Final Project Report

ELEC6910R Robotic Perception and Learning

Liu Siqi, Wang Ke

1. Introduction

The project is focused on these aims :

1) Build 2D grid map with laserscan data avia rviz
2) Control the mobile robot in the vrep simulation environment with keyboard
3) Image Recognition and localization
4) Visual tracking (follow the yellow ball)
5) roslaunch ros nodes

2. Environment setup

1) ROS install and catkin :

   a. Install catkin: http://wiki.ros.org/catkin

   b. Install ROS: http://wiki.ros.org/ROS/Installation

   c. Configure and create catkin workspace

```
$ echo "source /opt/ros/kinetic/setup.bash" >>
~/.bashrc
$ source ~/.bashrc
$ mkdir -p ~/catkin_ws/src
$ cd ~/catkin_ws/
$ catkin_make
$ echo "source ~/catkin_ws/devel/setup.bash" >>
~/.bashrc
$ source ~/.bashrc
```

   d. Install the needed ros packages

```
#install hector slam
$ sudo apt install ros kinetic hector slam
#install key teleop
$ sudo apt install ros kinetic key teleop
```

2) Install V-REP :

   a. Dowload V-REP from **http://www.coppeliarobotics.com/downloads.html**

```
$ cp V-REP_PRO_EDU_V3_4_0_Linux.tar.gz ~
$ tar -zvxf ~/V-REP_PRO_EDU_V3_4_0_Linux.tar.gz
```

```
$ mkdir ~/V-REP
$ mv ~/V-REP_PRO_EDU_V3_4_0_Linux ~/V-REP
```

b. open one terminal and run $ roscore

c. open another terminal and run $. ~/V-REP/vrep.sh

Please pay attention to these message and if you see

```
Plugin 'RosInterface': loading...
Plugin 'RosInterface': warning: replaced variable 'simROS'
Plugin 'RosInterface': load succeeded.
Plugin 'SDF': loading...
Plugin 'SDF': warning: replaced variable 'simSDF'
Plugin 'SDF': load succeeded.
Plugin 'SurfaceReconstruction': loading...
Plugin 'SurfaceReconstruction': warning: replaced variable 'sim
Plugin 'SurfaceReconstruction': load succeeded.
Plugin 'Urdf': loading...
Plugin 'Urdf': load succeeded.
Plugin 'Vision': loading...
Plugin 'Vision': load succeeded.
Using the 'MeshCalc' plugin.
Initializing the Bullet physics engine in plugin 'DynamicsBulle
Engine version: 2.78
Plugin version: 10
Initialization successful.
```

The most important is :

```
Plugin 'RosInterface': loading...
Plugin 'RosInterface': load succeeded.
```

d. open env.ttt in vrep's scene and press the start bottom

e. control the robot by keyboard

```
$ roslaunch key_vel.launch
```

```
sq@sq-GE63VR-7RF:~$ roslaunch catkin_ws/src/vrep_slam/key_vel.launch
```

f. launch the ros nodes for SLAM and face detection/recognition

```
sq@sq-GE63VR-7RF:~/catkin_ws$ roslaunch src/vrep_slam/hector.launch
... logging to /home/sq/.ros/log/508f9b14-62e7-11e8-81b5-9cb6d06a501b/roslaunch-sq-GE63VR-
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://sq-GE63VR-7RF:37451/

SUMMARY
========

PARAMETERS
 * /face_visualizer/cleanup_delay: 0
 * /facedetector/detector_file2: /home/sq/catkin_w...
 * /facedetector/detector_file: /home/sq/catkin_w...
 * /facedetector/recognition_file: /home/sq/catkin_w...
 * /facedetector/show_cv_window: True
 * /hector_mapping/advertise_map_service: True
 * /hector_mapping/base_frame: base_link
 * /hector_mapping/laser_z_max_value: 1.0
 * /hector_mapping/laser_z_min_value: -1.0
 * /hector_mapping/map_frame: map
 * /hector_mapping/map_multi_res_levels: 2
 * /hector_mapping/map_resolution: 0.05
 * /hector_mapping/map_size: 1024
 * /hector_mapping/map_start_x: 0.5
 * /hector_mapping/map_start_y: 0.5
 * /hector_mapping/map_update_angle_thresh: 0.06
 * /hector_mapping/map_update_distance_thresh: 0.4
 * /hector_mapping/odom_frame: camera_link
 * /hector_mapping/pub_map_odom_transform: True
 * /hector_mapping/scan_subscriber_queue_size: 5
 * /hector_mapping/scan_topic: vrep/scan
 * /hector_mapping/tf_map_scanmatch_transform_frame_name: scanmatcher_frame
 * /hector_mapping/update_factor_free: 0.4
 * /hector_mapping/update_factor_occupied: 0.9
 * /hector_mapping/use_tf_pose_start_estimate: False
 * /hector_mapping/use_tf_scan_transformation: True
 * /rosdistro: kinetic
 * /rosversion: 1.12.13

NODES
  /
    base2camera_broadcaster (tf/static_transform_publisher)
    base2laser_broadcaster (tf/static_transform_publisher)
    face_visualizer (detection_msgs/patch_visualizer)
    facedetector (opencv_detector/detector)
    hector_mapping (hector_mapping/hector_mapping)
    map2nav_broadcaster (tf/static_transform_publisher)
    nav2base_broadcaster (tf/static_transform_publisher)
    rviz (rviz/rviz)
    world2map_broadcaster (tf/static_transform_publisher)
```

```
$ roslaunch hector.launch
```

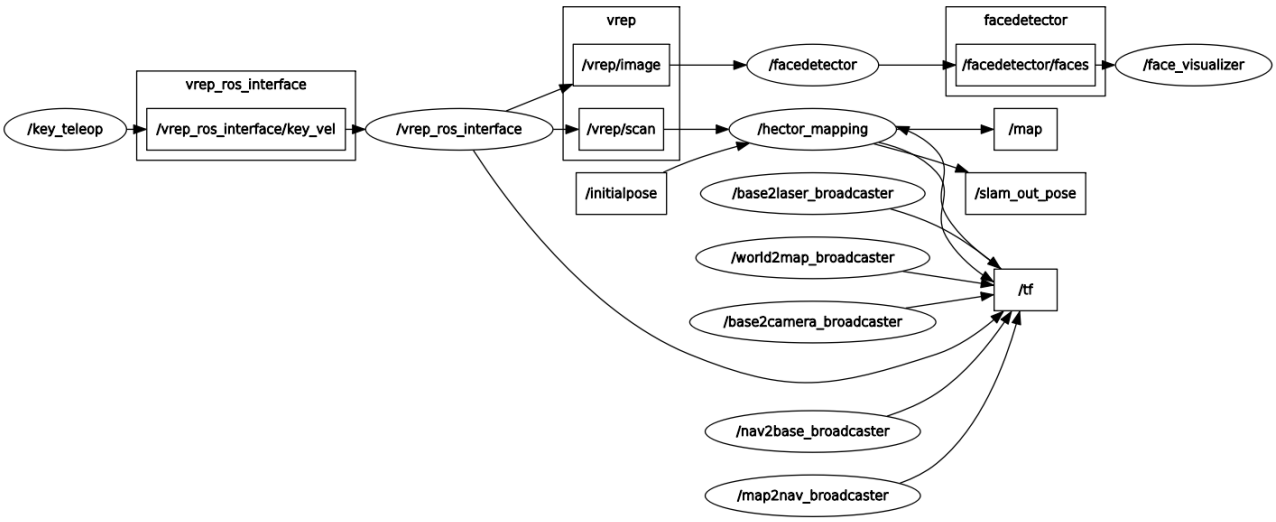
g. launch automatic ball tracking

```
$ roslaunch follow_ball.launch
```

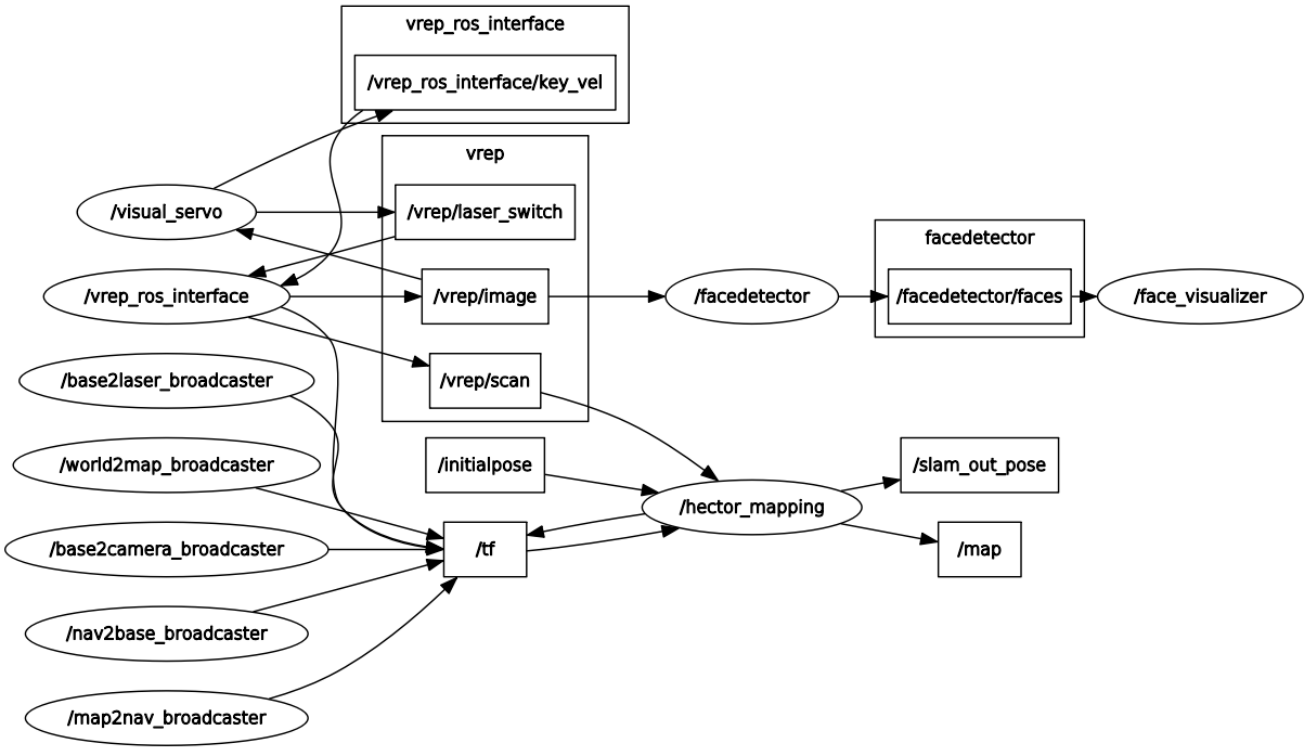
```
sq@sq-GE63VR-7RF:~$ roslaunch ./catkin_ws/src/vrep_slam/follow_ball.launch
```

3. System design and modules implement

The keyboard control mode :

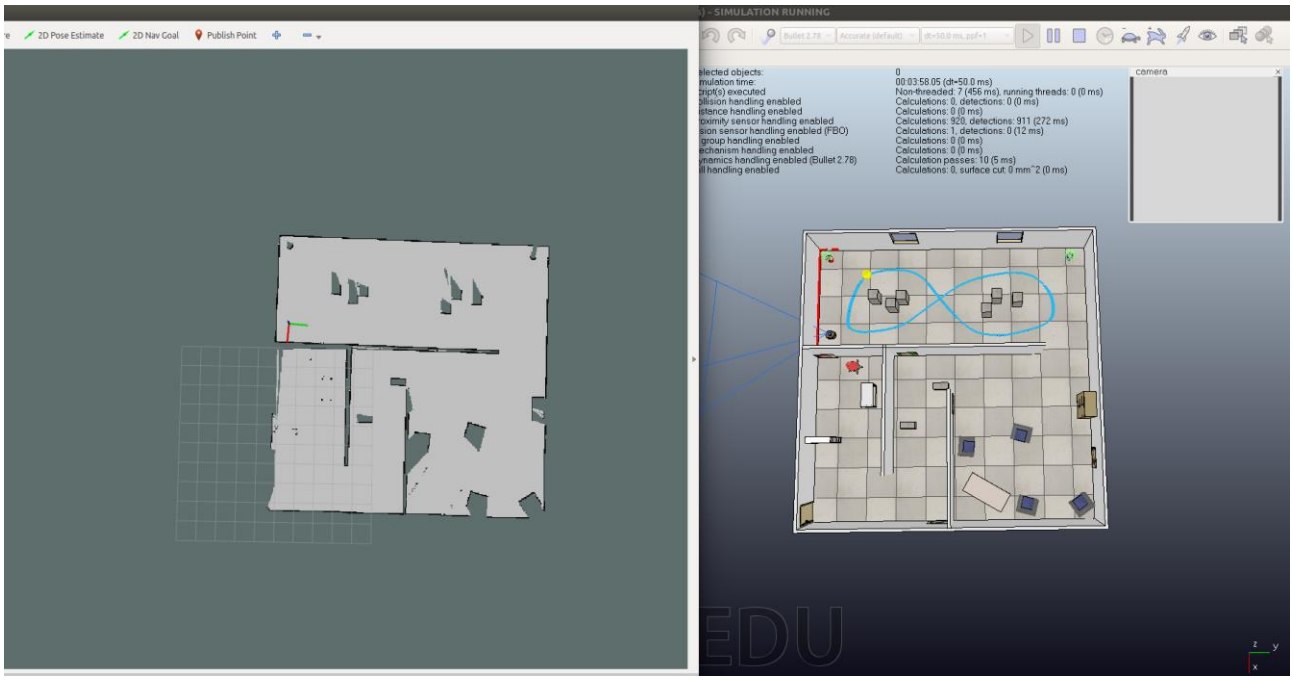


The follow ball mode :



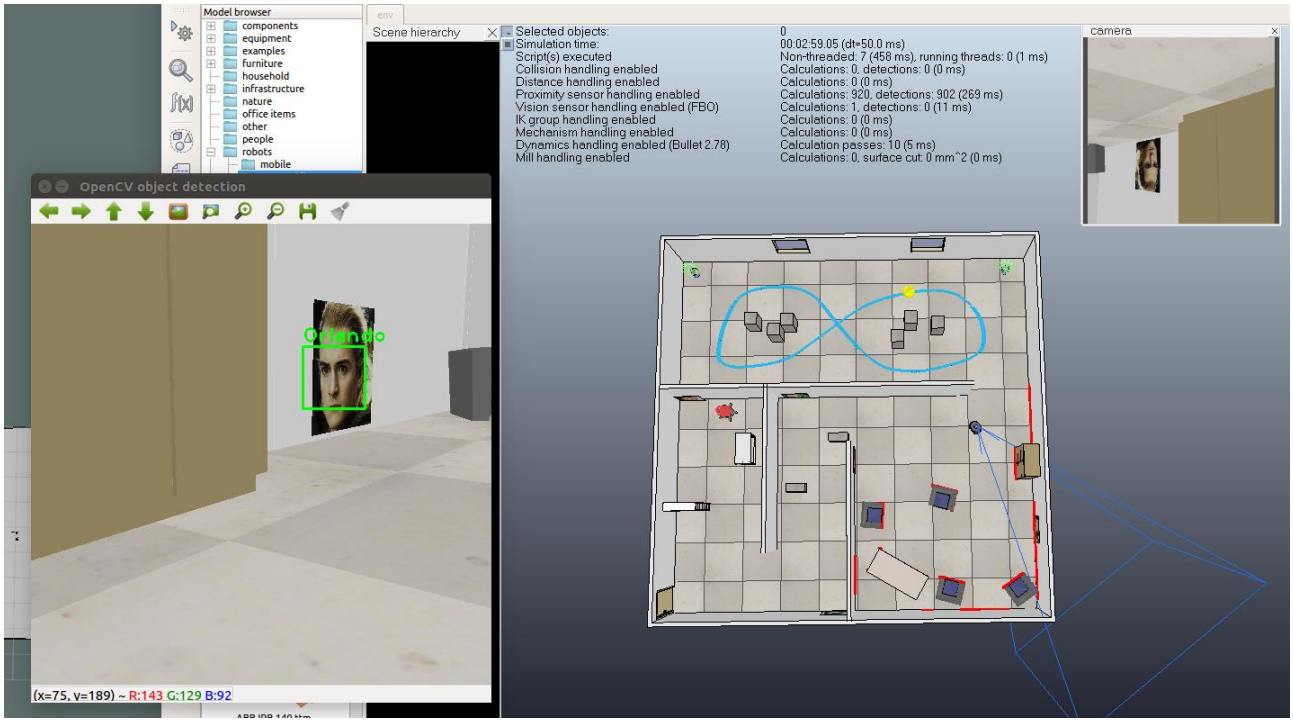
4. Tasks Results

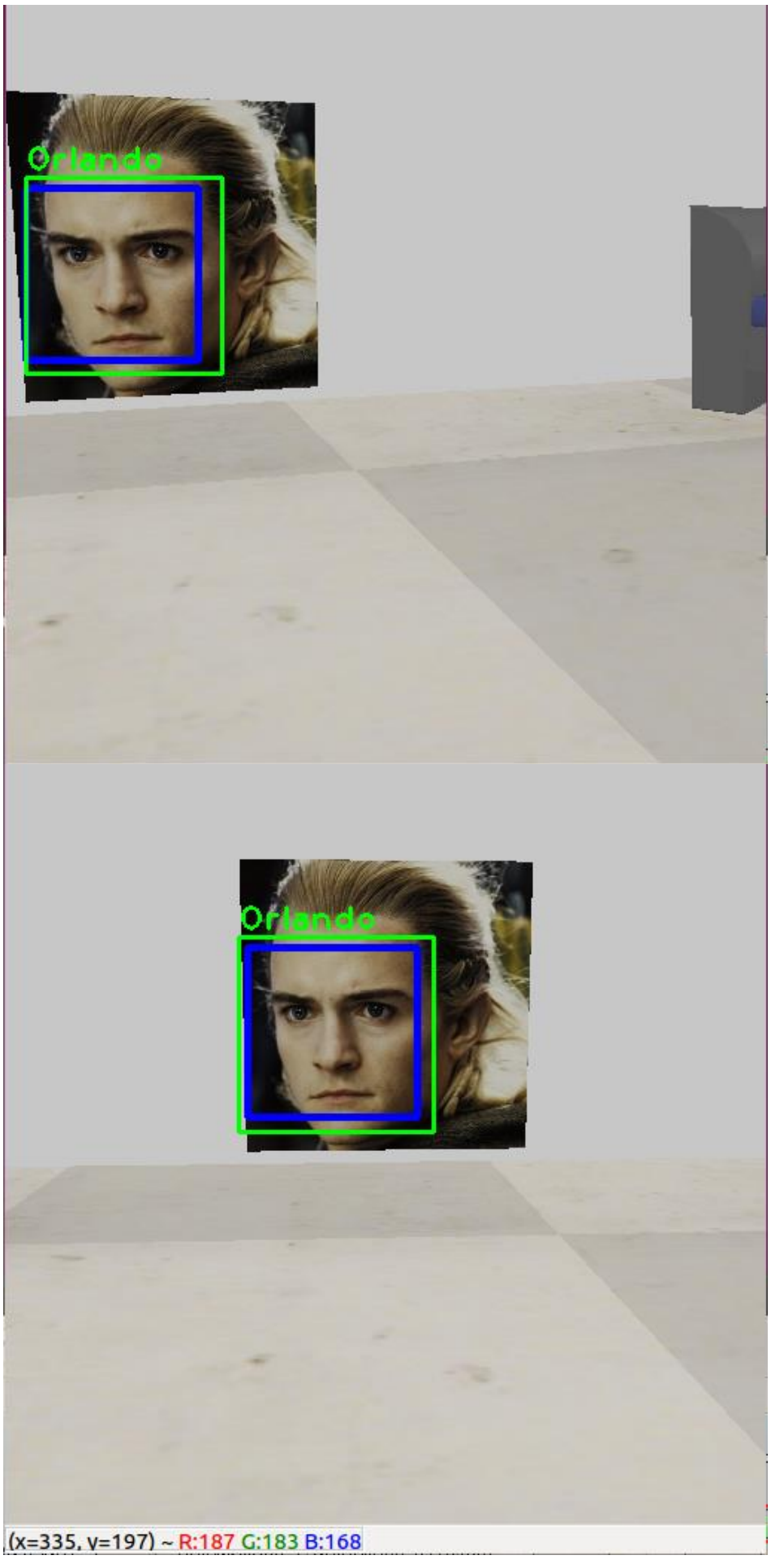
a. Keyboard control and Mapping

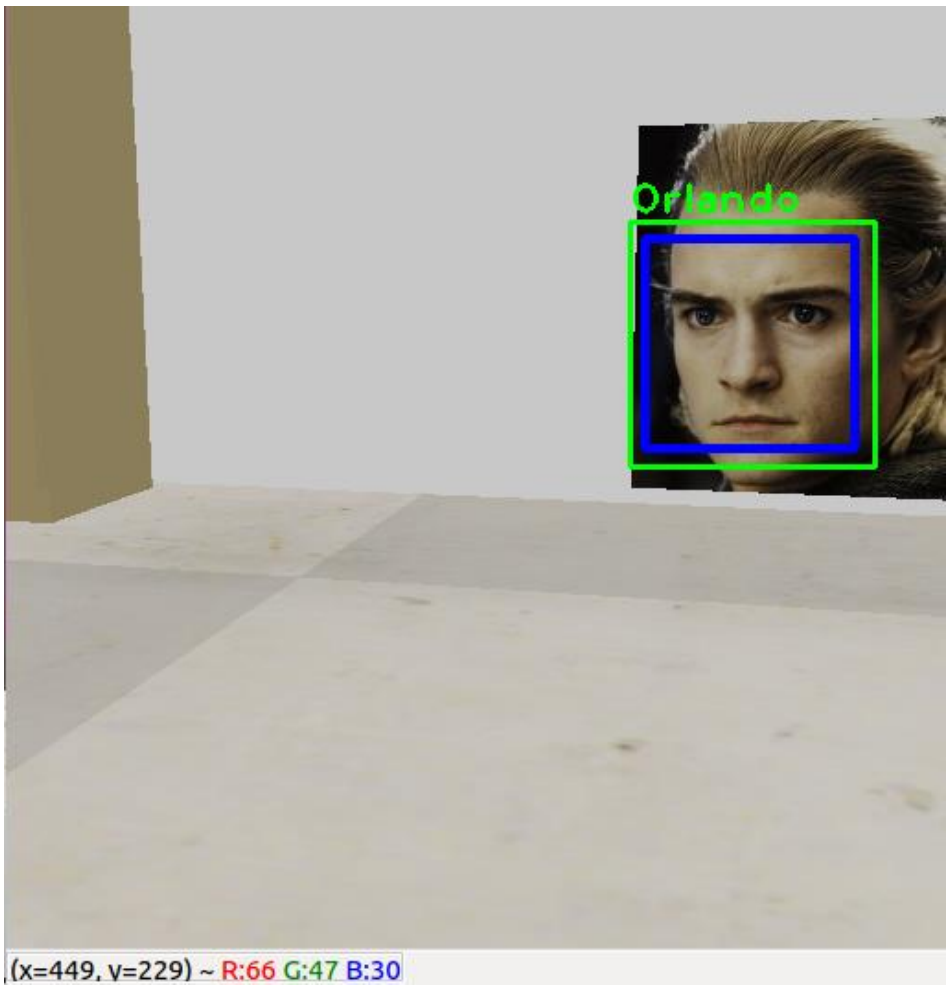


More details in attached video “key_control_and_mapping_video”

b. Face recognize

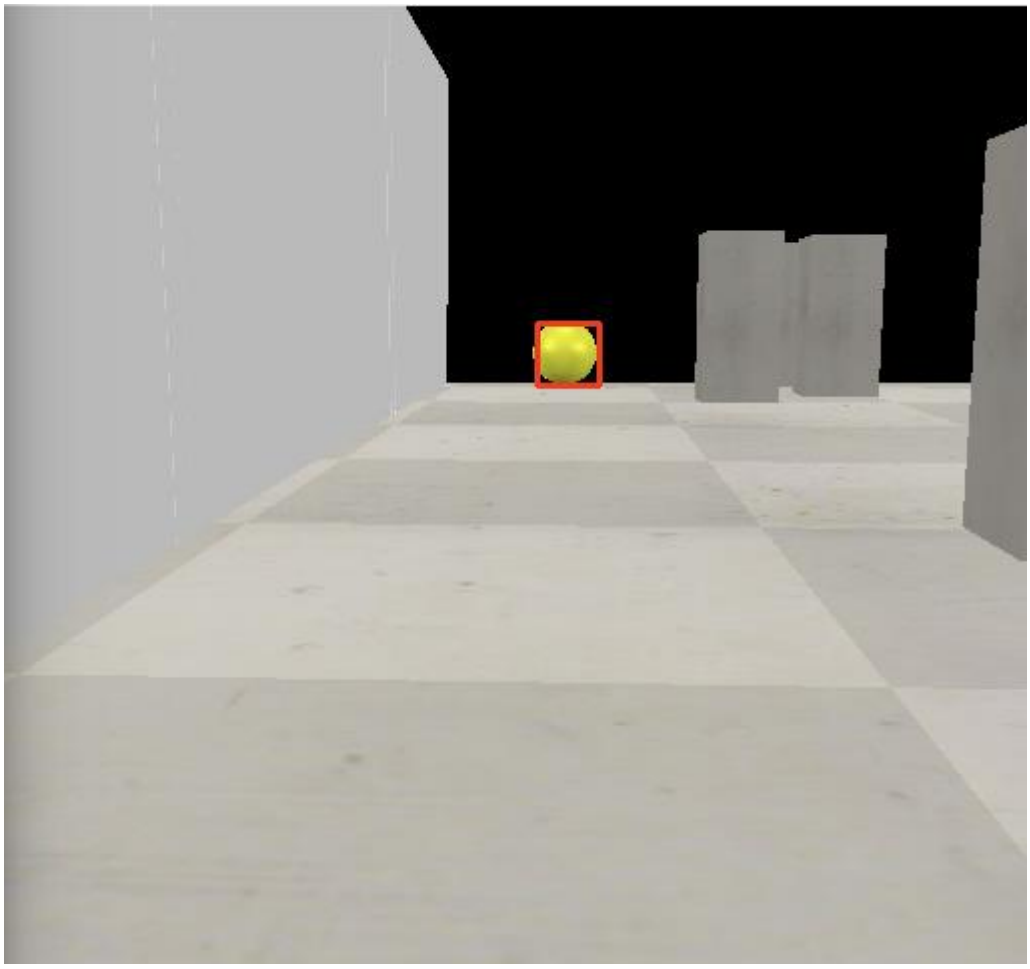



Orlando
Orlando
(x=335, y=197) ~ R:187 G:183 B:168



More details in attached video “face_recognize_video”

c. Auto follow ball



More details in attached video “auto_follow_ball_video”

4. Conclusion

1) Build 2D grid map with laserscan data avia rviz
2) Control the mobile robot in the vrep simulation environment with keyboard
3) Image Recognition and localization
4) Visual tracking (follow the yellow ball)

5. References

https://github.com/tu-darmstadt-ros-pkg/hector_slam

https://github.com/ros-teleop/teleop_tools

www.coppeliarobotics.com/helpFiles/en/rosInterf.htm