement and thresholding. In order to create a histogram from an image, use the **imhist** function. Contrast enhancement can be performed by the **histeq** function, while thresholding can be performed by using the **graythresh** function and the **im2bw** function. See Example 5.1, for a demonstration of **imhist**, **imadjust**, **graythresh**, and **im2bw**. If you want to see the resulting histogram of a contrast enhanced image, simply perform the **imhist** operation on the image created with **histeq**.

Example:

In this example the JPEG image created in Example 4.2 was used to create a histogram of the pixel value distribution and a negative of the original image. The contrast was then enhanced and finally the image was transformed into a binary image according to a certain threshold value. Figure 3, below, contains the M-file used to perform these operation. Figure 1 contains the histogram of the image pictured in Figure 2. As you can see the histogram gives a distribution between 0 and 1. In order to find the exact pixel value, you must scale the histogram by the

number of bits representing each pixel value. In this case, this is an 8-bit image, so scale by 255. As you can see from the histogram, there is a lot of black and white in the image.

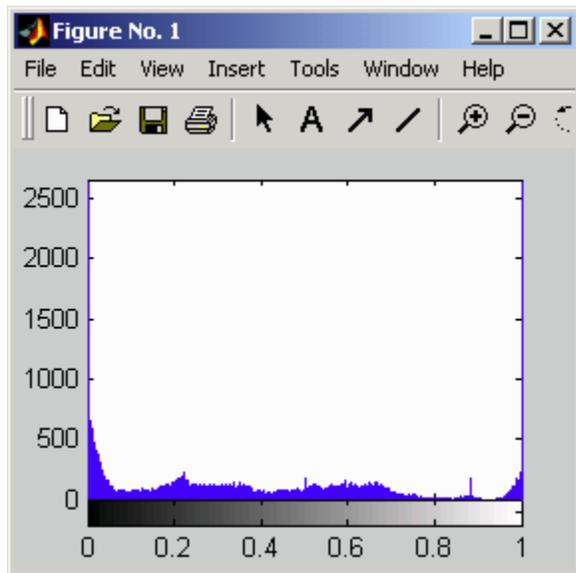


Figure:1

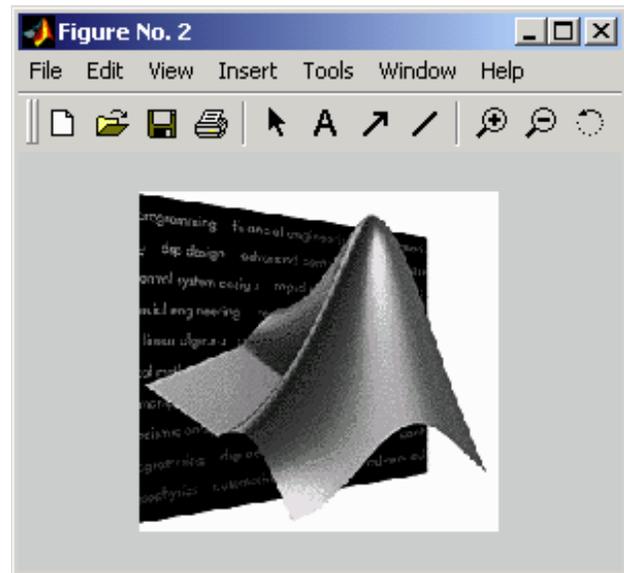


Figure:2

```
C:\MATLAB6p5\work\tutorial.m
File Edit View Text Debug Breakpoints Web Window Help
[Icons] Stack: Base
1  % Tutorial M-file
2  % Created by: Someone
3  % Created on: 9/11/03
4  % Last revised: 12/8/03
5
6  A = 'C:\MATLAB6p5\work\splash2new.jpg'; %designate matrix A as the specified file
7  B = im2double(imread(A, 'jpg')); %read and convert loaded image to class double
8  figure(1), imhist(B,256); %display histogram in figure window 1
9  D = imadjust(B,[0 1],[1 0]); %create negative of original image
10 figure(2), imshow(D); %display negative in figure window 2
11 E = histeq(B); %enhance contrast to equally spaced bins
12 figure(3), imshow(E); %display contrast enhancement in figure window 3
13 F = graythresh(B); %calculate threshold level
14 G = im2bw(B,F); %convert grayscale image to binary according to threshold
15 figure(4), imshow(G) %display binary image
script Ln 15 Col 50
```

Figure: 3

Reference(s): www.mathworks.com