UAV Autonomous Landing

Team Expeditus

SDSMT MCS

March 21, 2016

Introduction

UAV Autonomous Landing Project

Team Expeditus

Jonathan Dixon, Dylan Geyer, Christopher Smith, Steven Huerta

Sponsor

Dr. Larry Pyeatt

Goal

Demonstrate the capability of a UAV to autonomously take-off, navigate through some waypoints, return to the landing pad, and land with a minimum of distance and orientation error.

Requirements

Goal

- receive a set of waypoints
- autonomously take-off
- navigate through waypoints
- return to launch pad
- **land with $\pm$.1m distance and $\pm 15^\circ$ orientation error**

Limitations

- landing platform is a fixed position
- landing platform is a stable, horizontal surface
- environment is ideal(no wind, gps available, no obstacles)

User Stories/Backlog

- **User 1(U-1):**
  As a user, I want to communicate the waypoints to the UAV.
- **Owner 1(O-1):**
  As an owner, I want the UAV to autonomously take-off from the landing pad.
- **Owner 2(O-2):**
  As an owner, I want the UAV to autonomously navigate through a set of waypoints.
- **Owner 3(O-3):**
  As an owner, I want the UAV to autonomously return to the location of the landing pad.
- **Owner 4(O-4):**
  As an owner, I want the UAV to autonomously land on the landing pad without damaging the craft.
- **Owner 5(O-5):**
  As an owner, I want the UAV to autonomously land on the landing pad with the correct orientation.

As a user, I want to communicate the waypoints to the UAV.

Task No.	Task	Date Completed	Sprint
1	Review previous method/interface for communicating coordinates to UAV.	10/05/15	1
2	Review code that communicates with quadrotor	10/16/15	2
3	Review code that allows a user to input waypoints	10/16/15	2
4	Modify/Rewrite imlementation as necessary	01/23/2016	4

As an owner, I want the UAV to autonomously take-off from the landing pad.

Task No.	Task	Date Completed	Sprint
1	Review previous implementation for autonomous take-off.	10/05/15	1
2	Review code that enables the quadrotor to autonomously take-off from landing pad	10/16/15	2
3	Modify/Rewrite take-off imlementation as necessary	01/23/2016	4

O-2

As an owner, I want the UAV to autonomously navigate through a set of waypoints.

Task No.	Task	Date Completed	Sprint
1	Review previous implementation for navigating waypoints.	10/05/15	1
2	Review code that enables the quadrotor to autonomously navigate through a series of way-points	10/16/15	2
3	Modify/Rewrite take-off imlementation as necessary	01/23/2016	4

As an owner, I want the UAV to autonomously return to the location of the landing pad.

Task No.	Task	Date Completed	Sprint
1	Review previous implementation to autonomously return to location of landing pad	10/05/15	1
2	Review code that allows the autonomous return of the UAV to the landing pad.	10/16/15	2
3	Modify/Rewrite take-off imlementation as necessary	01/23/2016	4

As an owner, I want the UAV to autonomously land on the landing pad without damaging the craft

Task No.	Task	Date Completed	Sprint
1	Review previous implementation for autonomous landing	10/05/15	1
2	Install previous implementation	10/19/15	2
3	Test previous implementation	10/26/15	2

As an owner, I want the UAV to autonomously land on the landing pad with the correct orientation.

Task No.	Task	Date Completed	Sprint
1	Review previous implementation for autonomous landing	10/05/15	1
2	Install previous implementation	10/19/15	2
3	Test previous implementation	10/26/15	2

Initial Common Tasks

Task No.	Task	Date Completed	Sprint
1	Install Ubuntu 14.04 or some other ROS Indigo/Jade distro compliant OS.	09/25/15	1
2	Setup Gazebo 6.+	09/25/15	1
3	Download Rviz package	09/25/15	1
4	Setup Simulation Environment	11/02/15	2

Initial Common Tasks

Task No.	Task	Date Completed	Sprint
5	Review previous iteration of project documentation	09/25/15	1
6	Inspect current quadrotor configuration	09/28/2015	2
7	Identify parts needed for quadrotor	11/02/2015	2
8	Acquire parts needed for hexrotor	12/01/2015	3
9	Build UAV	01/17/16	4
10	Test flight under manual control	01/17/16	4

Sprint 1

Sprint 1 - Successes

- Revised project scope
- Product Backlog - User Stories
- Setup Development Environment
- Review previous years hardware and software

Sprint 1 - Setbacks

- Previous years UAV unusable
- Previous years flight code unusable

Sprint 2

Sprint 2 - Successes

- Visual Homography Code repurposed
- Created simulation environment
- Ordered parts for new Hex-copter

Sprint 2 - Setbacks

- Simulation only supports manual control

Sprint 3

Sprint 3 - Successes

- Assembled Frame, Motors, ESC's
- Many SITL simulations
- Waypoint Publisher publishes mavros commands
- Working image homography code
- Becoming familiar with python openCV libraries

Sprint 3 - Setbacks

- Pixhawk delayed 2 weeks, build not completed
- SITL simulations rejected waypoint files
- SITL simulations rejected mavros commands

Sprint 3.5 + 4 Successes

- Finished construction of UAV
- Manual flight of the UAV achieved
- Autonomous flight of the UAV achieved
- GPS Waypoint navigation achieved

Sprint 3.5 + 4 Setbacks

- AR Track Alvar not working
- Simulation tasks abandoned

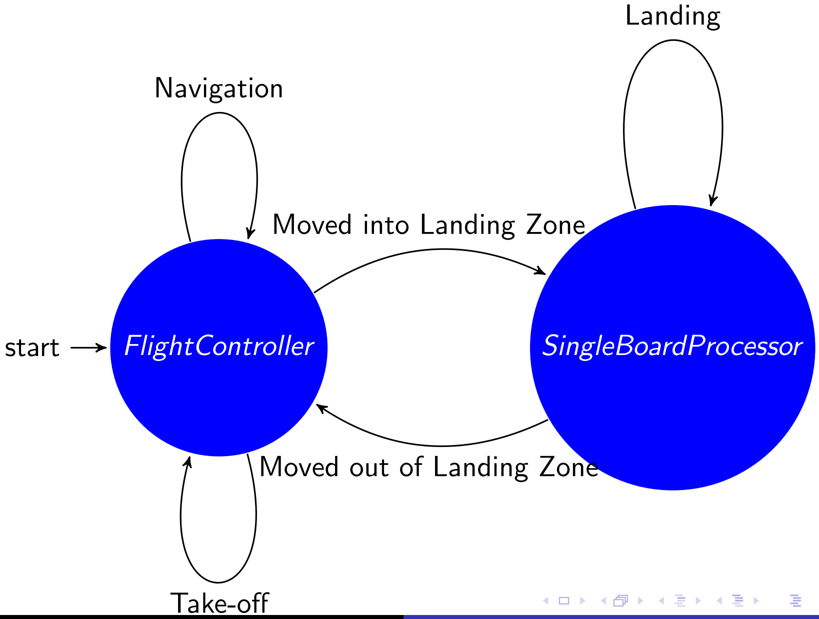
Sprint 5 Successes

- Non-ROS Alvar reading pose and position
- AR Track Alvar reading pose and position
- Basic off-board control on the Pixhawk

Sprint 5 Setbacks

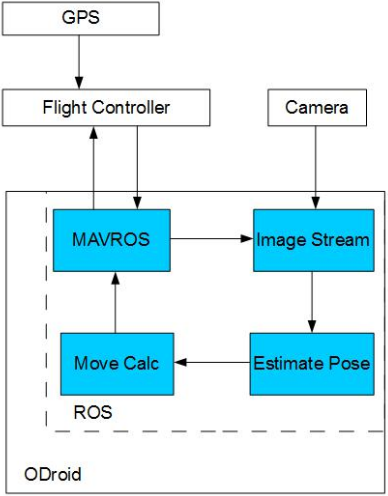
- Localization in off-board control

Design



Landing
Navigation
Moved into Landing Zone
start
FlightController
SingleBoardProcessor
Moved out of Landing Zone
Take-off

Architecture



Hardware Requirements

- ODroid XU4
- Pixhawk Flight Controller
- GPS peripheral
- Camera
- Battery
- UAV(Frame, Motors, ESCs, Power Distribution Board)

Software Requirements

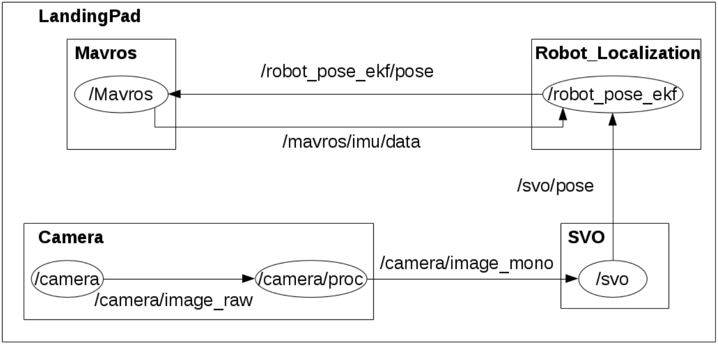
- Mavlink
- Python
- OpenCV
- Robot Operating System(ROS) Indigo/Jade Distro
- Ubuntu 14.04

UAV Design & Tech Specs

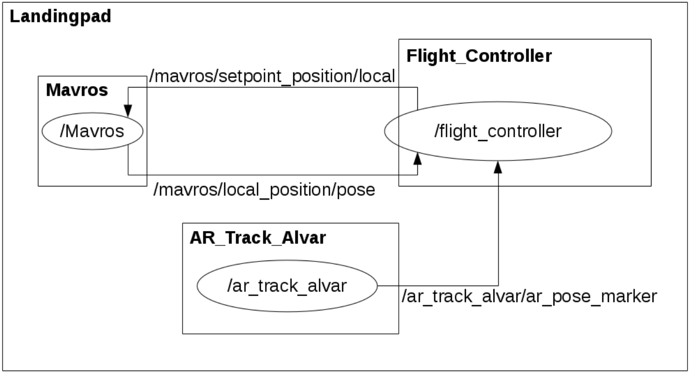
Physical design of the hex-copter is the **Turnigy Talon Hexcopter**

Localization Software Architecture

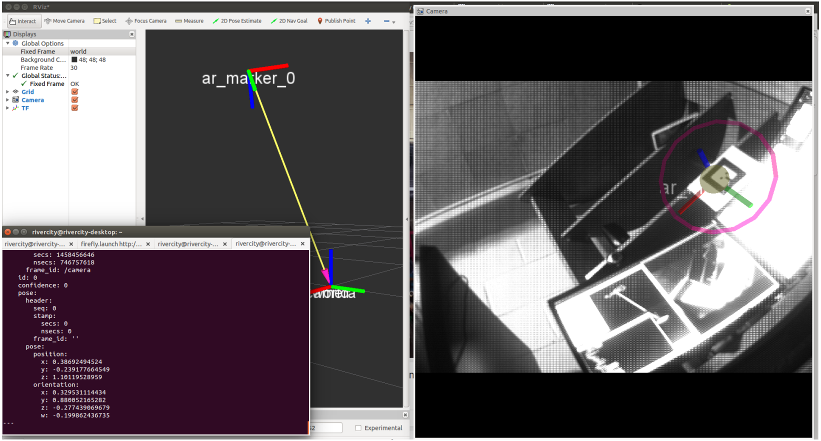
Architecture for communication between ROS nodes for evaluating a local position:



Architecture for communication between ROS nodes to accomplish laninding on the landing pad:



AR Track Alvar can provide pose and orientation of the AR tag relative to the frame of the camera:



As an owner, I want the UAV to autonomously take-off from the landing pad.

Task No.	Task	Testing
1	Review previous implementation for autonomous take-off.	Send mission containing takeoff followed by hover command in simulation.
2	Review code that enables the quadrotor to autonomously take-off from landing pad	Upload code to physical UAV and send mission containing takeoff followed by hover command with manual override enabled.

UAV Testing - O-2

As an owner, I want the UAV to autonomously navigate through a set of waypoints.

Task No.	Task	Test
1	Review previous implementation for navigating waypoints.	Send mission containing takeoff followed by way-points and once the last waypoint is reached a hover command all in simulation.
2	Review code that enables the quadrotor to autonomously navigate through a series of way-points	Upload code to physical UAV and send mission again with manual over-ride enabled

As an owner, I want the UAV to autonomously return to the location of the landing pad.

Task No.	Task	Test
1	Review previous implementation to autonomously return to location of landing pad	The last waypoint from the previous test should be the landing pad but to verify a image match will be used in simulation that will verify that the UAV is in fact above the landing pad.
2	Review code that allows the autonomous return of the UAV to the landing pad.	The physical UAV will match what it sees with its camera to a image it has stored of the landing pad at multiple heights.

As an owner, I want the UAV to autonomously land on the landing pad without damaging the craft

Task No.	Task	Test
1	Review previous implementation for autonomous landing	The UAV should be able to detect the landing pad
2	Install & modify previous implementation	The UAV should be able to begin to lower onto the pad
3	Test implementation	The UAV should land gently on the pad

Visual Homography Landing Testing - O-5

As an owner, I want the UAV to autonomously land on the landing pad with the correct orientation.

Task No.	Task	Test
1	Review previous implementation for autonomous landing	The UAV should be able to calculate its angle wrt the pad
2	Install& modify implementation	The UAV should be able to rotate to match the pad
3	Test implementation	UAV should maintain orientation throughout descent

Integration - U-1

As a user, I want to communicate the waypoints to the UAV.

Task No.	Task	Test
1	Review previous method/interface for communicating coordinates to UAV.	Connect GPS to pixhawk and verify it recieves a connection and in Mavros verify we can retrieve gps coordinates from gps in ROS.
2	Review code that communicates with quadrotor	Verify that all commands in Mavros can be sent to and accepted by the Pixhawk and it acts appropiately
3	Review code that allows a user to input waypoints	Verify that user entered waypoints can be uploaded into Pixhawk Successfully with mavros.

Remaining Backlog

- **Owner 4(O-4)**
- **Owner 5(O-5)**

Revised Goals

- User will need to create file containing missions for offboard control to use, not
- Use of only AR Tag, not AR Tag + Landing Lights.

Risk Analysis

- Localization: Without a better approach to localization, our estimates of our current pose are very poor. As a result our UAV is unable to navigate to the AR tag.

Risk Mitigation

- Localization
  - Use visual odometry with the camera we already are using
  - Attach more sensors if necessary

Timeline

Sprint 6 3/21/16 4/15/16

- Finish Landing Algorithm Simulations(O-4,O-5)

Sprint 6 necessitates solving our localization issues, so that we can combine our vision and message passing framework to complete this project.

Budget

Item	Qty	Price	Total
Frame	1	$79.99	$79.99
Motors	8	$23.99	$191.92
ESCs	8	$17.78	$142.24
Pixhawk	1	$199.99	$199.99
Power Distribution	1	$19.99	$19.99
GPS Mast	2	$10.00	$20.00
GPS	2	$89.99	$179.98
Power Module	1	$24.99	$24.99
Odroid XU4	1	$75.95	$75.95
Props(set of 4)	3	$7.55	$22.65
TOTAL			$957.70

IP & Licensing

Intellectual Property:
Project is owned by SDSMT

Licensing for Dependencies:

- OpenCV: BSD
- ROS: BSD
- Mavlink: LGPL version 3
- QGroundControl: GPL version 3
- ROS Packages
  - mavros: GPLv3, LGPLv3, BSD
  - ar_track_alvar: BSD
  - pointgrey_camera_driver: BSD
  - usb_cam: BSD

Additionally, all members of the team are registered with the FAA and have received their drone pilot's license.

Demos

- Communication
- Offboard Control
- AR Track Alvar

END

Questions