**Assignment 8**

**Part 1 Code**

#include <opencv2/core/core.hpp>

#include <opencv2/highgui/highgui.hpp>

#include <opencv2/imgproc/imgproc.hpp>

#include <opencv2/nonfree/features2d.hpp>

#include <opencv2/nonfree/nonfree.hpp>

#include <iostream>

#include <fstream>

#include <vector>

#include <math.h>

using namespace cv;

using namespace std;

int main()

{

//load the image

IplImage\* img = cvLoadImage("SIFT\_SURF.png", 0);

Mat featureImgSIFT(img, true);

Mat featureImgSURF(img, true);

// 特征点的向量

vector<KeyPoint>keypoints;

// 构造SIFT特征检测器

SiftFeatureDetector sift(

0.03, // 特征的阈值

10.); // 用于降低

// 检测SIFT特征值

sift.detect(img, keypoints);

// 绘制SIFT特征图

drawKeypoints(featureImgSIFT, // 原始图像

keypoints, // 特征点的向量

featureImgSIFT, // 生成图像

cv::Scalar(255, 255, 255), // 特征点的颜色

cv::DrawMatchesFlags::DRAW\_RICH\_KEYPOINTS); // 标志位

cvSaveImage("featureImgSIFT.png", &IplImage(featureImgSIFT));

// 特征点的向量

vector<KeyPoint>keypoints2;

// 构造SURF特征检测器

SurfFeatureDetector surf(2500);

// 检测SURF特征值

surf.detect(img, keypoints2);

// 绘制SIFT特征图

drawKeypoints(featureImgSURF, // 原始图像

keypoints2, // 特征点的向量

featureImgSURF, // 生成图像

cv::Scalar(255, 255, 255), // 特征点的颜色

cv::DrawMatchesFlags::DRAW\_RICH\_KEYPOINTS); // 标志位

cvSaveImage("featureImgSURF.png", &IplImage(featureImgSURF));

cvReleaseImage(&img);

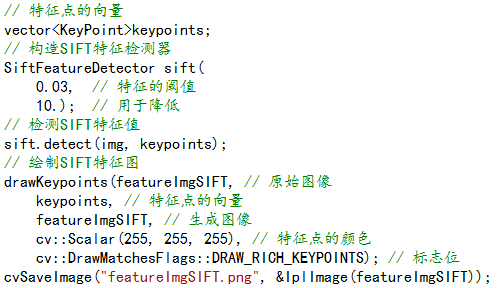
featureImgSIFT.release();

featureImgSURF.release();

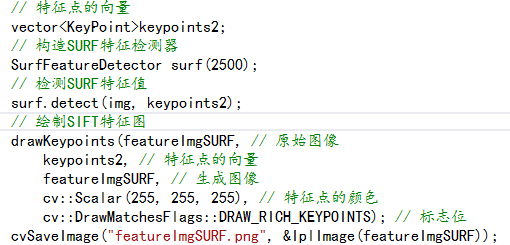
}

**Part 2 explanation**

1. SIFT的apply



1. SURF的apply



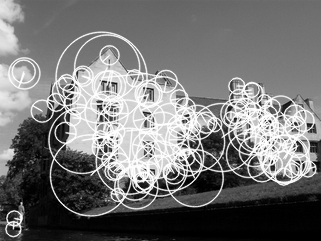
**Part 3 Output**



原图



SIFT处理后



SURF处理后