

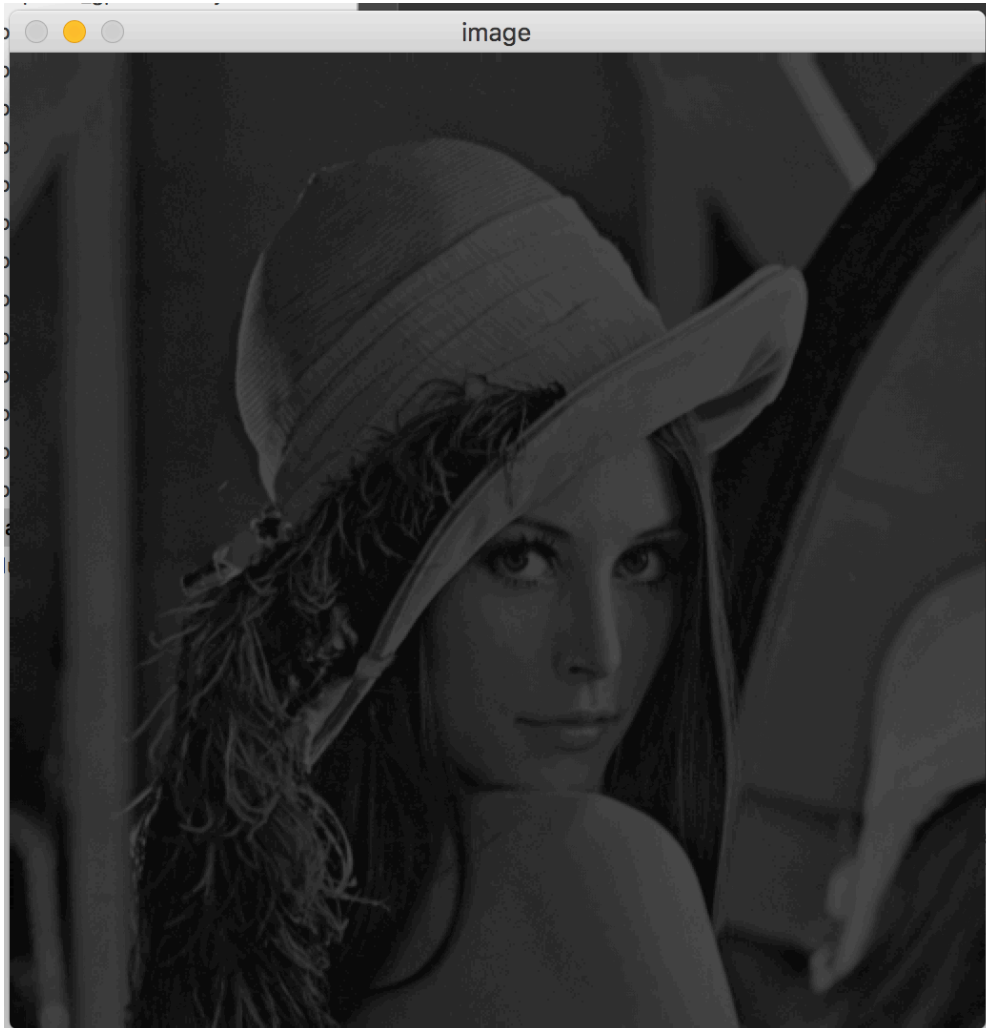
電腦視覺作業三

Computer Vision HW3
R04525092 工科碩二 鄭力文

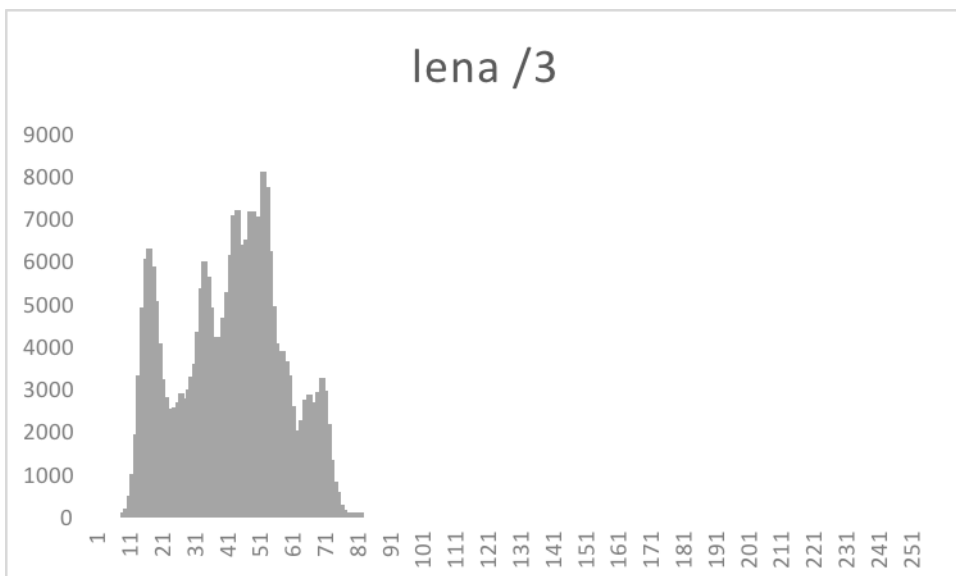
使用語言 C++ with openCV

1. Histogram Equalization

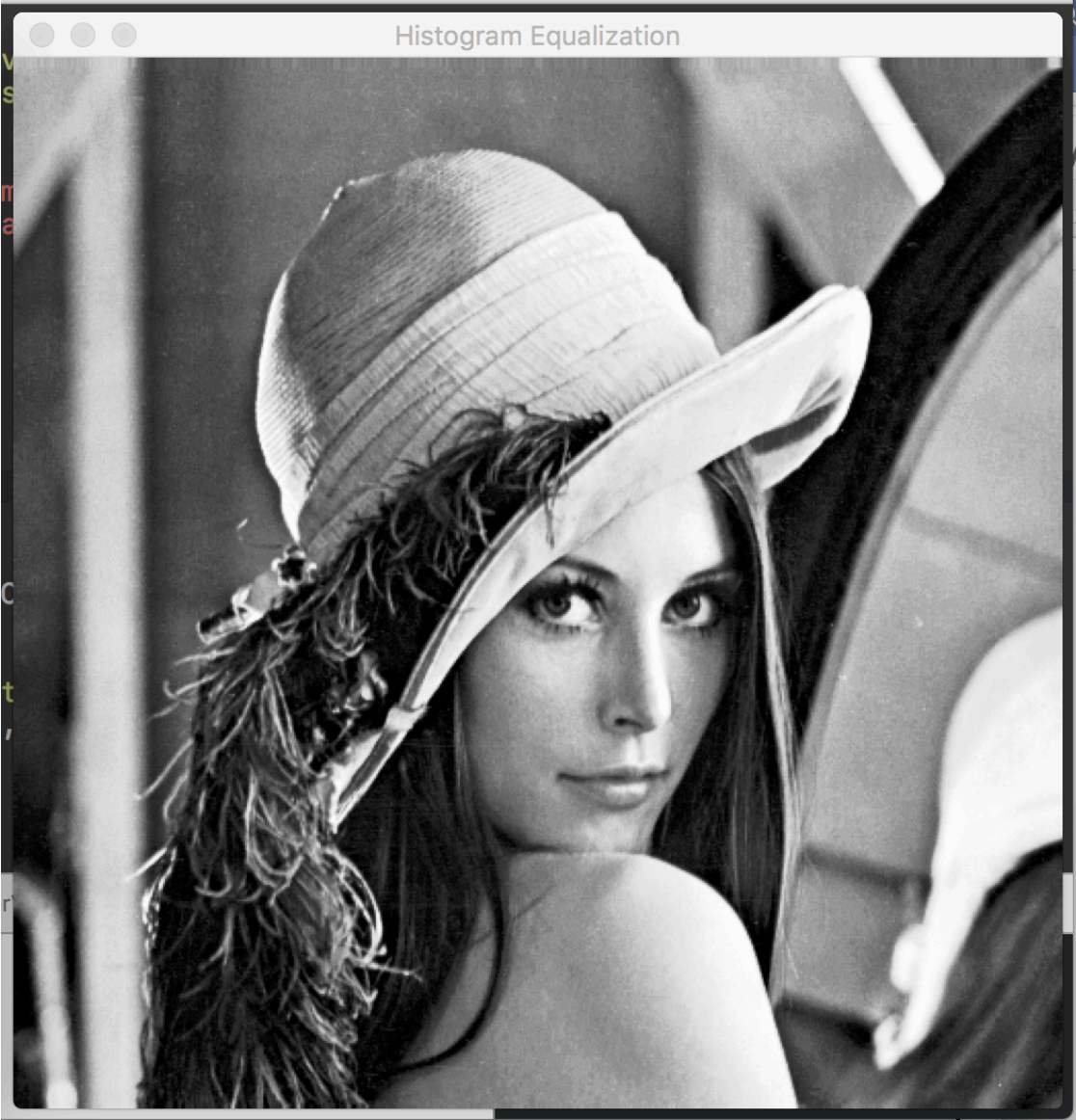
原圖 lena /3



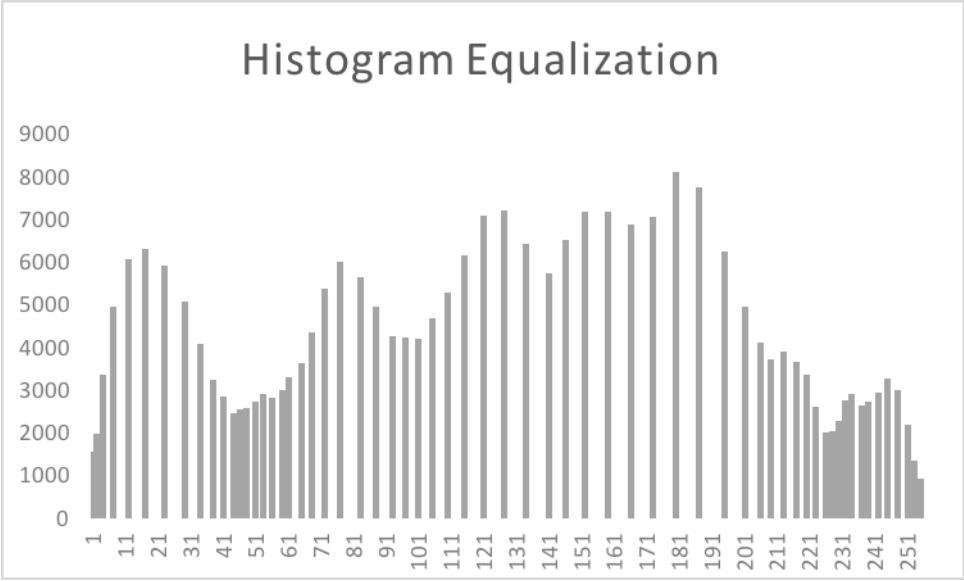
原圖 lena /3 Histogram



Histogram Equalization



原圖 Histogram Equalization Histogram



fuction sk

計算每一個像素的 Sk function

- histogram equalization histogram linearization

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

```
//fuction sk
double sk(int k){
    double value=0;
    for (int i=0;i<k;i++){
        value+=histogram[i];
    }

    value=value*255/(rows*cols);

    return value;
}
```

主函式

將個點進行運算

```
//主函式

int main(int argc, const char * argv[]) {

    cout<<"R04525092 工科 鄭力文"<<endl;

    //建立histogram table

    for (int y=0;y<cols;y++){
        for(int x=0;x<rows;x++){
```

```
//          將每一像素除以3

    for(int z=0;z<3;z++){
        image.at<Vec3b>(x,y)[z] =image.at<Vec3b>(x,y)[z]/3;
        thresholdingImg.at<Vec3b>(x,y)[z]
=thresholdingImg.at<Vec3b>(x,y)[z]/3;

    }
    histogram[thresholdingImg.at<Vec3b>(x,y)[0]]+=1;
}
}
```

- for every pixel if $I(im, i, j) = k$ then $I(imhe, i, j) = s_k$

```
for (int y=0;y<cols;y++){
    for(int x=0;x<rows;x++){
        int k =thresholdingImg.at<Vec3b>(x,y)[0];
        int ske=int(ske(k));
        for(int z=0;z<3;z++){
            thresholdingImg.at<Vec3b>(x,y)[z] =ske;
        }
        histogram2[ske]+=1;
    }
}
```

//2值化 輸出csv

```
ofstream fsOutput("histogram.csv", ios::app);
ofstream fsOutput2("histogram2.csv", ios::app);

for(int i=0;i<256;i++){
    fsOutput <<i<<","<<histogram[i] << endl;
    fsOutput2 <<i<<","<<histogram2[i] << endl;
}
```

```
fsOutput.close();
fsOutput2.close();

//顯示結果

namedWindow("image", WINDOW_AUTOSIZE);
imshow("image", image);

namedWindow("Histogram Equalization", WINDOW_AUTOSIZE);
imshow("Histogram Equalization", thresholdingImg);

waitKey(0);

return 0;
}
```