Task 1.1 – Problem Statement

Implementing Position Holding of Drone

From the tutorials, you have learned

- The basics of PID control.
- Tuning PID parameters using the ROS packages provided to you.

Scene Description:

Load the given scene *task1_1_hb.ttt* in V-REP simulator. The scene looks as shown in Figure 1:

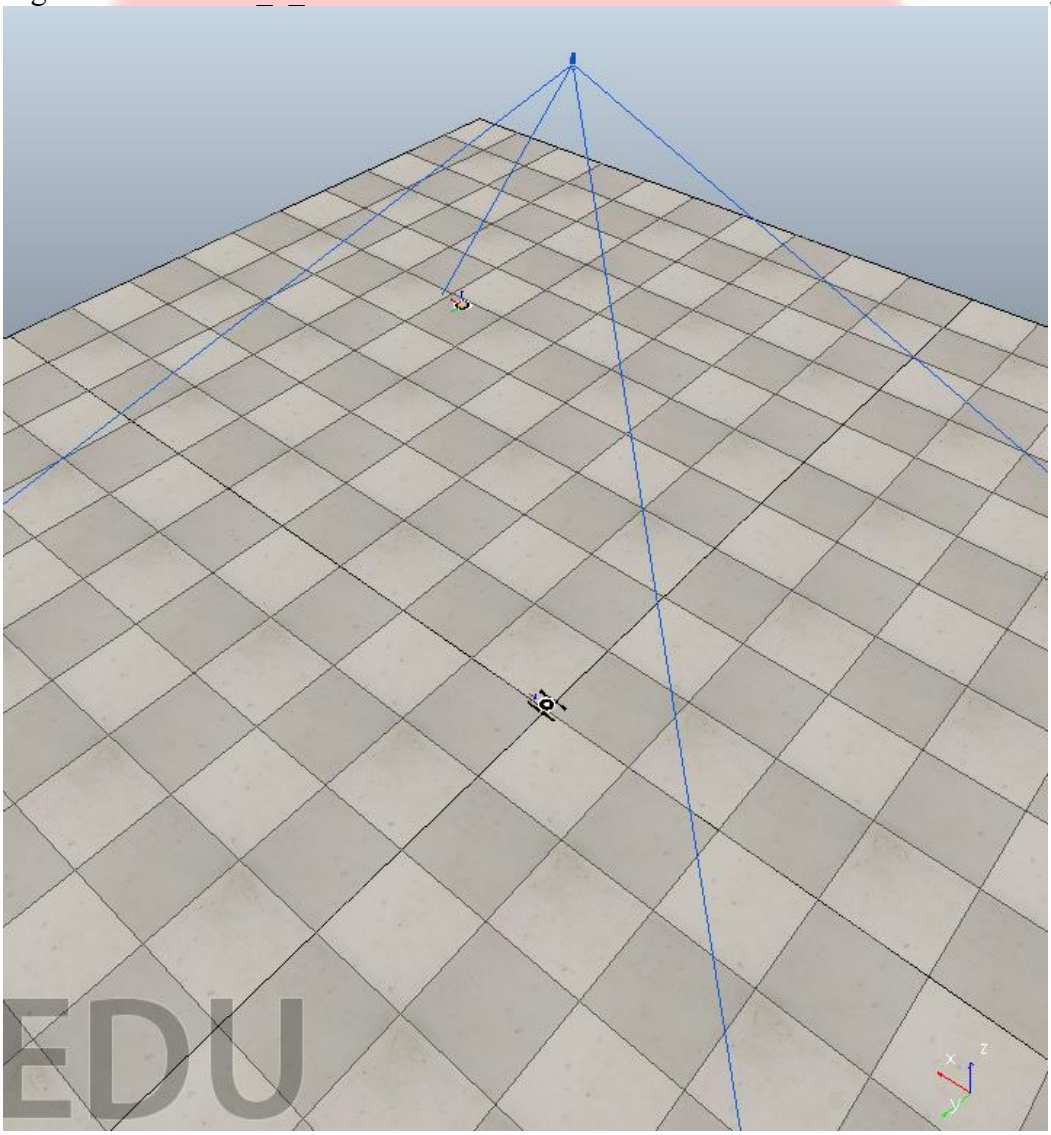

Figure 1 : task1_1_hb.ttt

Following are the various objects in the scene:

eDrone : This the the Drone model provided to you. It subscribes to topic /drone_command
Vision_sensor : This gives an image within the blue region with a resolution 1280 x 1024.
Position_to_hold : Dummy representing the position to hold.
whycon_marker : whycon_marker associated with the dummy. Double click the whycon_marker to open scene object properties and set it as renderable as shown in the figure so as to know the corresponding whycon coordinates. Note down the coordinates and uncheck renderable to prevent detection of whycon marker again.

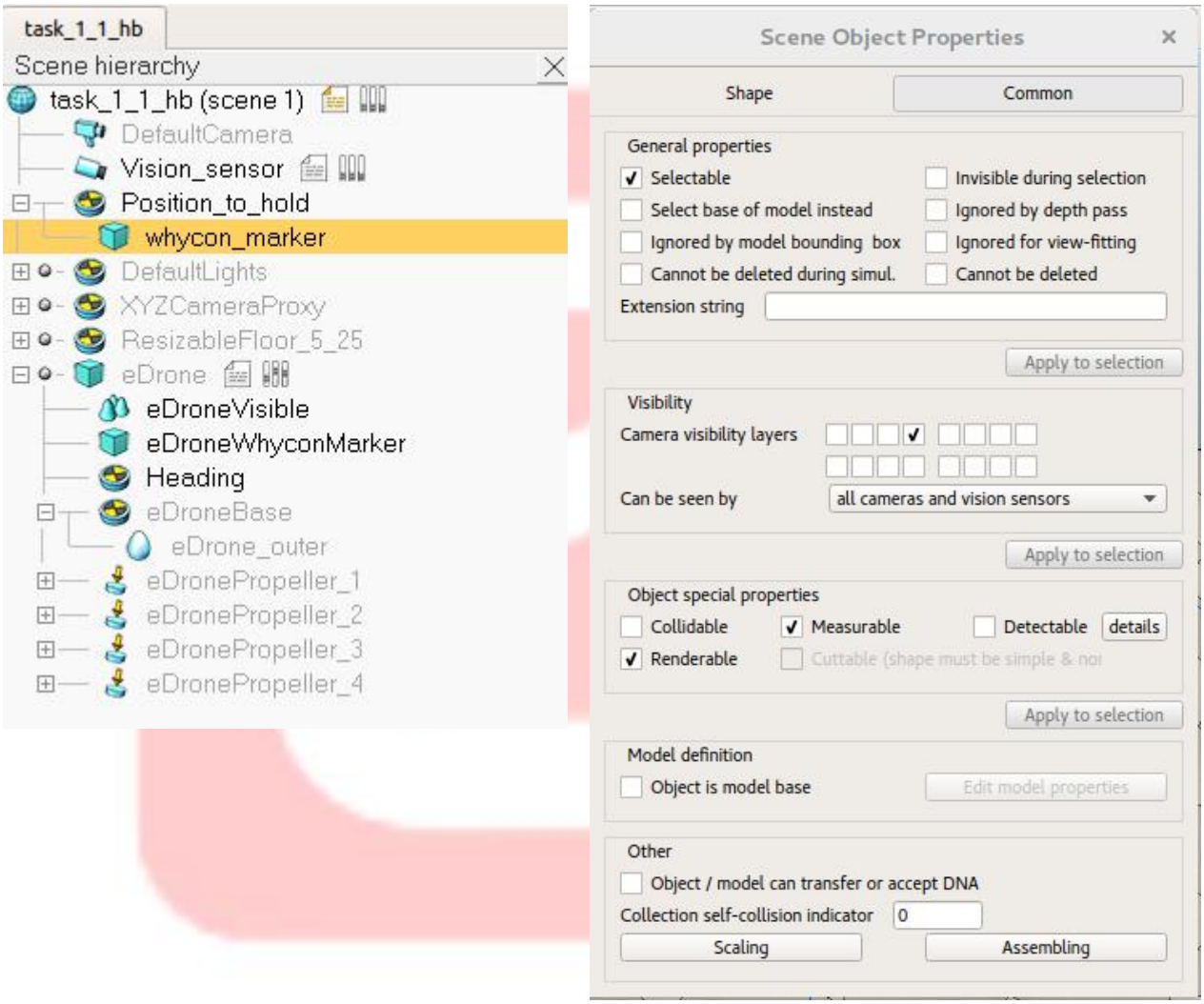

Figure 2: whycon_marker and its Scene Object Properties

Problem Statement:

Complete the python script : *position_hold.py* to command the eDrone to hold its position in vrep scene at *Position_to_hold* dummy.

Procedure:

1. Copy "**position_hold.py**" to scripts folder in the hungry_bird folder which you have cloned in the src folder of catkin_ws in the previous task. i.e. to `~/catkin_ws/src/hungry_bird/scripts`
2. Copy "**position_hold.launch**" to launch folder in the same hungry_bird folder as above i.e. to `~/catkin_ws/src/hungry_bird/launch`
   This launch file only runs the whycon node and outputs /whycon/image_out
3. Complete the "**position_hold.py"** script **to publish /drone_command** to hold the position of e-Drone. Fill the script as directed in the template.
4. Load "**task_1.1.ttt"** in V-REP simulator after launching roscore in terminal
5. Run the launch file by the following command:
   `>> roslaunch hungry_bird position_hold.launch`
6. After completing the position_hold.py file, run the python file using the following command:
   `>> rosrun hungry_bird position_hold.py`
7. This should ideally make the e-Drone model hover or hold its position at the *Position_to_hold* dummy.

Points to remember:

- Make sure you watch the video tutorials(3 parts) mentioned in "*Getting_familiar_with_PID_control.pdf*" before proceeding with the Task.
- Make sure you make the "eDroneWhyconMarker" as Renderable and "whycon_marker" as unrenderable (Refer Figure 2).
- Run "*rosmsg show topic_type*" to see what is the message type of the corresponding topic.

Rules:

- The following simulation settings are default and **should not be changed**.
  - Dynamics engine : Bullet 2.78
  - Dynamics settings : Accurate (default)
  - Simulation time step : dt = 50 ms (default)
  - Real-time mode: **Enabled**
- You are not permitted to make changes in the scene to be submitted, i.e. task_1_1_hb.ttt.

Submission Instructions:

Follow the instructions below to submit your Task.

1. Bag File:

a) Launch your *position_hold.launch* file after running simulation in V-REP:

```
>> roslaunch hungry_bird position_hold.launch
```

b) Run your python script, *position_hold.py* by running the following command in another terminal:

```
>> rosrun hungry_bird position_hold.py
```

c) Run the rosbag command to record the /whycon/poses. The following command records the topics */whycon/poses* for a duration of 20s. **This command should be executed only after you start the simulation and the launch file but before running the python file.**

```
>> rosbag record whycon/poses --duration=20s --chunksize=10
```

NOTE: The duration parameter is set to 20s in the above command. This has to be set as per the estimated time your code takes to complete. For example, if it takes approximately 30 seconds for your code to complete the task, you must change the duration to 30 seconds. Maximum allowed duration is **120s ONLY.**

d) Compress the bag file

```
>> rosbag compress -j bag_file_name.bag
```

e) This will compress the *bag file* and now the size will be below 5mb approximately. Look the size of the bag file in its properties to distinguish between the original and the compressed bag file.

f) Rename the **compressed bag file** as **<team_id>_task_1_1.bag** For eg: if your Team ID is HB#105, rename it as 105_task_1_1.bag

NOTE: Don't follow the same procedure as in the demo video(Since we have asked to move the drone in the demo video) while recording bag file. Just run the simulator, launch the file and run the python script. Record the rosbag till it become stable at the destined position.

2. Code:

a) After completing your python script, position_hold.py, rename it as **<team_id>_task_1_1.py**. For eg: if your Team ID is HB#105, then rename it as 105_task_1_1.py

Compress these files into a .zip file before uploading. Name the .zip file as your <team_id>_task_1_1.zip. For eg: if your Team ID is HB#105, then rename it as 105_task_1_1.zip

3. Video:

a) Upon verifying that your task is complete, record a maximum 2 minute video using a screen recorder like simplescreen recorder or kazam.
b) The video must be as follows:
i. Team Slide –All members' details in a slide.
ii. Any One member of the team, running the scripts and launch files in terminal.Output of V-REP and whycon image_out window on the PC screen captured clearly.
c) Please record the video as shown in this demo.

The video should not be edited in any manner. Teams uploading an edited video will be disqualified from the competition. e-Yantra reserves the rights to disqualify any team if any foul play is suspected.

Uploading video/s on YouTube:

● Upload a one-shot continuous video with the title eYRC#HB#Task1_1#<TeamID>
(For example: If your team ID is 1234 then, save it as eYRC#HB#Task1_1#1234)
● Please note that while uploading the video on YouTube select the privacy setting option as Unlisted as shown in Figure 2. You need to upload the video as instructed on the portal.

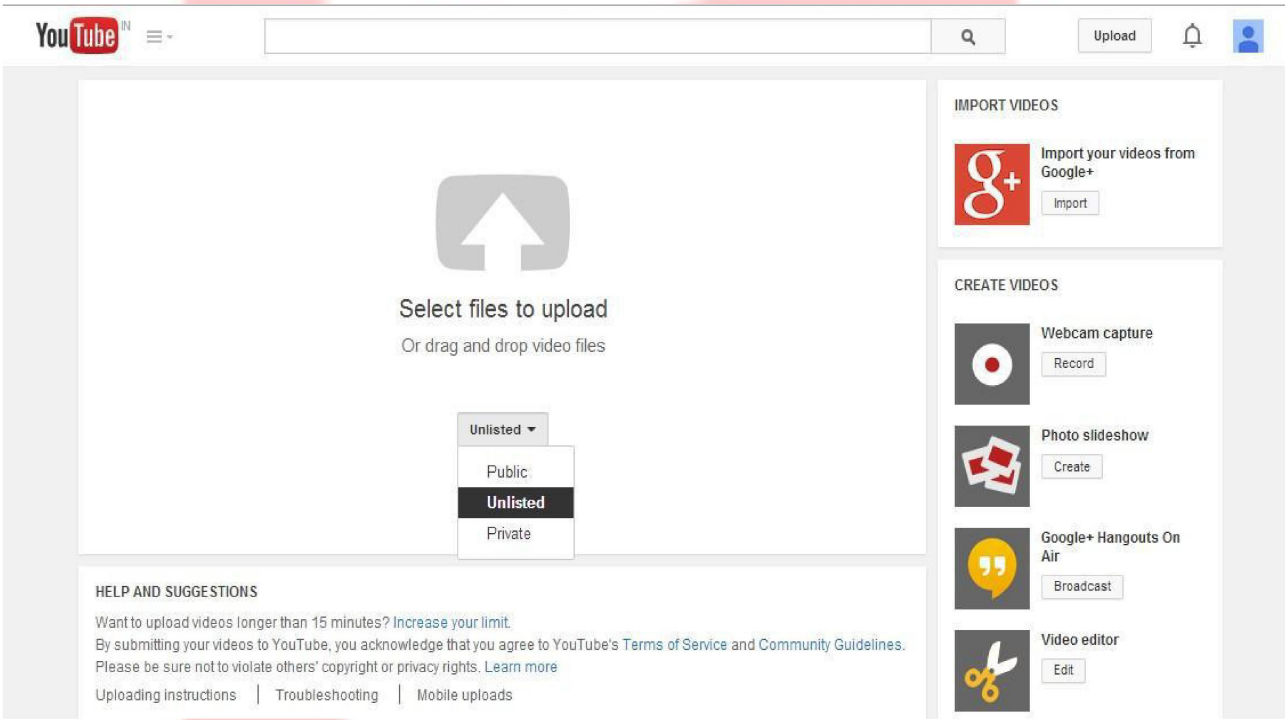

Figure: Uploading unlisted video on YouTube

NOTE: **You must upload all of the following: (i) rosbag file, (ii) Python code (iii) Video in order to be evaluated. Please place files (i) and (ii) inside a .zip file before uploading. Name the .zip file as your <team_id>_task_1_1.zip. Please follow the naming convention strictly as specified in each step. For eg: if your Team ID is HB#105, then rename it as 105_task_1_1.zip.**

Instructions for uploading the folder will be provided on the portal.

NOTE: **Deadline for the Task 1.1 submission is on 14th November 11:59 pm. You will incur penalty if submission is delayed. Details regarding penalty are provided on the portal.**