Simulation of a Feature-based Algorithm for the Ship Deck Landing System

Himanshu Mittal (2330478)
Mentor:-Dr. Abhishek
Helicopter Laboratory, Department of Aerospace Engineering IIT Kanpur

Holybro X500 + Pixhawk4 Build

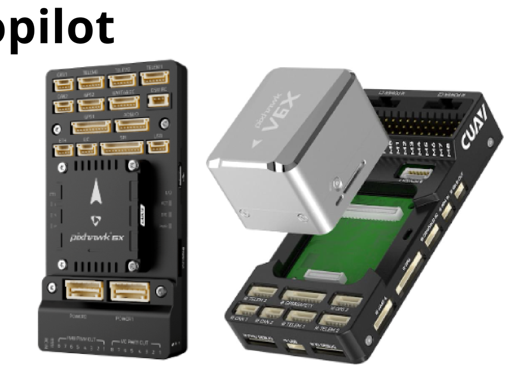
Introduction

Ship deck landings pose unique challenges due to dynamic environments, limited landing space, and the need for precise positioning and control. The proposed algorithm aims to enhance the accuracy and reliability of the UAVs landing procedure by leveraging advanced computer vision techniques and feature extraction methodologies.

Basic Algorithm Design

The algorithm follows a multi-step approach. Initially, the UAV captures images or video frames of the ship deck using its onboard camera system. These images are then processed to extract relevant features such as ship deck edges, landing markings corner detection, and template matching are employed to identify and locate these critical elements.

PX4-Autopilot



Literature Review

Papers from IEEE 'Toward Visual Autonomous Ship Board Landing of a VTOL UAVs- by Jose Luis Sanchez-Lopez, Srikanth Saripalli, 'Quadrotor Autonomous Landing on Moving Platform' by Pengyu Wang, Max Q.-H. Meng and NAVAIR papers on Image-Based Navigation for Shipboard Landing – by Doug Duehring and Avinash Gandhe are studied

Methodology and Software structure

For testing our Algorithm we proceeded with first making its simulation. For this, a software structure needs to be established which was done using:-

- **Ubuntu 20.04 Interface**
- **PX4-Autopilot**
- **ROS Noetic**
- **Gazebo**
- **OpenCV**
- **MAVROS and MAVlink**

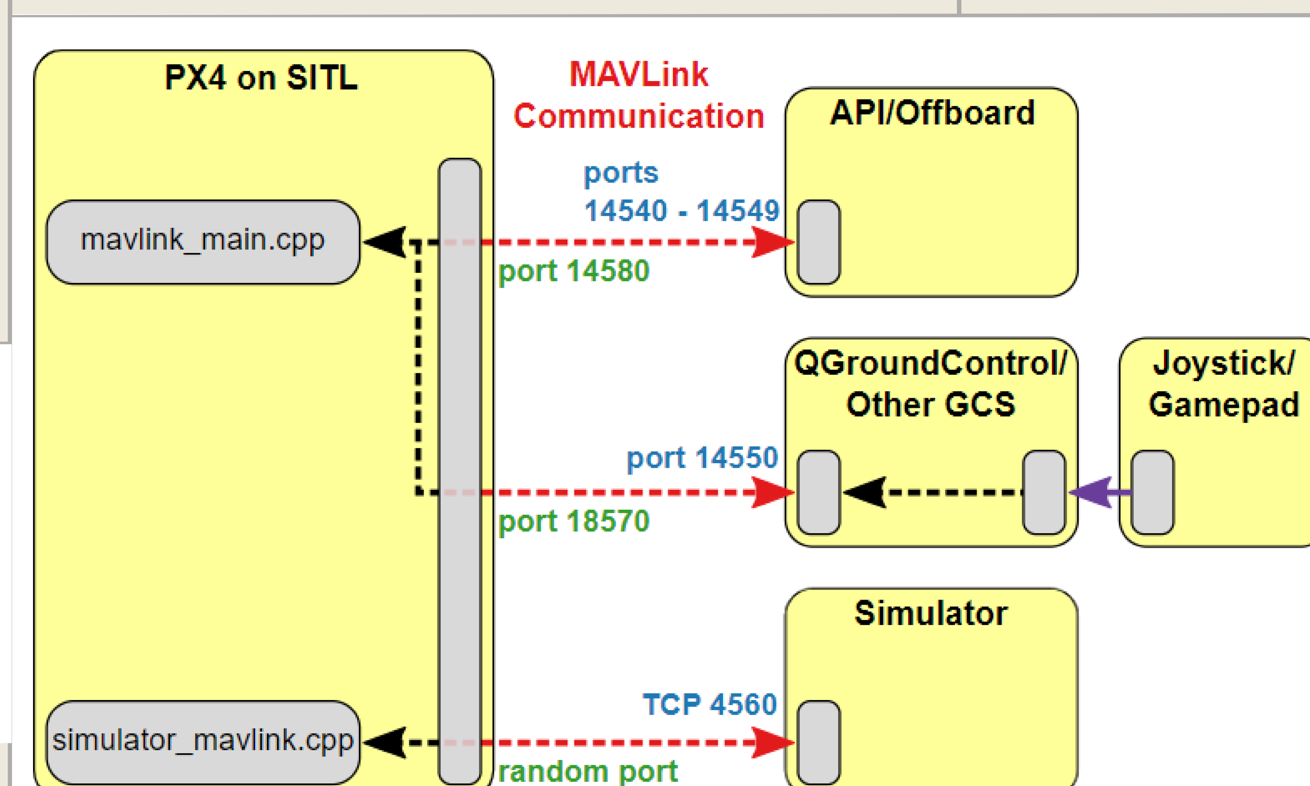
For the implementation of the Algorithm and its command Python with ROSPY Library is used

Feature based algorithm

We proceeded further with implementing our Feature-Based Algorithm on the multicopter. The flow chart on the right shows the underlying scheme of our Algorithm.
We first arm and take off and wait at a specified point for Aruco Marker to be detected. After detection, next phase of the Algorithm takes place which includes planning the trajectory and tracking the marker.

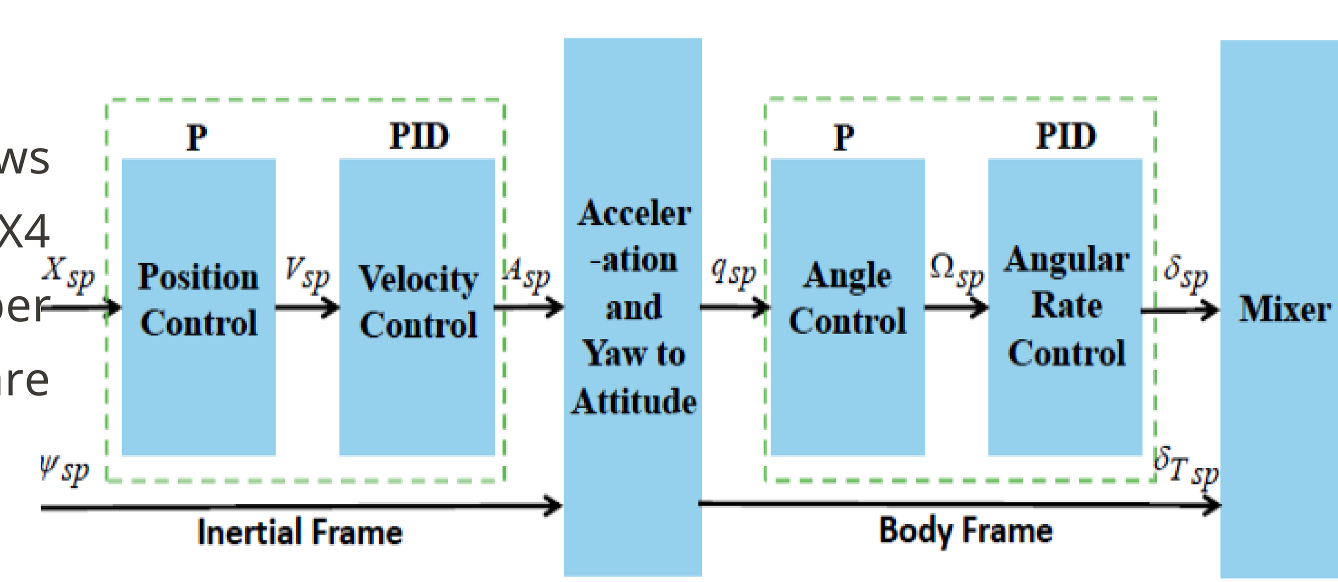
Tracking and Planning

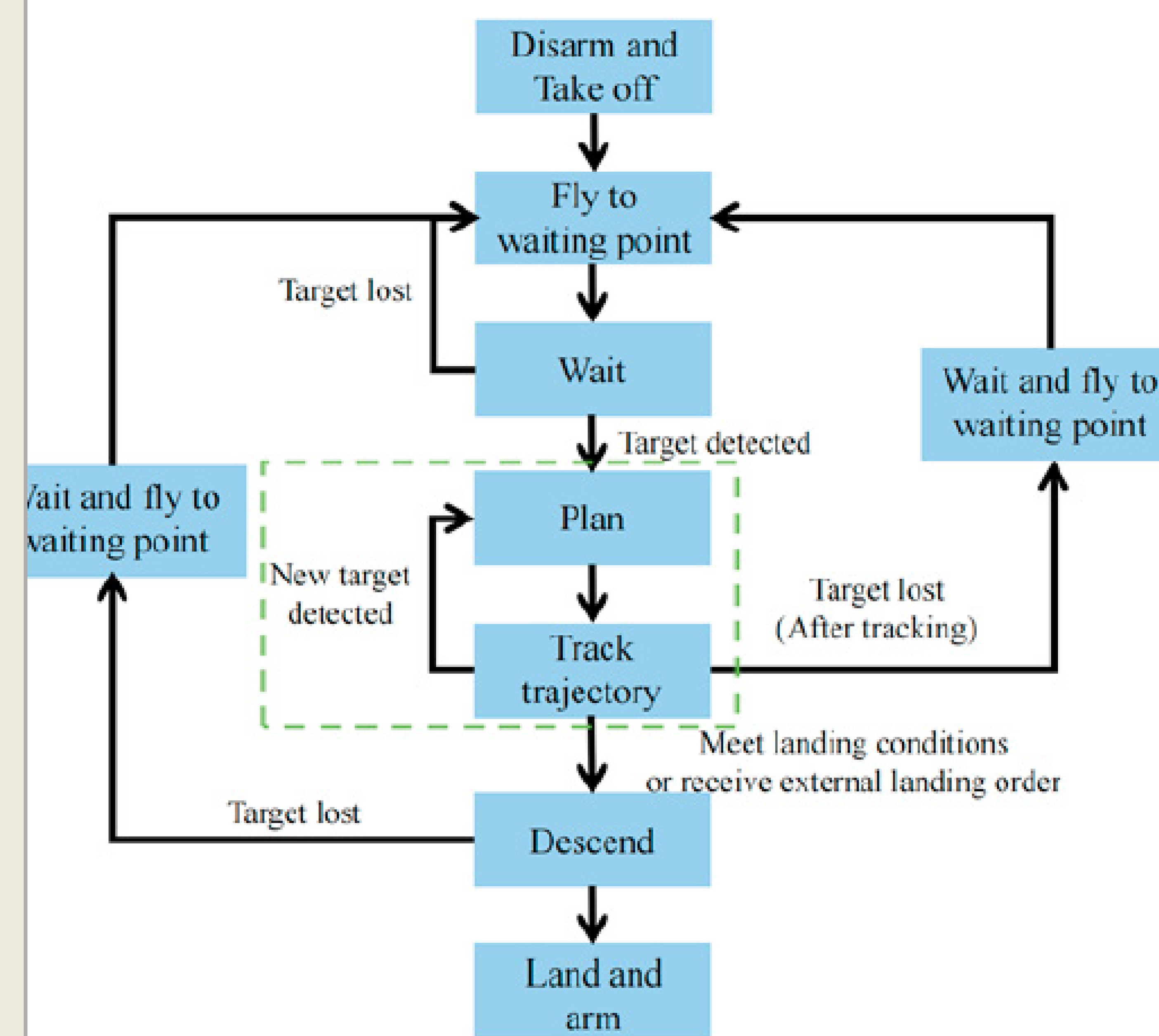
Knowing n-reference 3D points and corresponding 2D projections, we constructed the Perspective-n-Point (PNP) problem to estimate the camera's pose relative to the moving platform thus giving us a relative vector which is then given to PX4 which then gives the command to the UAV



This image shows the flowchart of various software connected. Here PX4 Autopilot is run in SITL mode it's command is communicated to QGroundControl and Offboard mode API through MAVLink established through port 14580 and parallel also established between PX4 and Gazebo simulator

The flowchart on the right shows the functioning inside PX4 Autopilot. It gives a proper schematic of how Commands are processed by the controller

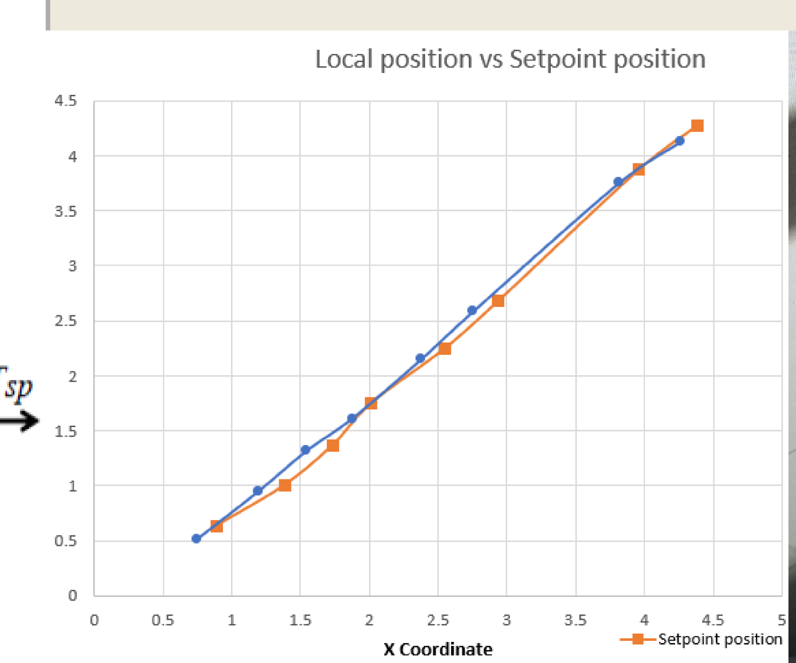


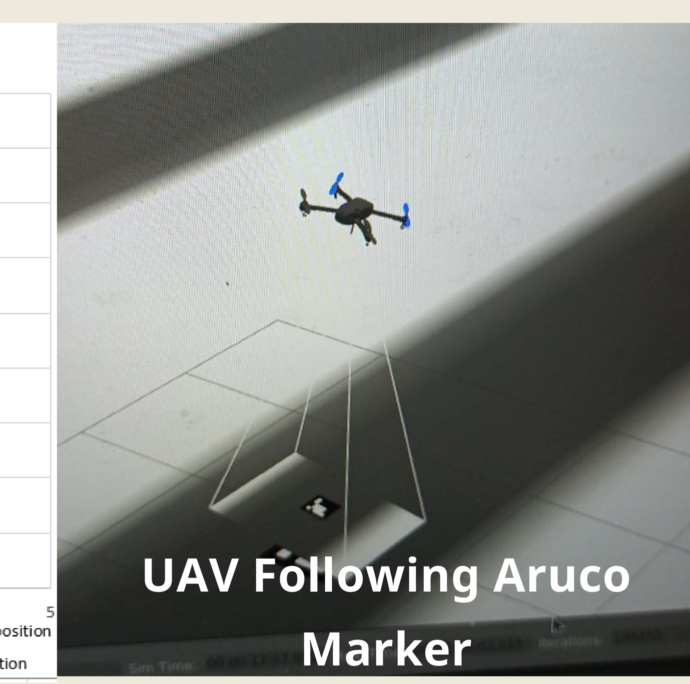


Landing Algorithm Flowchart

Results and Future Work

A graph is plotted between the Local position of the UAV and the Setpoints given through the vision technique. This showed clearly that within an error of ± 0.015 UAV matches its trajectory with the Landing Target.
Based on this work a further refinement using B-spline Regressions in Trajectory planning can be implemented which will reduce errors which will further help in precision Landing and thus can be extended towards ship-deck landing as well





UAV Following Aruco Marker