Assignment#2

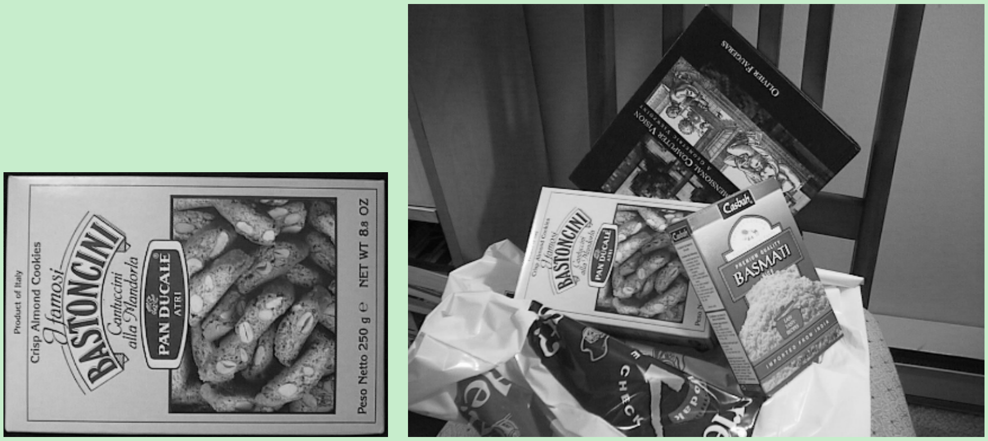
Wang Ruchen

April 29, 2015

In this assignment, I used c++ and OpenCV to complete the assignment.

1. Strategy

I implement the object detection with the figures are shown in Figure 1:



(a)

(b)

图 1: object image and scene image

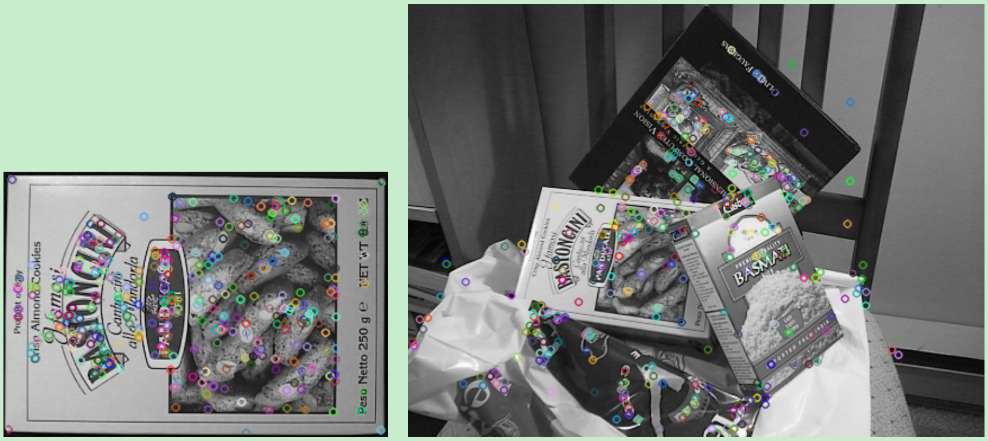
2. Color space conversion

The object image and scene image are converted from RGB to Gray.

3. Feature2D detection and calculate descriptors

I choose SIFT to detect keypoints of object image and scene image. OpenCV_Contrib is an extensive library of OpenCV. The library has a function about SIFT. The function can detect keypoints and calculate descriptors. The results of keypoints detection are shown in Figure 2.

```
    Ptr<SIFT> sift=SIFT::create (400);
    vector<KeyPoint> keyPoiObj, keyPoiSce;
    Mat outKeyPoiObj, outKeyPoiSce;
    sift->detectAndCompute(objectImg, Mat(), keyPoiObj, outKeyPoiObj);
    sift->detectAndCompute(sceneImg, Mat(), keyPoiSce, outKeyPoiSce);
```



(a) (b)

图 2: Features2D detection

4. Match descriptors

I used Brute-force match to match keypoints between object image and scene image. The distance calculation algorithm is euclidean distance in this function. The match result is shown in Figure 3.

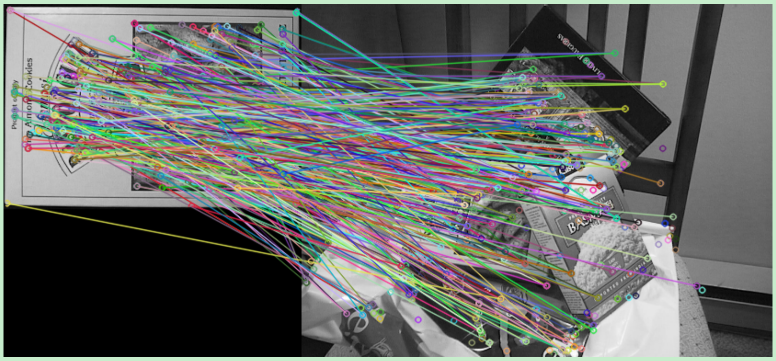


图 3: match result

The matches whose distances are less than 2*mindistance are good matches. The result of good matches is shown in Figure 4.

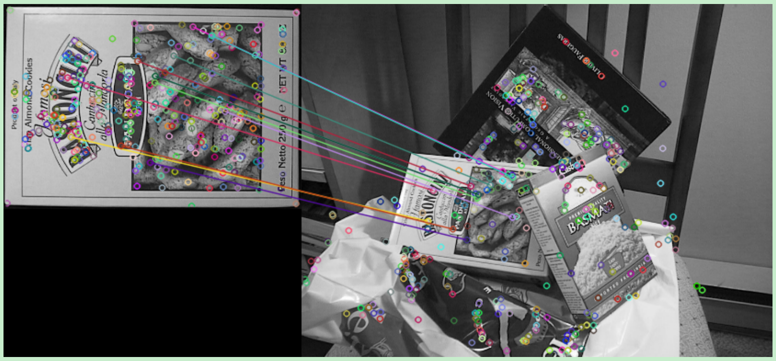


图 4: good match result

5. Find homography transformation

I used “findHomography” function to get a matrix of homography transformation between two different point sets.

6. Perspective transform

I used “perspectiveTransform” function and the matrix of homography transformation to get four points of scene image.

7. Result

According to the matrix of homography transformation, draw lines between the four points. The result of object detection is shown in Figure 5:

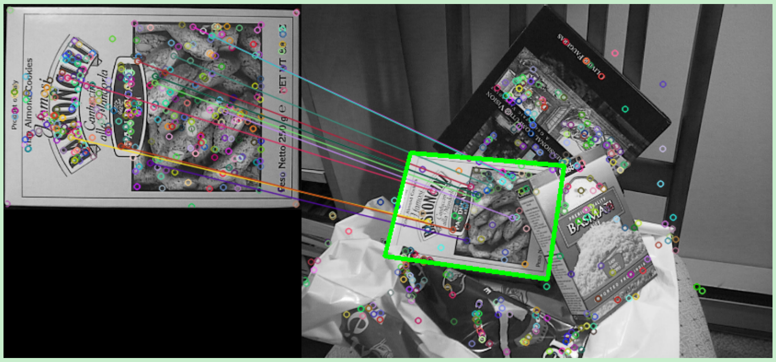


图 5: result