**VISUAL COMPUTING AND MIXED REALITY**

**ASSIGNMENT 1**

**PROJECT REPORT**

**GROUP MEMBERS:**

NAVEENA KATPALLY

101551664

SHRUTHI KURA

101516439

**TASK 1**

**Write a MATLAB script which reads, converts a color image into grayscale image, and performs the histogram equalization on the grayscale one. Note that you are not allowed to use histeq, instead you have to implement a function to do histogram equalization. Do find some examples where histogram equalization achieves the worse results. Please discuss the pros and cons of histogram equalization method.**

**Analysis:**

* In this task we used ‘bear.jpg’ image of size (256\*256).To read this image we used ‘imread’ function.
* Next, we convert ‘bear.jpg’ rgb image to gray image by using rgb2gray function

**Histogram:**

A histogram is a graph it has a bars that represents the frequently occurring data in the whole data.

**Histogram of an image:**

Histogram of an image shows the frequency of pixel values. In an histogram the x-axis represents the intensity of gray levels and y-axis represents the frequency of pixels in these intensities.

‘histogram’ or ‘imhist’ are the functions used to create the histograms.

**Histogram equalization:**

Histogram equalization is used to enhance the contrast of an image. In this histogram equalization we distribute the pixels at each gray levels.The input image graylevels and output image graylevels may or may not be same.

‘histeq’ is an inbuilt function used for histogram equalization. As we are not supposed to use the histeq function we are using the histogram equalization algorithm for histogram equalization as below

**gray\_img=double(gray\_img);**

**maxv\_gray=max((max(gray\_img)));**

**[x y]=size(gray\_img);**

**l=x\*y;**

**m=zeros(1,300);**

**n=zeros(1,300);**

First we change the size of gray image from unsigned integer to double to make image with more clarity.

* We calculate max value of the gray image

Using maxv\_gray=max((max(gray\_img)))

Here max(gray\_img) prints the max value in each column and make it as vector matrix.

Then max(max(gray\_image)) is used to give the max value of the whole vector matrix and stores it in maxv\_gray.

* Compute the size of gray image and multiply rows and columns to get the total number of pixels.
* Preallocating of zeros to m and n variables so that they can be overwritten in the output.

**for i=1:x**

**for j=1:y**

**if gray\_img(i,j) == 0**

**gray\_img(i,j)=1;**

**end**

**end**

**end**

In the above code the pixels of grayscale image which are equal to 0 are replaced with 1 to enhance the contrast in image.

**for i=1:x**

**for j=1:y**

**t = gray\_img(i,j);**

**m(t) = m(t) + 1;**

**end**

**end**

The above loop is used to calculate the histogram of original image. The pixel values in the grayscale image are stored in t variable and the zeros are overwritten in the m with the values of t and the index of m is incremented.

**probf = m/l;**

**cumf(1) = probf(1);**

**for x=2:maxv\_gray**

**cumf(x) = probf(x) + cumf(x-1);**

**end**

**v = round(cumf \* maxv\_gray);**

**v = v + 1;**

The probability function is calculated as frequency/total no of pixels.

Cumulative distribution helps in equal distribution of pixel values and it enhances the contrast of the image.

Here we perform cumulative distribution as prob(x)+cum(x-1), we multiply the cumulative frequency values with maxv\_gray and we round off to the nearest value.

**for r=1:x**

**for s=1:y**

**temp=gray\_img(r,s);**

**val(r,s)=v(temp);**

**t=val(r,s);**

**n(t)=n(t)+1;**

**end**

**end**

Here, the loop is used to calculate the histogram of image after applying histogram equalization. we store the round off values in temporary variable val(r,s) and again we overwrite the values of n variable with new values.

**subplot(2,1,1)**

**bar(m)**

**title('Histogram of the Original Image');**

**subplot(2,1,2)**

**bar(n)**

**title('Histogram equalisation of Image');**

The above statements are used to represent the histogram of the original image and histogram of the equalization image in one figure using subplot function.

**ISSUES & Solutions :**

* The max(gray\_img) results in error during computing and it shows

Error using \*

Inner matrix dimensions must agree.

To resolve the above error ,we should use max(max(gray\_image)) which means it results maximum value of gray image.

* If the statement **m(t) = m(t) + 1** is not used then the correct histogram result is not displayed as it is not overwriting the preallocated values.
* We want to display the two histograms in one figure but by using plot function the output was not as expected .so to overcome that issue we use the sublot function.

**PROS :**

* Histogram equalization is used to increase the contrast of image.
* Using the histogram equalization the intensities can be equally distributed on the histogram.
* This technique is fairly straight forward and an invertible operator.
* Histogram equalization will work the best when applied to rgb images which have a high intensities rather than the gray scale image.

**Cons:**

* Histograms are not unique. It is indiscriminate.
* It may increase the contrast background noise , while decreasing the details of the image.
* It can provide unexpected results when it is applied to the images that have low color intensity

**Task 2:**

**Write a MATLAB script which reads, converts a color image into grayscale image, and performs min, max and median filtering on the grayscale image. Please discuss/elaborate on each filtering method (min, max, and median).**

**Analysis :**

To remove noise from the image we use Min,Max,Median filters.,and these filters are non linear filters

Min filter replaces the dark value in central pixel of a window.in this the pixel values are equal to 0.

Max filter is used to find the brightest poinst in an image,and gives the maximum value of the window.Max filter the pixes values are equal to 1.

Example: Consider the scenario in which the text is lightly printed

When the min filter is applied to above scenario it makes the letter thicker.

Consider another scenario in which the text string drawn with the thick pen.

When the max filter is applied to above scenario it makes the letter skinner

Here we read the image and convert the rgb image to grayscale image using “imread” and “rgb2gray”

**image=imread('spin.jpg');**

**figure**

**imshow(image)**

**title('rgb image')**

**image=rgb2gray(image);**

**figure**

**imshow(image);**

**title('gray image');**

we use the “spin.jpg” image here

we add the noise to the image and we show the image.

**image=imnoise(image,'salt & pepper',0.05);**

**figure,imshow(image)**

**title('noise image')**

To perform the median filter,we padd the zeros and find the medain value and replace it with center pixel

we preallocate the ouput matrix and perform zero padding as shown below

**L=zeros(size(A));**

**z=padarray(A,[1 1]);**

**x=[1:3]';**

**y=[1:3]';**

To perform Min filtering,we perform the reshape of the image and store it in an window,and find the maximum value in the window and store in a variable.Next change the ouput image to unsigned integer so that it ranges from 0-255

**for image= 1:size(z,1)-2**

**for j=1:size(z,2)-2**

**window=reshape(z(image+x-1,j+y-1),[],1);**

**L(image,j)=max(window);**

**L=uint8(L);**

end

end

To perform Max filtering, we perform the reshape of the image and store it in window and we find the minimum value of the window and store in the variable and next convert the variable into unsigned integer.

**for image= 1:size(z,1)-2**

**for j=1:size(z,2)-2**

**window=reshape(z(image+x-1,j+y-1),[],1);**

**L(image,j)=min(window);**

**end**

**end**

**L=uint8(L);**

**Issues and solutions:**

* If the rows and columns are not initialized the image will be resized and the imageproper image will not be displayed
* If we doesnot convert the output of min and max filters to unsigned integer it causes the figure blank.

**Contribution of group members:**

Naveena Katpally : Analayse the problem

Shrithi Kura : performing testing with different images

Combined effort: Developed the code and discussed issues and solutions