Report

I. Image enhancement on night-time place image

A. Summary about the enhancement procedure

1) Histogram equalization: The initial idea of histogram equalization is to convert the original image into an image with an nearly average distribution shown in histogram, which increases the range of a gray scale image.

- Convert a RGB image into a gray scale image and calculate the PDF of this gray scale image.
- Calculate the CDF of the gray scale image and round results, which is the regenerated gray scale value.
- To map gray values into the image and get an equalized image.

MATLAB code:

```
function [output_img] =hist_equ(input_img ,
    maxL)
%input_img: a grayscale image
%maxL: the maximum value in the range of
    input_img

n=size(input_img ,1)*size(input_img ,2);
L=maxL-1;
count_num=imhist(input_img);
p_rk=count_num/n;
s=L*cumsum(p_rk);
s=round(s);
output_img=uint8(zeros(size(input_img)));
for i=1:size(s,1)
    output_img(input_img==i-1)=s(i);
end
output_img = grs2rgb(output_img,hot);
imshowpair(output_img,input_img,'montage');
title 'After histogram equalization(left)/
    original image(right)'

end
```

2) Contrast using Adaptive Histogram Equalization: An adaptive histogram equalization (AHE) is to redistribute the lightness values of the image. It is suitable for improving the local contrast and enhancing the definitions of edges in each region of an image. Contrast limited adaptive histogram equalization (CLAHE) limits AHE's tendency to overamplify noise in relatively homogeneous regions of an image.

- An original image is divided into several blocks.
- Do equalization method respectively in every section in accordance with the histogram distribution of every block.
- The adjacent areas are stitched by linear difference, effectively reducing noise caused by the boundary artifacts.

MATLAB code:

```
image=imread('assignment.jpg');
I=rgb2gray(image);
n=size(I,1)*size(I,2);
L=256;
s=size(image);
i_test=double(image);
%contrast-limited Adaptive Histogram
    Equalization
count_r=imhist(image);
count_gray=imhist(I);
Lab=rgb2lab(image);
L=Lab(:,:,1)/100;
A=Lab(:,:,2)/100;
B=Lab(:,:,3)/100;
L = adapthisteq(L,'NumTiles',[2 2],'ClipLimit
    ',0.02);
A = adapthisteq(A,'NumTiles',[2 2],'ClipLimit
    ',0.01);
B = adapthisteq(B,'NumTiles',[2 2],'ClipLimit
    ',0.01);
Lab(:,:,1)=L*100;
Lab(:,:,2)=A*100;
Lab(:,:,3)=B*100;
J = lab2rgb(Lab);
figure
imshowpair(image,J,'montage');
title('Original image(left)/CLAHE(right)')
```

3) Piecewise linear grayscale stretching: Implement different linear functions into different corresponded gray scale blocks. In the piecewise function, the larger coefficient, the greater the stretch of the gray scale of the segment and the sensory contrast is enhanced. In the original image, the gray scale of image is ranging from 0 to 255.

The formulation is divided into three sections. The first range is from 0 to Xa, sequentially, the second from Xa to Xb, and the third from Xb to 255. The corresponded value in y coordinate is first ranging from 0 to Ya, second from Ya to Yb, third from Yb to 255. Every variants above can be various according to the effect of contrast. The enhanced image will be obtained by multiplying the original image with coefficients decided by myself.

MATLAB code:

```
function [dis_image] = linear_stretch(image,
    maxL,ya,yb,xa,xb)
input_img=rgb2gray(image);
[a,b]=size(input_img);
dis_image=uint8(zeros(a,b));
src_img=double(input_img);
%slope coefficient
k1=ya/xa;
k2=(yb-ya)/(xb-xa);
k3=(maxL-yb)/(maxL-xb);
for i=1:a
    for j=1:b
        if src_img(i,j)<=xa
            dis_image(i,j)=k1*src_img(i,j);
        elseif (src_img(i,j)>xa) && (src_img(
           i,j)<=xb)
                dis_image(i,j)=k2*(src_img(i,
                    j)-xa)+ya;
            else
                dis_image(i,j)=k3*(src_img(i,
                    j)-xb)+yb;
        end
    end
end
dis_image = grs2rgb(dis_image,hot);
imshowpair(dis_image,image,'montage');
title 'Piecewise linear stretching(left)/
    original image(right)'
end
```

B. Enhancement results

Compare regenerated images gained from different algorithms with original one:

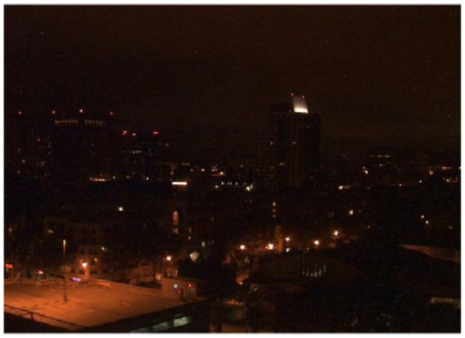

Fig. 1. original image

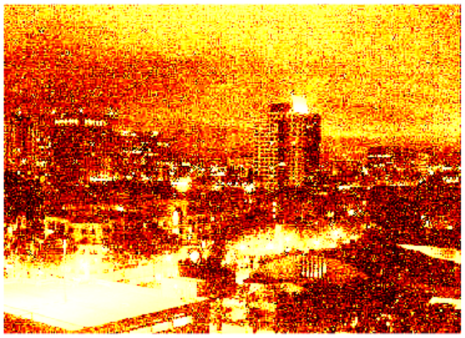

Fig. 2. After histogram equalization image

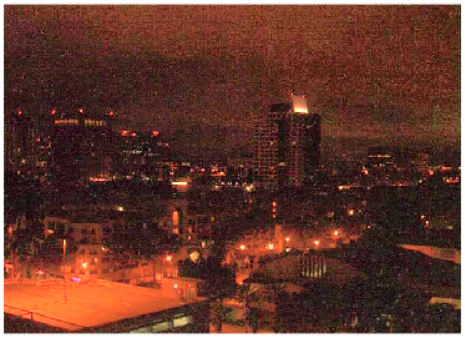

Fig. 3. CLAHE image

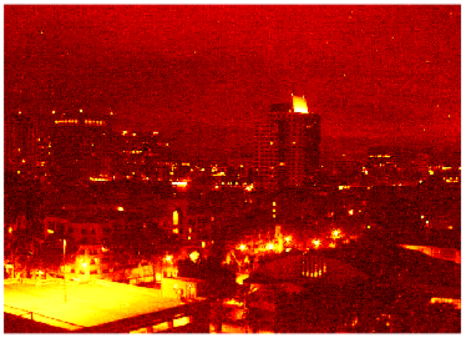

Fig. 4. Linear stretch image