**Experiment 3. Histogram equalization**

1. **Purpose of the experiment**

1. Program to realize the following functions: read the stored black and white grayscale image and display it, display the grayscale histogram, perform histogram equalization processing on the image, display the processed image and histogram, and draw the grayscale transformation curve, and store the processed image.

1. **Experimental principle**

The central idea of the histogram equalization process is to change the grayscale histogram of the original image from a certain grayscale interval in the comparison set to a uniform distribution in the entire grayscale range. Histogram equalization is to stretch the image non-linearly and redistribute the pixel values of the image so that the number of pixels within a certain grayscale range is roughly the same.

1. **Experimental procedure**

Histogram equalization experiment code:

clc;

close all;

clear all;

Imag = imread('onion.png');

figure()

imshow(Imag),title('original image');

ImagGray = rgb2gray(Imag); % get grayscale image

figure()

imshow(ImagGray),title(' grayscale image ');

[r,c] = size(ImagGray); % get image size

% Statistical grayscale histogram

GrayPixNum = zeros(1,255);

for i = 1:r

for j = 1:c

% Count the number of occurrences of each pixel value

GrayPixNum(1,ImagGray(i,j)) = GrayPixNum(1,ImagGray(i,j))+1;

end

end

% Normalize the grayscale histogram

GrayPixPro = GrayPixNum./(r\*c);

figure()

plot(GrayPixPro),title('image histogram');

%-----------------------------------------------------------

% ----------------------- Histogram equalization ------------------------

%-----------------------------------------------------------

% Histogram accumulation

GrayAdd = zeros(1,255); %Get a vector with row 1, column 255, and value 0

GrayAdd(1,1) = GrayPixPro(1,1);

for i = 2:255

GrayAdd(1,i) = GrayAdd(1,i-1)+GrayPixPro(1,i); %Calculate the pixel cumulative sum

end

NewGray = round(GrayAdd.\*254.+0.5);

NewGrayPro = zeros(1,255);

for i = 1:255

GrayTemp = NewGray(1,i);

%Calculate the pixel values of the new image after equalization

NewGrayPro(1,GrayTemp) = NewGrayPro(1,GrayTemp)+GrayPixPro(1,i);

end

figure()

% ‘.\*’: Multiply the corresponding element values

plot(NewGrayPro.\*(r\*c)),title('equalization Histogram');

% Equalized image with equalization histogram and mapping

NewImag = zeros(r,c);

for i =1:r

for j = 1:c

NewImag(i,j) = NewGray(1,ImagGray(i,j));

end

end

NewImag = uint8(NewImag);

figure()

imshow(NewImag),title('equalized image');

figure()

plot(NewGray),title('grayscale transformation curve');

1. **To Consider**

Observe the experimental results, what are the changes and effects of the image after histogram equalization?