Feature Dectection Lab - Report

Minh-Quan DAO, Isaac Gutierrez Payan

1 Introduction

The goal of this lab is to track a predefined object in a video using feature detection and matching tools. This is done with the four-step algorithm below

1. Find features in query and train image
2. Match features found in two images
3. Find the homography between two images using matched features
4. Localize the object in train image by performing perspective transform on bounding box of the object in query image

Here, the query image refers to the reference image which contains the predefined object, while the train image is any arbitrary frame which one wants to track the object.

Since this lab focuses on putting the four steps above together to successfully track the object through out the video, the implementation of each step is not elaborated in this report. However, to avoid ambiguity, the features detector and matching algorithm are respectively chosen to be SURF and FLANN based matcher.

There are two approaches to defining the query image (i.e. reference image) which are

- Using the first frame (tracking mode 0)
- Using the previous frame (tracking mode 1)

The usage of the accompany code of this report is:

```
$ python2 tracking_full.py path_to_video_file tracking_mode
```

2 Tracking Algorithm

The tracking algorithm with the query image being either the first frame or the previous frame in Section 1 is implemented according to the flow chart in Fig.1

2.1 Region of interest in query image

Beside the object of interest, the query image (Fig.2) contains other objects which can inject wrong features (features do not belong to the object of interest) to feature matching steps. Therefore, the search for features in query image should be limited inside a Region Of Interest (ROI) defined by the space inside the rectangle in Fig.3

For SURF detector to understand, the ROI is represented by a mask (Fig.4) which is a binary image of the same size as query image, composed by white pixels inside the ROI and black pixels elsewhere.

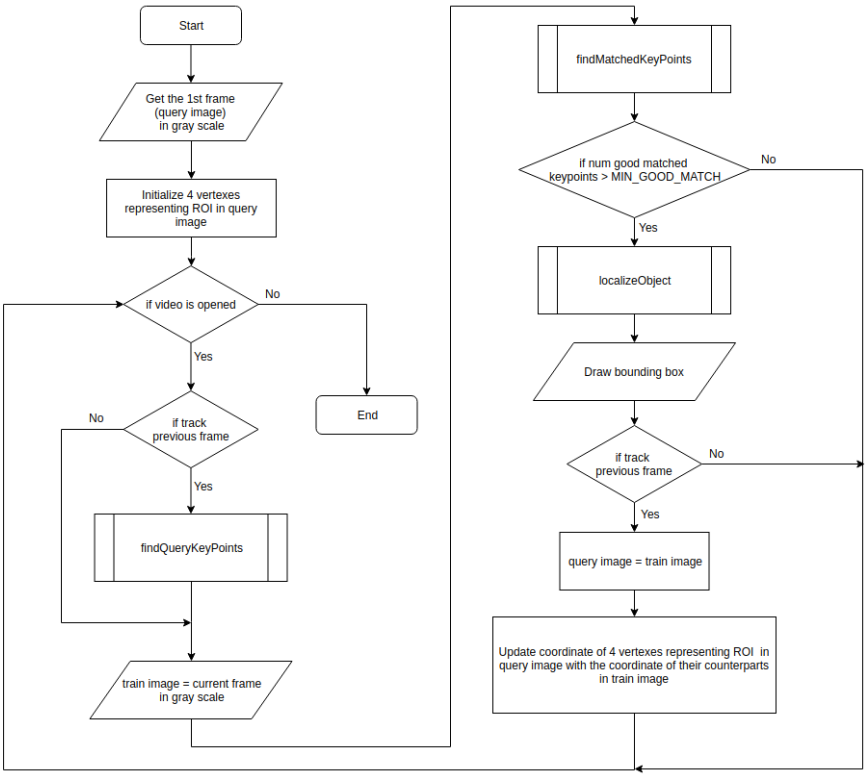

Figure 1: The flow chart of tracking algorithm

Figure 2: An example of query image

Figure 3: ROI

Figure 4: Mask

Given the query image in gray scale an its ROI, the keypoints and their descriptors are identified according to the procedure in Fig.9. The key points detected by SURF without and with mask are shown in Fig.5 and Fig.6, respectively. As can be seen from these figure, the application of mask results in only key points belong to the object of interest.

2.2 Find match key points

The sub procedure to find the key points in the train image which have a match in the query image is illustrated by the flow chart in Fig.10.

Since the KNN Matcher is used, each of the feature descriptor found in the train image is matched with a number of closest descriptor in the query image. This number of closest descriptors is set to 2, which means each descriptor in train image is matched with two descriptors in query image. An example of matched keypoints between query image and train image is shown in Fig.7. As can be seen from this figure, there are a lot of matches between two images which leads to a large number of points fed to estimation of the homography step. However, these matches are not good input to find homography since each match is not one-one.

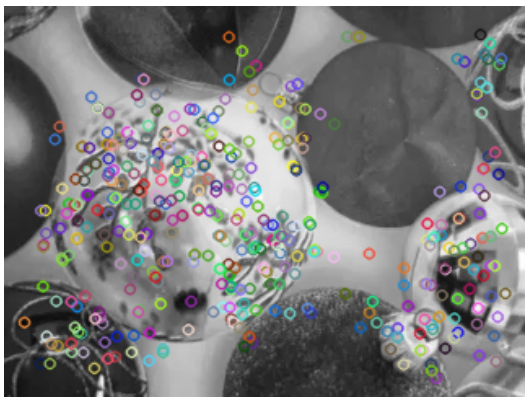

Figure 5: Key points without ROI

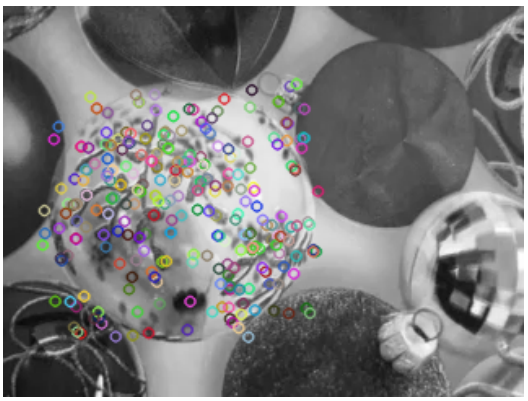

Figure 6: Key points with ROI

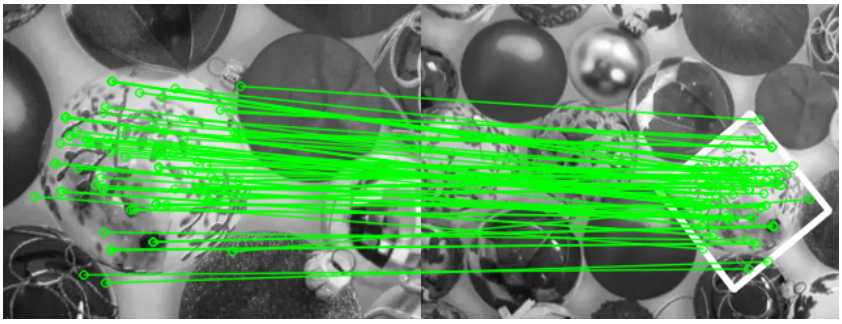

Figure 7: Matched keypoints without Lowe test

To provide good pairs of matched keypoints for estimation of homography step, for each descriptor in the query image, the Lowe ratio test is carried out to choose which one of its two matched descriptors in train image is the better match. An example of matched keypoints after conducting Lowe test is shown in Fig.8.

2.3 Localize object in train image

The location of the object of interest in the train image is retrieved by mapping four vertexes of the ROI in query image to train image. The flow chart of this procedure is displayed in Fig.11

This step depends on estimating the homography between train image and query image which in turn depends on the number of well matched keypoints. The evolution of this number during tracking the first frame and previous frame are shown in Fig.12 and Fig.13, respectively. As can be seen from these figures, the number of well match keypoints while tracking the first frame is always higher. This leads to more stable tracking result (the bounding box of

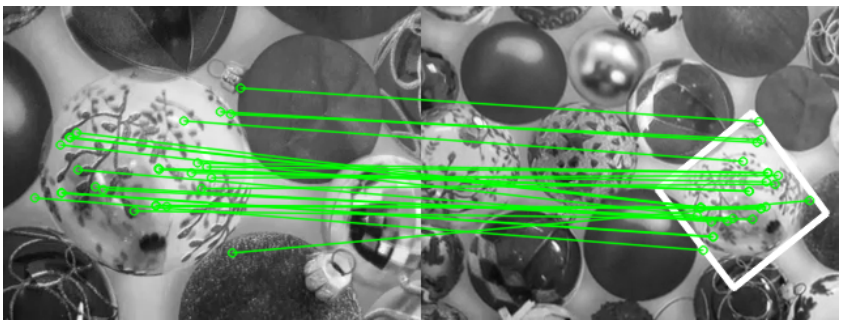

Figure 8: Matched keypoints with Lowe test

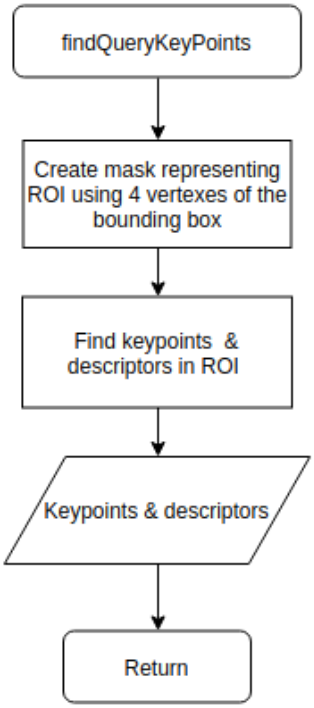

Figure 9: The flow chart for keypoints and descriptor in queyr image

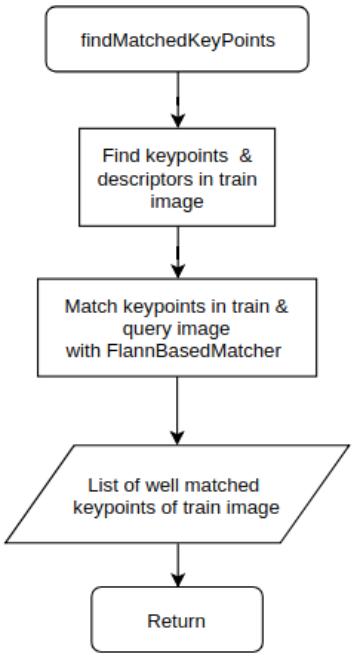

Figure 10: The flow chart for finding match key points in query and train image

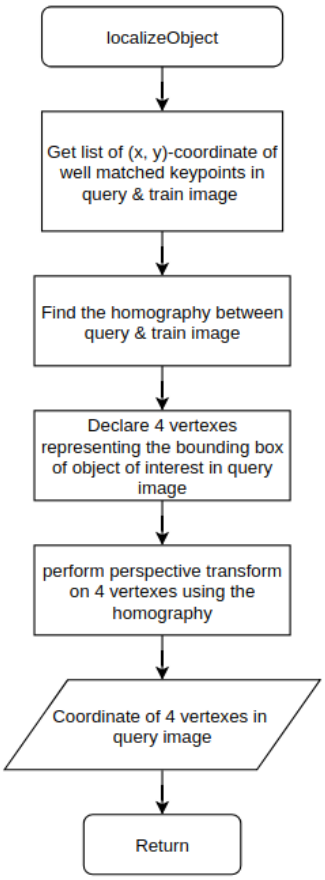

Figure 11: The flow chart for localizing object in train image

the object in train image keeps its rectangular shape relatively well). Furthermore, when the camera zooms out (in the middle of the video), the number of well match keypoints significantly decreases, nearly equal to the minimum number needed to estimate the homography - 8 points (the green line). This results in the strong distortion in the bounding box of tracked object.

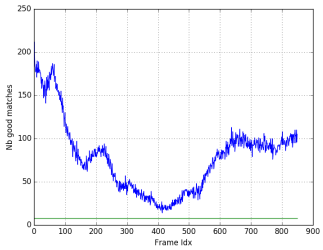

Figure 12: The number of well match keypoints while tracking the first frame

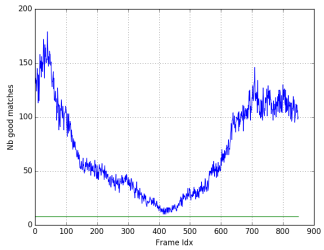

Figure 13: The number of well match keypoints while tracking the previous frame