PROJECT REPORT

On
Vision Assisted Pure Pursuit based Landing For Micro Aerial Vehicles

For the Fulfillment of Summer Internship
(2021-22)

By

YASH JANGIR

Summer Research Intern, Intelligent Systems Group, CSIR-CEERI

Scientist-in-charge: **Mr. Kaushal Kishore**

ACKNOWLEDGEMENTS

I would like to thank Mr. Kaushal Kishore for providing this excellent opportunity to pursue a Summer Research Internship at CSIR-CEERI, Pilani. I would also like to express our gratitude to all those people whose efforts have made possible our professional experience at CSIR-CEERI, Pilani, a success even in this remote work from home mode during the pandemic.

I would like to thank Mr. Mahammad Irfan and Mr. Sagar Dalai and all the personnel of CSIR-CEERI, for facilitating the process and guiding us whenever needed.

Last but not the least I would like to thank everyone from Intelligent Systems Group, CSIR-CEERI for their immense support throughout the project.

Table Of Contents

Introduction

Aerial vehicles like quadrotors, have shown to be very useful in many applications like mapping, object delivery, tracking etc. A key feature for these applications is that the vehicle needs to perform these tasks autonomously. Autonomous landing of an unmanned aerial vehicle or a drone is a challenging problem in aerial vehicles. This problem has been tried to solve by combining multiple sensors such as global positioning system (GPS) receivers, inertial measurement unit, and multiple camera systems. Although these approaches successfully estimate an UAVs location , many calibration processes and filtering techniques are required to achieve good landing accuracy. In other cases where UAVs operate in heterogeneous environments with no GPS signal should be considered. In mapping and object delivery applications, the vehicle landing area is constrained, and hence requires precision landing. External disturbances like winds and visibility add another challenge in actual landing scenarios. In this work, the guidance technique of pure pursuit is adopted and modified for quadrotors. This work focuses on the development of vision-assisted guidance techniques for landing of a quadrotor. The concern of a guidance strategy is to enable persistent tracking of the landing pad or track a specific landing trajectory ending to a landing pad which is stationary and its position is being estimated by a vision based method using a single downward facing camera to accurately land on the target.

Background Research

PID Controller

The proportional-integral-derivative (PID) [1]controller is the most widely used feedback design. A high level PID controller is designed for maneuvering the quadrotor to trajectory points.

$$u(t) = K_p e(t) + K_I \int_0^t e(t)dt + K_D \frac{de(t)}{dt},$$

where KP is the proportional gain, KI is the integral gain and KD is the differential gain. e(t) is the calculated error to the target state, given the current state.

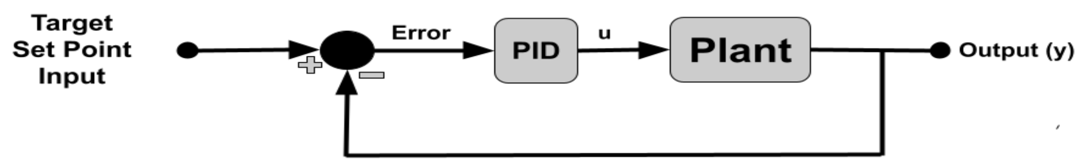

Fig. A Closed Loop PID controller

Pose Estimation using Fiducial Tags (ArUco Markers)

Pose estimation is of great importance in many computer vision applications, robot navigation, augmented reality, and many more. This process is based on finding correspondences between points in the real environment and their 2d image projection is usually difficult, thus it is common to use fiducial markers to make it easier. Fiducial marker systems consist of patterns that are placed in the environment and are automatically detected with a camera appropriate for the marker detection algorithm. Multiple tags have been developed for different applications[2]. In our application we are using AR tags to detect a precision estimate of the target landing site. The aruco module is based on the ArUco library, a popular library for detection of square fiducial markers developed by Rafael Muñoz and Sergio Garrido.

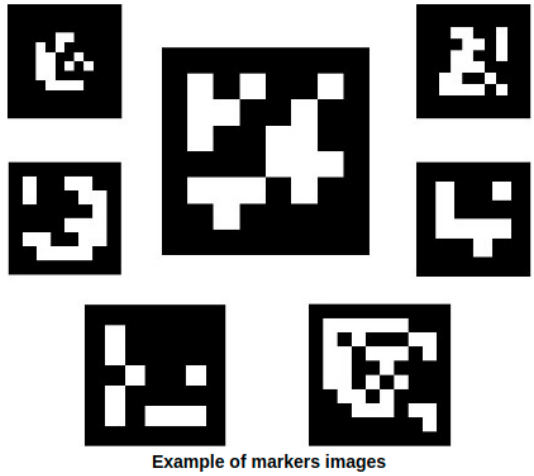

Fig. Example of Markers (source: ArUco library)

Classical Pure Pursuit

Pure-pursuit was used for the pursuit of missiles to a target, where the missile velocity vector was always directed toward the instantaneous target position. Wallace et al in 1985 [3] developed a pure-pursuit strategy for mobile robots, where they calculated the steering angle for a smooth trajectory with some look ahead distance for the path to be tracked. With this approach any vehicle on ground was able to track multiple trajectories and smoothly navigate. Refer Wallace et al in 1985 [3] for more information.

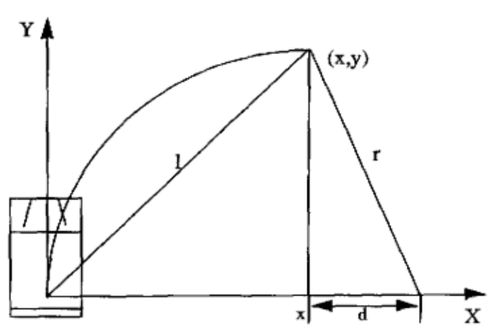

Fig. Showing the Trajectory Generation for the ground vehicles using Pure pursuit (source: Wallace et al in 1985 [3])

Methodology

Modified Pure Pursuit for Quadrotors

In our method we have a 3D space with a goal point and a feedback of odometry. We know the goal pose for the drone landing site. A simple point to point maneuvering PID controller for the drone is used to maneuver the quadrotor to the points generated by high level Pure pursuit trajectory tracking algorithm.

In initial conditions, assuming that the UAV is at some point A(Xa,Ya,Za) in space and origin is the landing site. We need to calculate a smooth trajectory for the landing. We pursue the direct goal point on every step to generate new corrected trajectories. With each step the quadrotor descends down a fixed distance and follows a smooth trajectory.

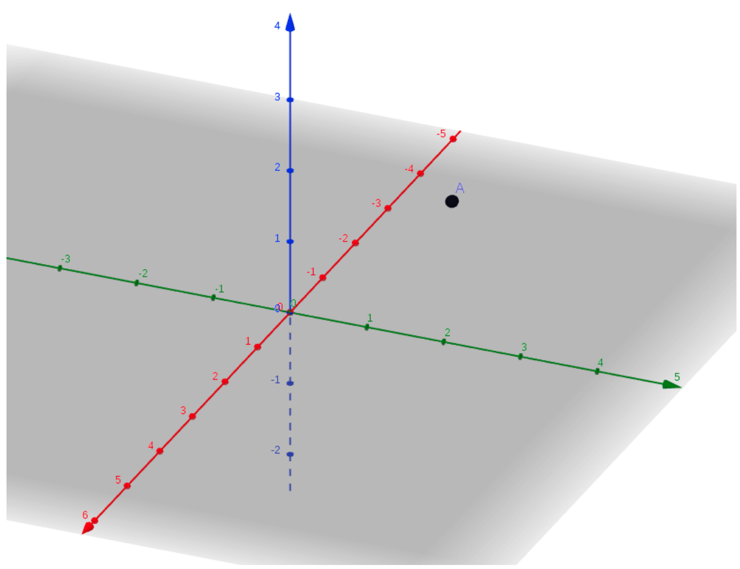

Fig. Point A in the space is the initial position of the drone

We calculate the euclidean distance of the drone from the landing site by :

$$d = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2 + (z_1 - z_0)^2}$$

We convert the 3d space in a 2d problem rotating the X-Z plane with Z as axis of rotation with an angle theta we define this angle as angle of trajectory projection so that the plane passes through the point A(Xa,Ya,Za) the angle is calculated by:

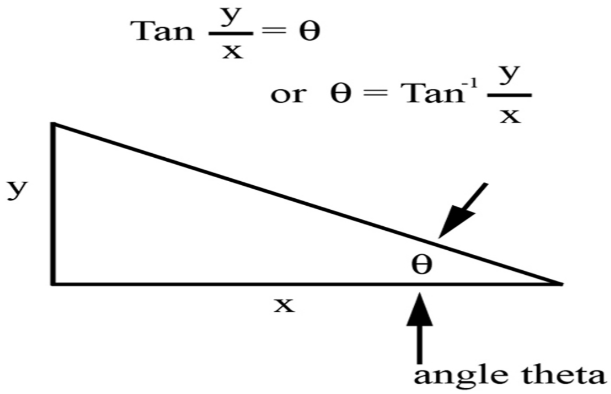

Fig. Calculation of angle of trajectory projection

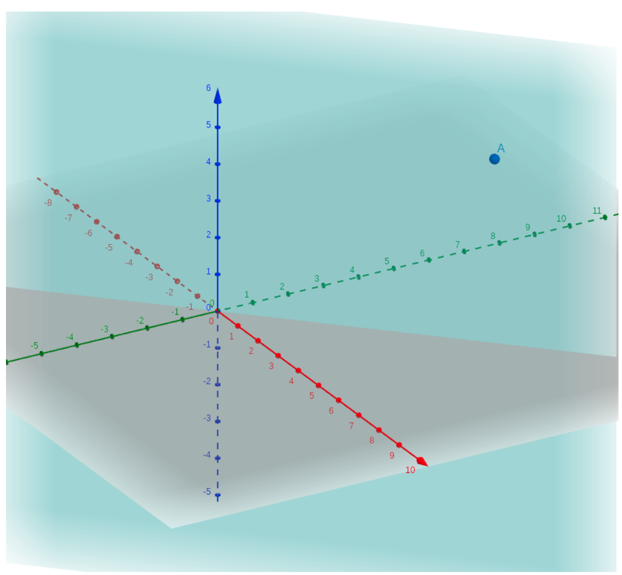

Fig. S-Z plane

Now we have reduced the problem to a 2d cartesian System defined as the S-Z plane the trajectory generated will line in this plane for the pursuit of the goal. We follow the classical method of Pure Pursuit to generate arc trajectories to the go at each step.

We calculate an arc trajectory at every step.

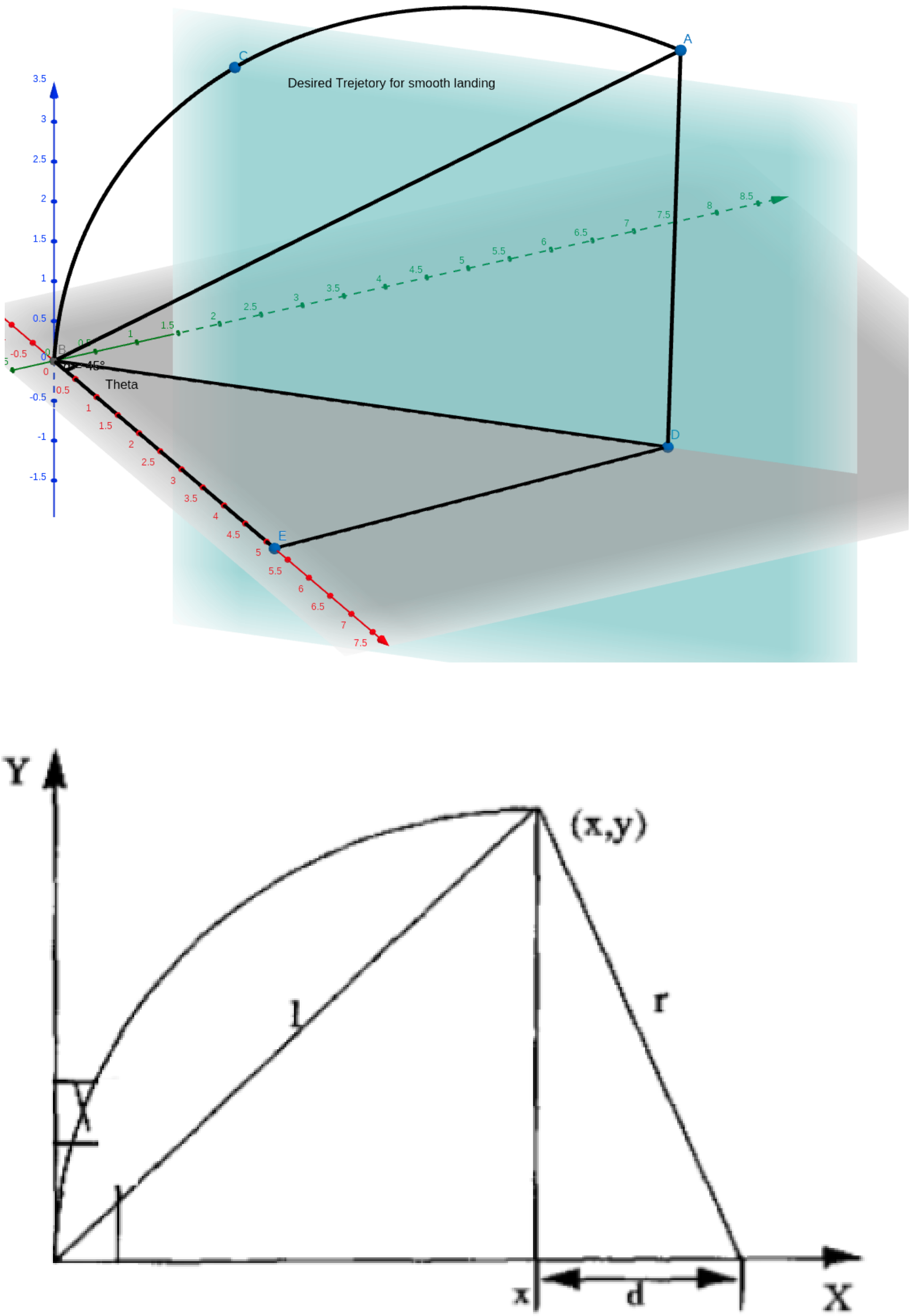

Fig. showing comparison of the 2D and 3D space model (source: Wallace et al in 1985[3])

$$x^2 + y^2 = l^2$$

$$x + d = r$$

$$d = r - x$$

$$(r - x)^2 + y^2 = r^2$$

$$r^2 - 2rx + x^2 + y^2 = r^2$$

$$2rx = l^2$$

$$r = \frac{l^2}{2x}$$

Fig. Equations for trajectory calculation(source: Wallace et al in 1985[3])

Hence we are able to calculate a smooth trajectory for landing using the radius and two boundary points.

We parametrize the altitude and define a fixed height for each step descend on the next trajectory. Using the equation we are able to calculate the coordinates to the next look ahead goal and we project those coordinates to the X-Z and Y-Z planes to get the X and Y coordinates of the next goal.

Hence for each step we are able to find the next point in 3D Trajectory using these simple calculations.

Implementation

Simulation Environment

We use the Gazebo simulator to perform evaluations of our approach. Gazebo enables us to perform realistic simulations using superior rendering and an accurate physics engine. Varied parameters such as illumination in the environment, landing sites, surface against which the landing site is simulated and the number of objects present in the vision cone are simulated. We use a typical Iris drone with a monocular downward facing camera.

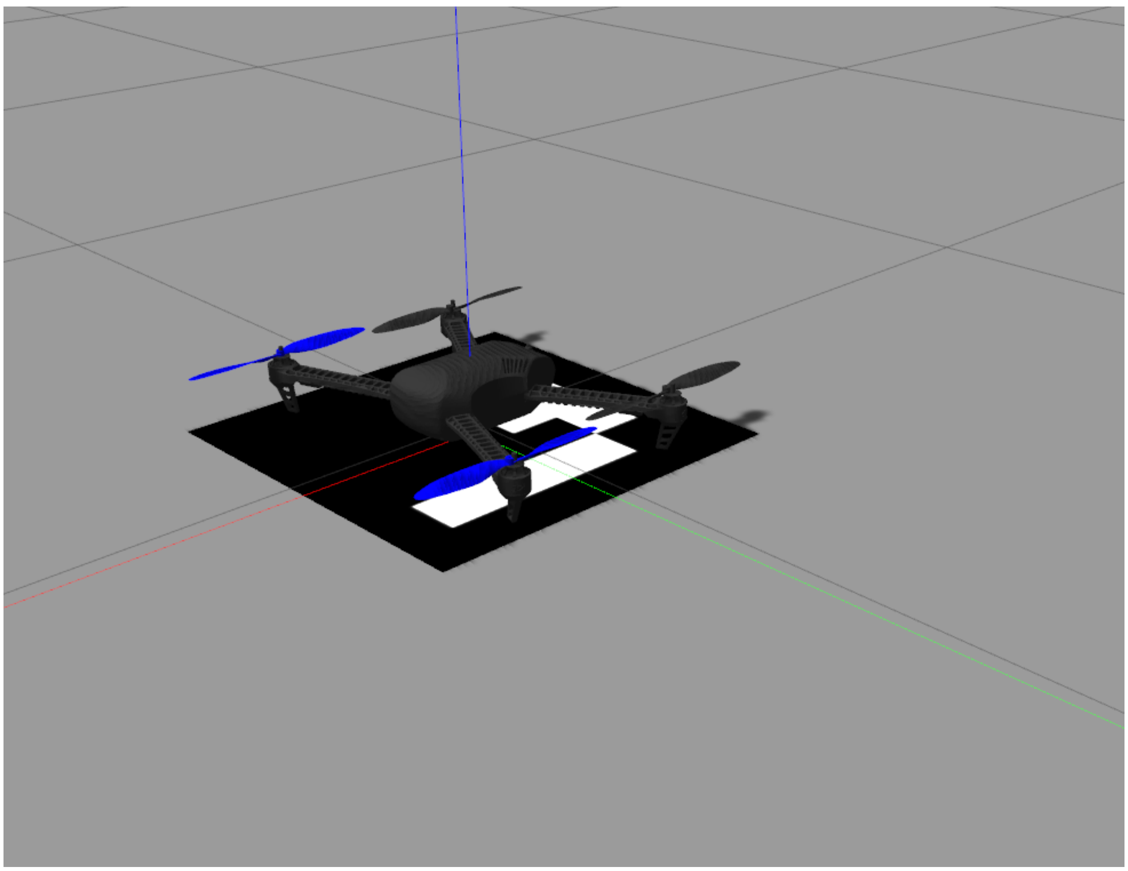

Fig. Gazebo Simulation setup

Camera Calibration

In general approaches, distortion is not considered in simulations. However, to test the complete effectiveness of the algorithm.

The camera attached has been calibrated using a chessboard using the built in Camera Calibration Package in ROS.We used a custom made script to hover the drone and spawned a chessboard in the gazebo based environment. The figure below shows the calibration simulations for the camera.

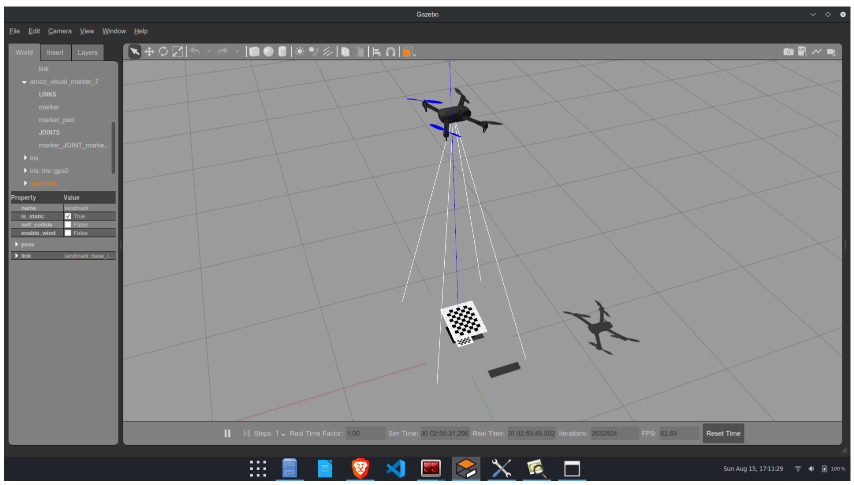

Fig . Calibration setup for Iris Monocular Cam

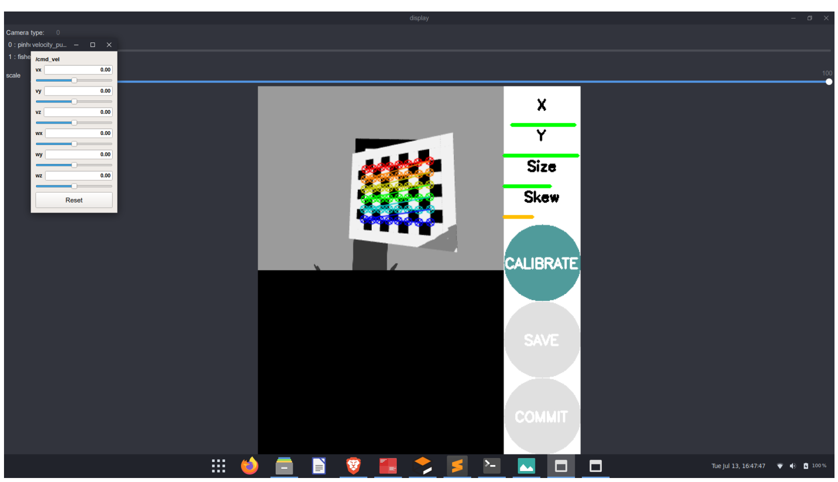

Fig. Camera Calibration using Built in Package

Pose Estimation

Once the camera is calibrated we use OpenCV's ARuco tag detection and Pose estimation Package to estimate the pose of the tag when it is detected in the frame of the camera. We receive a Pose message which is the pose of the UAV with respect to the tag.

For our application and testing we have placed the tag on the global origin. In case we want to use this on any point in the space we use a simple transformation of the coordinate system when it is detected to get the coordinates of the tag in the global frame.

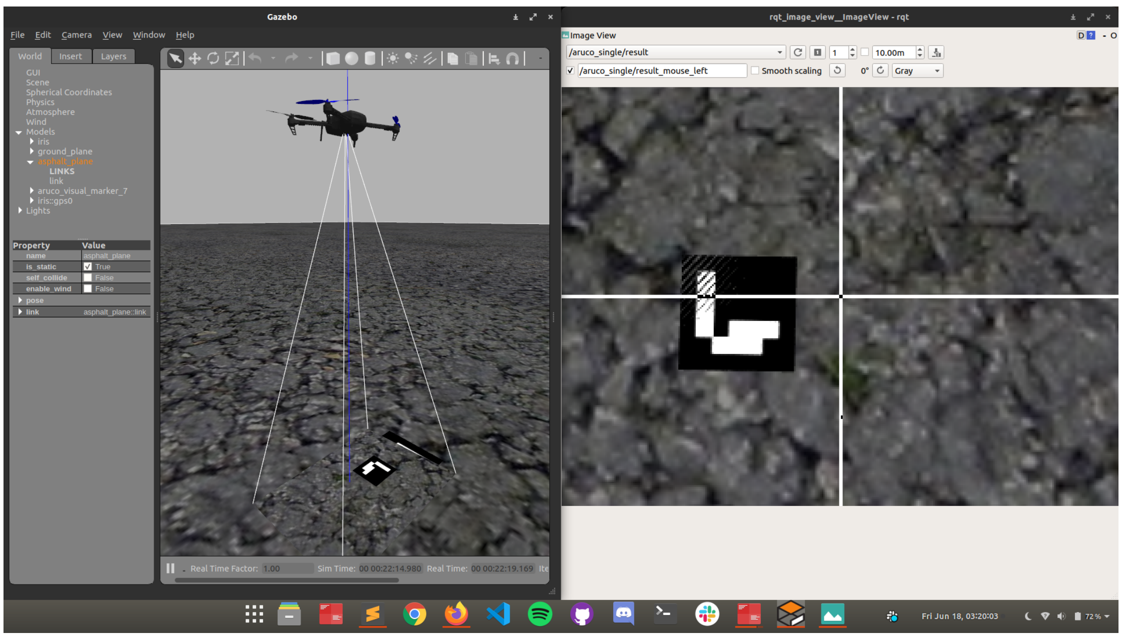

Fig. Pose Estimation Using OpenCV

Drone maneuvering and Landing

When the landing sequence is started a custom the landing package as soon as the camera detects the landing sites starts the process of landing. We test the landing on multiple scenarios. WE use the PX4 framework for the automation of the low level controls and use a high level closed loop controller for the maneuvering of the UAV.

Figure below shows the Drone landing on to the target site.

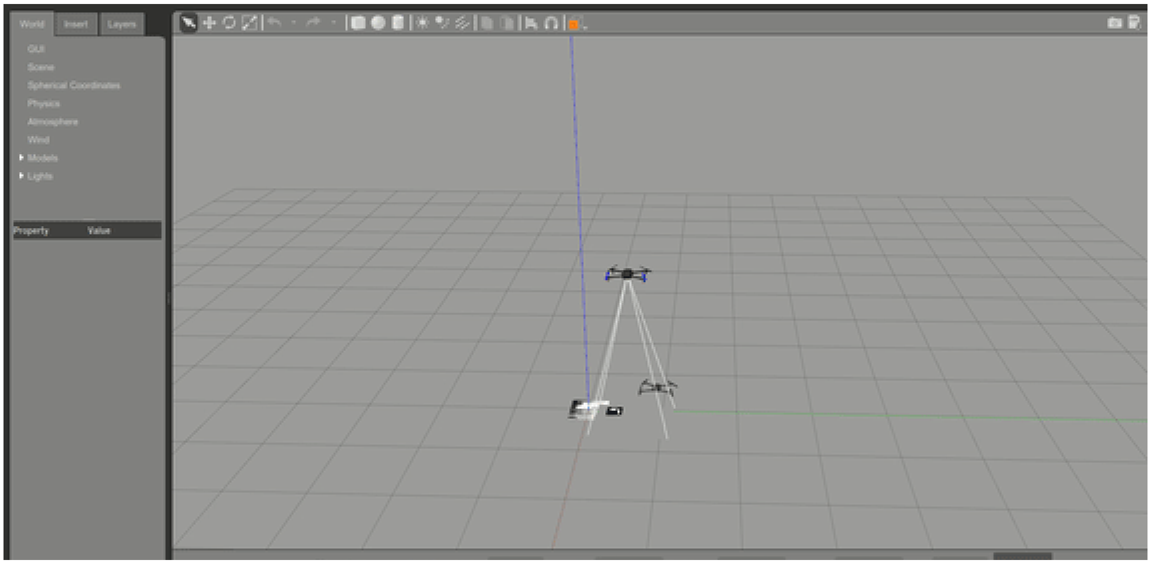

Fig. Landing of the UAV using out controller

Evaluation Studies

Landing is a very sensitive action to perform, directly maneuvering the drone to the land site without generating a favoured trajectory leads to very rash landing.Our controller was able to generate smooth trajectories and landed the drone without generating any high jerk or snap. Below is a plot of the landing trajectory that was recorded during experimentation.

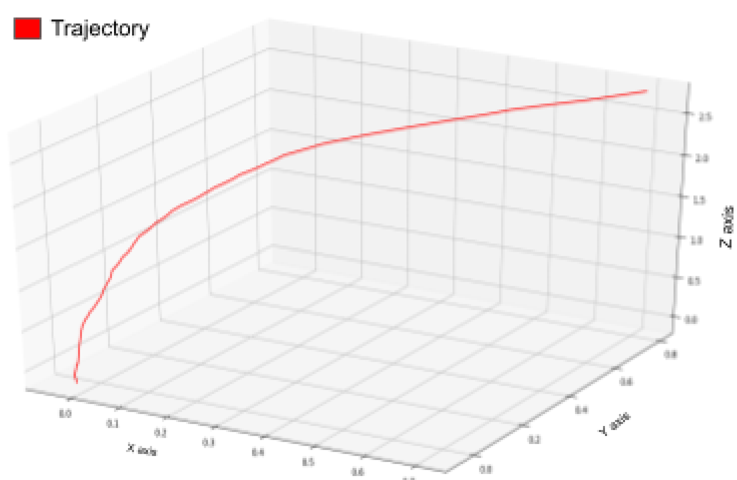

Fig. Trajectory of the landing in case of the Pure Pursuit based controller

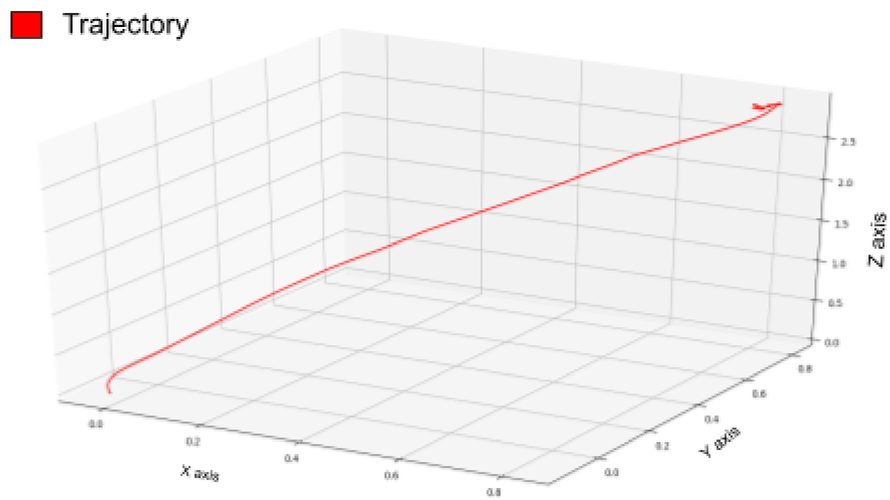

Fig. Trajectory of Landing using a simple PID controller

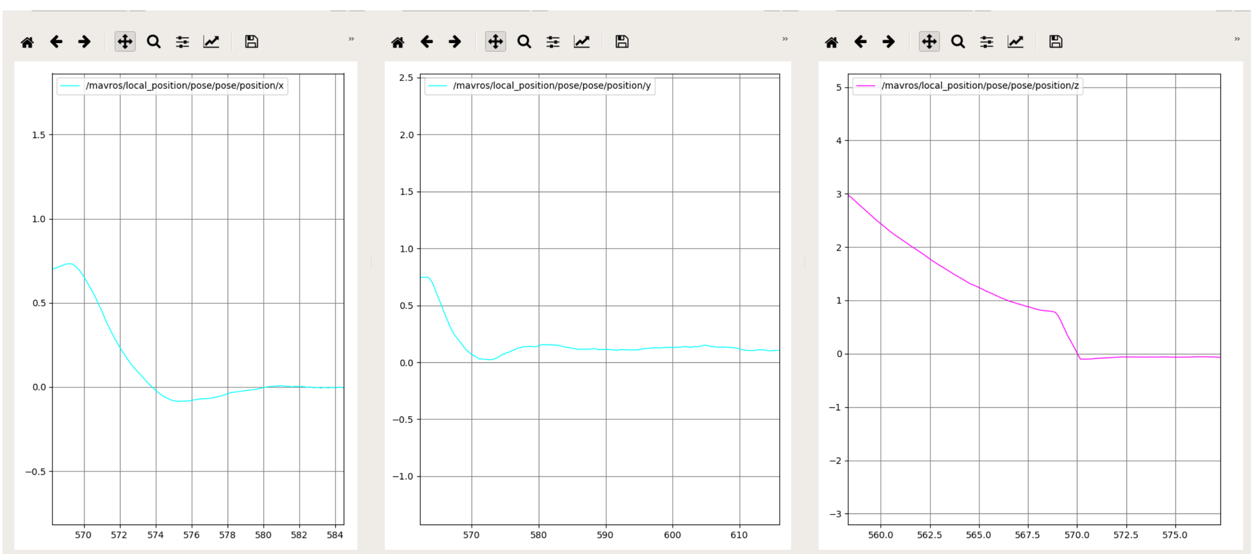

Fig. Graph of individual X, Y, Z with respect to time for the Pure Pursuit based controller

Future Work

There are a lot of future ways to continue this work. Right now the trajectory that has been generated is a simple arc, In real world applications or obstacle avoidance we can generate minimum snap and jerk trajectories to the goal given a map of environment, this will help us to create much better trajectories and then to follow the trajectories using pure pursuit based controller to have better landing.

Other future prospects include landing the drone in an unknown environment with multiple obstacles. This includes generating trajectory while mapping the environment to land. For example, having four to five obstacles around the landing site the drone needs to generate safe trajectories and land using our pure pursuit based controller.

Citations

[1] Bora Erginer and Erdinç Altuğ, “Modeling and PD Control of a Quadrotor VTOL Vehicle”, 2007 IEEE Intelligent Vehicles Symposium Istanbul, Turkey, June 13-15, 2007

[2] Artur Sagitov1 , Ksenia Shabalina1 , Leysan Sabirova1 , Hongbing Li2 and Evgeni Magid, “ARTag, AprilTag and CALTag Fiducial Marker Systems: Comparison in a Presence of Partial Marker Occlusion and Rotation”, ICINCO 2017 - 14th International Conference on Informatics in Control, Automation and Robotics

[3] R. Craig Coulter, “Implementation of the Pure Pursuit Path tracking Algorithm”, 1992

Bibliography

- “Open Source Autopilot for Drone Developers - PX4”
- “Robot Operating System - ROS”
- “Gazebo Simulator”