Computer Vision HW#3

R09922144 王懷德

- (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

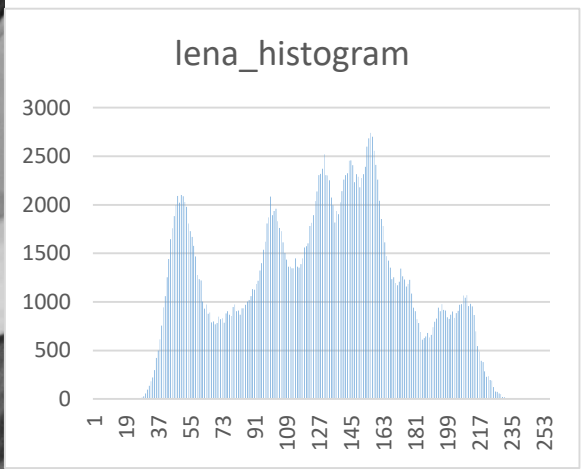
Tools:

C++,opencv(read write),excel

Report:

A:

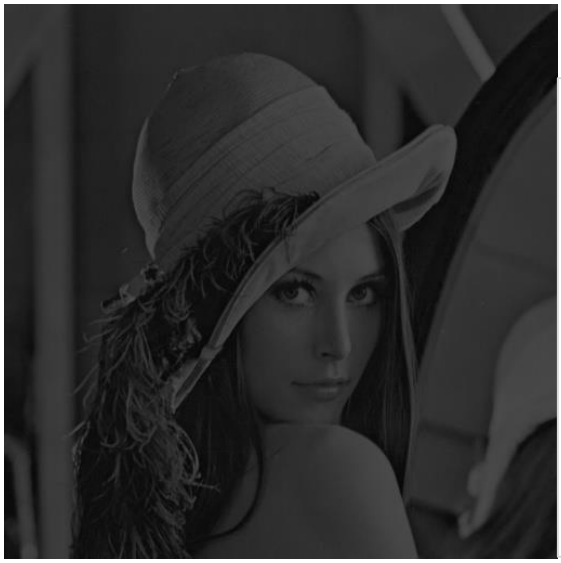
Using nested loop go through the image , and count each pixel value ,save the result in csv .plot the histogram with excel.

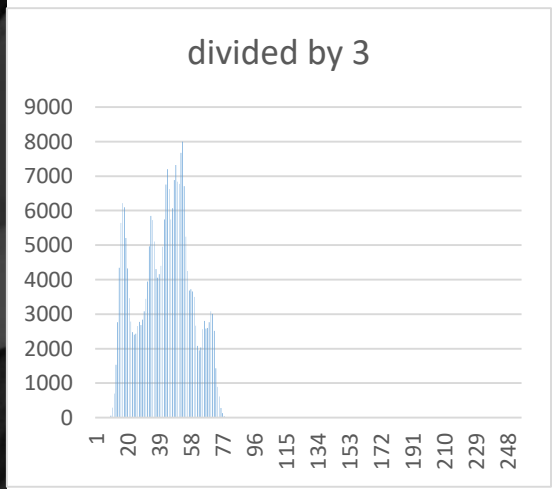


```
11
12
13  void histogram(Mat img) {
14      int hist[256] = { 0 };
15      for (int i = 0; i < img.rows; i++) {
16          for (int j = 0; j < img.cols; j++) {
17              hist[img.at<uchar>(i, j)]++;
18          }
19      }
20
21      // write image
22      std::fstream f;
23      f.open("histogram.csv", std::ios::out);
24      for (int i = 0; i < 256; i++)
25      {
26          f << hist[i];
27          f << '\n';
28      }
```

B:

Divide the intensity by 3 and repeat A.



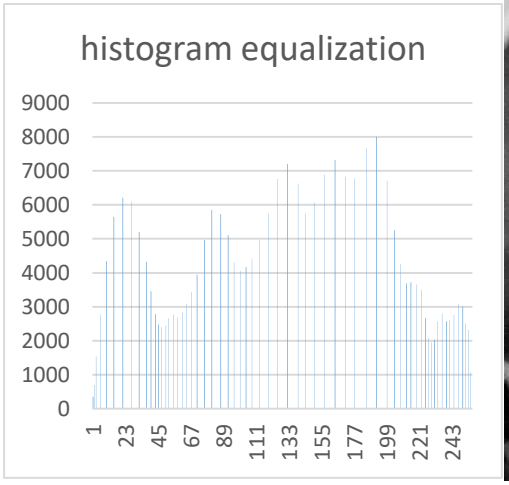



```
}
void div3(Mat img) {
    img /= 3;

    int hist[256] = { 0 };
    for (int i = 0; i < img.rows; i++) {
        for (int j = 0; j < img.cols; j++) {
            hist[img.at<uchar>(i, j)]++;
        }
    }

    // write image
    std::fstream f;
    f.open("histogramdv3.csv", std::ios::out);
    for (int i = 0; i < 256; i++)
    {
        f << hist[i];
        f << '\n';
    }
    imwrite("dividedby3.jpg", img);

}
void equalization(Mat img) {
```

C:

After B , calculate the pdf . And turn it to cdf ,then multiplied each pixel with 255 and the cdf of it's value.

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



```
void equalization(Mat img) {
    int hist[256] = { 0 };
    for (int i = 0; i < img.rows; i++) {
        for (int j = 0; j < img.cols; j++) {
            hist[img.at<uchar>(i, j)]++;
        }
    }
    double cdf[256] = { 0 };
    int total = 0;
    for (int i = 0; i < 256; i++) {
        total += hist[i];
        cdf[i] = total;
    }
    for (int i = 0; i < 256; i++) {

        cdf[i] /= total;
    }

    for (int i = 0; i < img.rows; i++) {
        for (int j = 0; j < img.cols; j++) {
            img.at<uchar>(i, j) = cdf[img.at<uchar>(i, j)] * 255;
        }
    }
    int hist2[256] = { 0 };
    for (int i = 0; i < img.rows; i++) {
        for (int j = 0; j < img.cols; j++) {
            hist2[img.at<uchar>(i, j)]++;
        }
    }
```