Computer Vision I: Homework 3

Department: CSIE   Student ID: d11922016   Name: Jia-Wei, Liao

October 1, 2022

Write a program to generate images and histograms:

(a) original image and its histogram

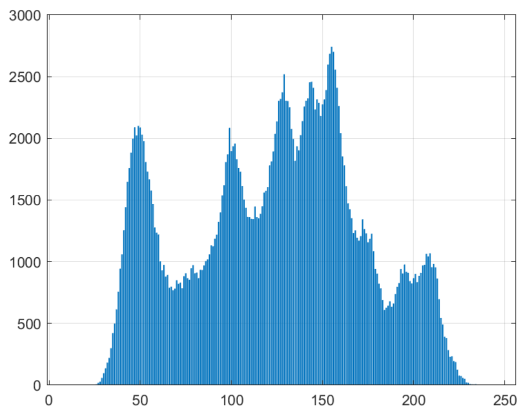
(b) image with intensity divided by 3 and its histogram

(c) image after applying histogram equalization to (b) and its histogram

1 Original Image

```
for g = 0:255
    for i = 1:m
        for j = 1:n
            if img(i, j) == g
                img_hist(1, g+1) = img_hist(1, g+1) + 1;
            end
        end
    end
end
```

(a) original image



(b) histogram of original image

Figure 1: original image

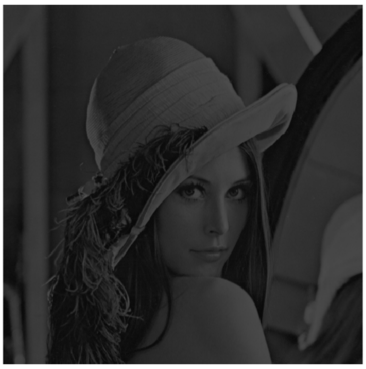
2 Low Contrast Image

Let Ω be the image domain and $g(i,j)$ be the intensity of image. We generate the low contrast image with divided the intensity by 3, that is,

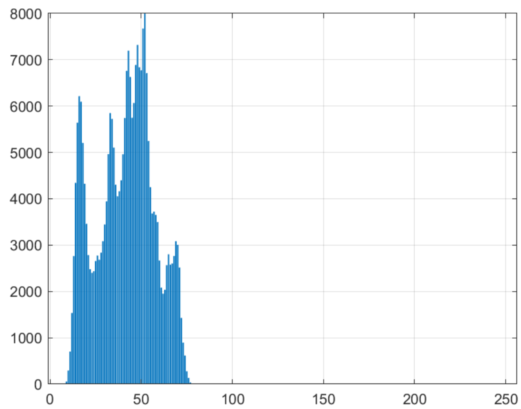
$$g'(i,j) = round(g(i,j)\,/\,3),\ (i,j) \in \Omega$$

```
for i = 1:m
    for j = 1:n
        img_divided_by_3(i, j) = round(img(i, j) / 3);
    end
end

img_divided_by_3 = uint8(img_divided_by_3);
img_divided_by_3_hist = computeHist(img_divided_by_3);
```



(a) low contrast image



(b) histogram of low contrast image

Figure 2: image intensity divided by 3

3 Histogram Equalization

1. Compute the min-max normalized image $\bar{g}(i,j)$.

$$\bar{g}(i,j) = \frac{g(i,j) - min}{max - min},\ (i,j) \in \Omega$$

   where Ω is image domain.

2. Compute the intensity distribution $f_X(k)$ of normalized image $\bar{g}(i,j)$.

$$f_X(k) = \frac{n_k}{n},\ k = 0, 1, ..., 255,$$

   where $n_k = |\{(i,j) \in \Omega \mid \bar{g}(i,j) = k\}|$ and $n = \sum_{k=0}^{255} n_k$.

3. Compute the c.d.f. $F_X(k)$ of normalized image $\bar{g}(i,j)$.

$$F_X(k) = \sum_{i=0}^{k} f_X(i),\ k = 0, 1, ..., 255.$$

4. Compute the histogram equalization image $\tilde{g}(i,j)$.

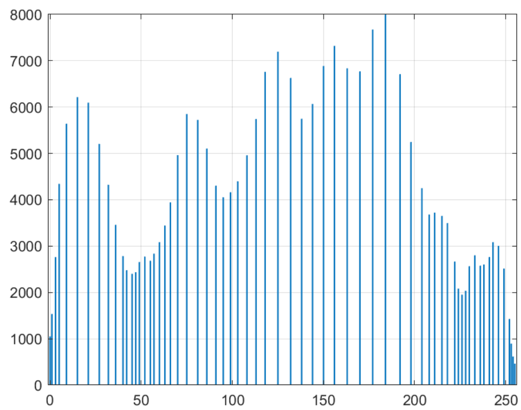
$$\tilde{g}(i,j) = round(F_X(\bar{g}(i,j)) \times 255),\ (i,j) \in \Omega$$

```
img_normal = computeMinMaxNormalize(double(img_divided_by_3));
img_pdf = computePDF(img_normal);
img_cdf = computeCDF(img_pdf);
he_img = zeros(m, n);

for i = 1:m
    for j = 1:n
        if img_normal(i, j) > 0
            he_img(i, j) = round(img_cdf(img_normal(i, j)) * 255);
        end
    end
end

he_img = uint8(he_img);
he_img_hist = computeHist(he_img);
```

(a) image after histogram equalization



(b) histogram of image after histogram equalization

Figure 3: histogram equalization

Summary.

In this homework, we aim to enhance the contrast between image intensities. We use MATLAB to implement the histogram equalization algorithm. The goal of histogram equalization is to transform the image intensity distribution into a uniform distribution.

Since the distribution of image intensity is discrete, we must round the result, which makes the result look different from a uniform distribution. But when we computed the c.d.f, it appears approximately uniform distribution.