Computer Vision HW3

R07921010 余奇安

1.Histogram Equalization

$$s_k = 255 \sum_{j=0}^{k} \frac{n_j}{n}$$

$k = 0, 1, ..., 255$, n_j: number of pixels with intensity j

n : total number of pixel

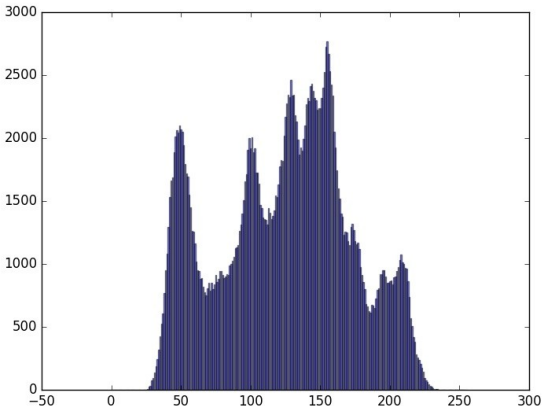
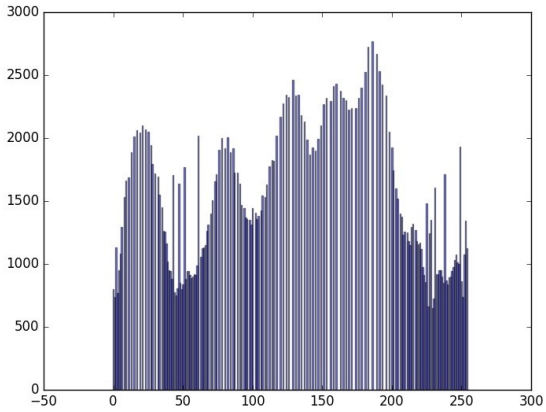
If pixel I(im,i,j)=k, after transform I(im, i,j)=S_k

Main code:

```
def Histo_eq(img):
    nim=img.copy()
    row=img.shape[0]
    col=img.shape[1]
    n=row*col

    y=[0]*256
    s=[0]*256
    for i in range(row):
        for j in range(col):
            y[img[i][j]]+=1
    #print (y)
    s[0]=255*y[0]/n
    for i in range(1,256):
        s[i]=(s[i-1]+255*float(y[i])/n)
    #print(s)
    for i in range(row):
        for j in range(col):
            k=nim[i][j]
            nim[i][j]=s[k]
    #print(nim)
    cv2.imwrite("Histo_eq.jpg",nim)
    cv2.imshow("Histo_eq",nim)
    cv2.waitKey(0)
    draw_Histo(nim)
```

List y to calculate the numbers of pixel with intensity y[k],(k=0~255)
List s store the value after transform.

Original figure	Linear histogram equalization
	
Original histogram	Linear histogram equalization histogram
	

We can find that after histogram equalization, the contrast of image becomes more distinct, and the range of histogram becomes wider.