MSFT-Visual Servoing Practical 2
3D Model Based Tracking

Masters in Computer Vision

UNIVERSITE DE BOURGOGNE

Centre Universitaire Condorcet - UB, Le Creusot

19 December 2017

Team Members:
Mohit Kumar Ahuja
GopiKrishna Erabati
Dousai Nayee Muddin Khan

Supervisor:
Dr. Nathan Crombez

Task: 3D Model-Based Tracking

What we do:
We tried to track a 3D model (box) in a sequence of images using tracking functions.

How we do:
The Image acquisition and initialization of tracker is done by the following sequence:

1. We need to start Kinect for this we used *MyFreenectDevice* object, as we acquired 50 frames we stopped streaming to see the initial object to select points.
2. We have to replace the 3D points in "Box.cao" file with the measured 3D points of Box which we have to track in video. The distances of tracked object are given in "Box.cao" file.
3. We have to edit the faces in the "Box.cao" file with accordance to the sequence of 3D corner points we gave above as seen in Fig.1.

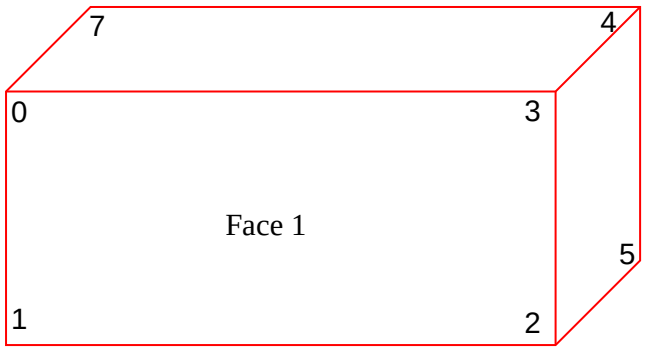

Fig. 1 Wireframe diagram of Cuboid object we used to track

4. On the last image grabbed, we will right click to provide 4 points of corners but those 4 points should not be on the same frame and we have to define those points in the "Box.init" file.
5. When we will have 4 points, press left click to validate. And it will start tracking the box in every frame. Here, camera parameters are required to get the object in image frame which is used to display the tracked object correctly.

And the tracking of box in sequence of images is done in following sequence in loop:

1. getvideo, -- Get the frames
2. display , -- Display frames
3. track, -- Track the Box
4. getPose -- Get the pose of the Box in every Frame
5. display , -- Display the tracked box in every frame.
6. displayFrame -- Display the coordinate frame of the box in blue.
7. flush. -- To flush the display and effectively display image

Why we do:
This task is done to effectively grab the frames of kinect and display them as a video flow and how VISP library can be used to detect and track 3d Box which can be used as a basis for visual servoing. The tracking result of box can be seen in Fig. 2.

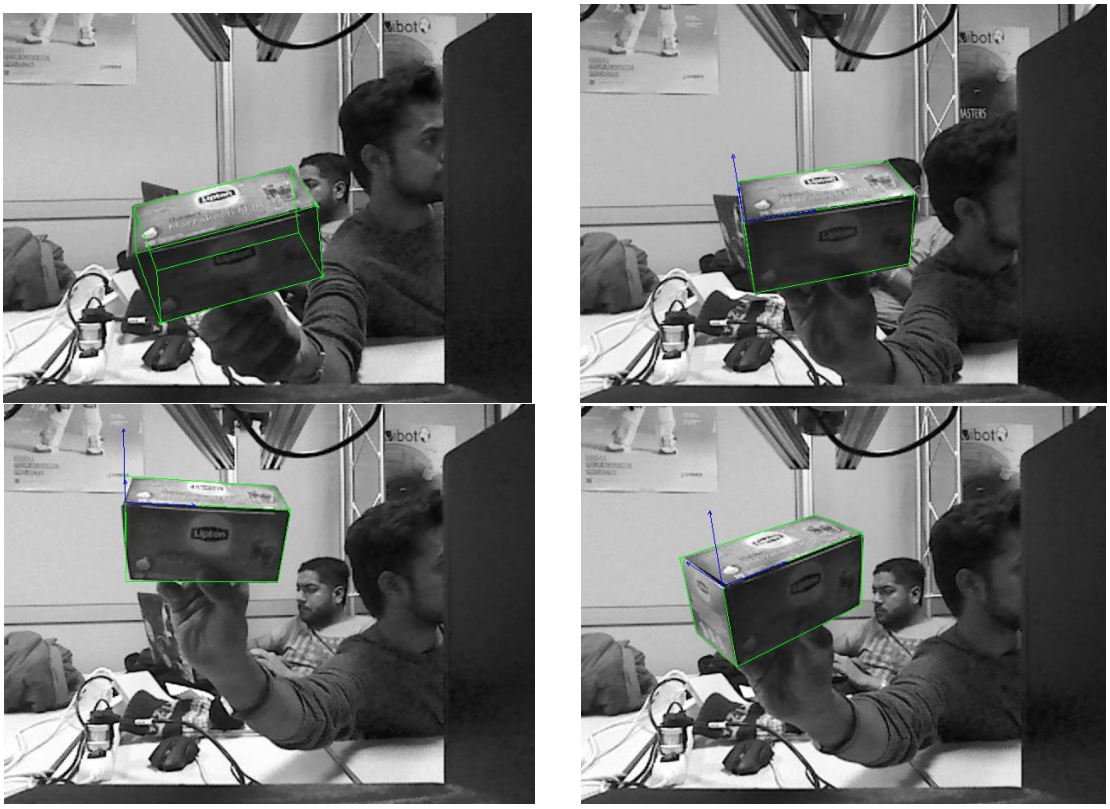

Figure 2. Result of tracking of the Box in every frame

Conclusion:

We have done 3D model object detection, tracking, computing pose using VISP library. We had seen great advantages of this library as we can track and detect using a single line of code and the data structures defined for holding data are well defined and the class and methods used in the library are well documented with many examples. We found it's easy to use this library for visual servoing in the context of our tasks.

The code is found here.
The video can be found here.