

# UAV Autonomous Landing

Team Expeditus

SDSMT MCS

December 9, 2015

# 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  $\pm .1$  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

**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

**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

**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

**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

# C Continued

## 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

## 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 - Successes

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

## Sprint 2 - Setbacks

- Simulation only supports manual control

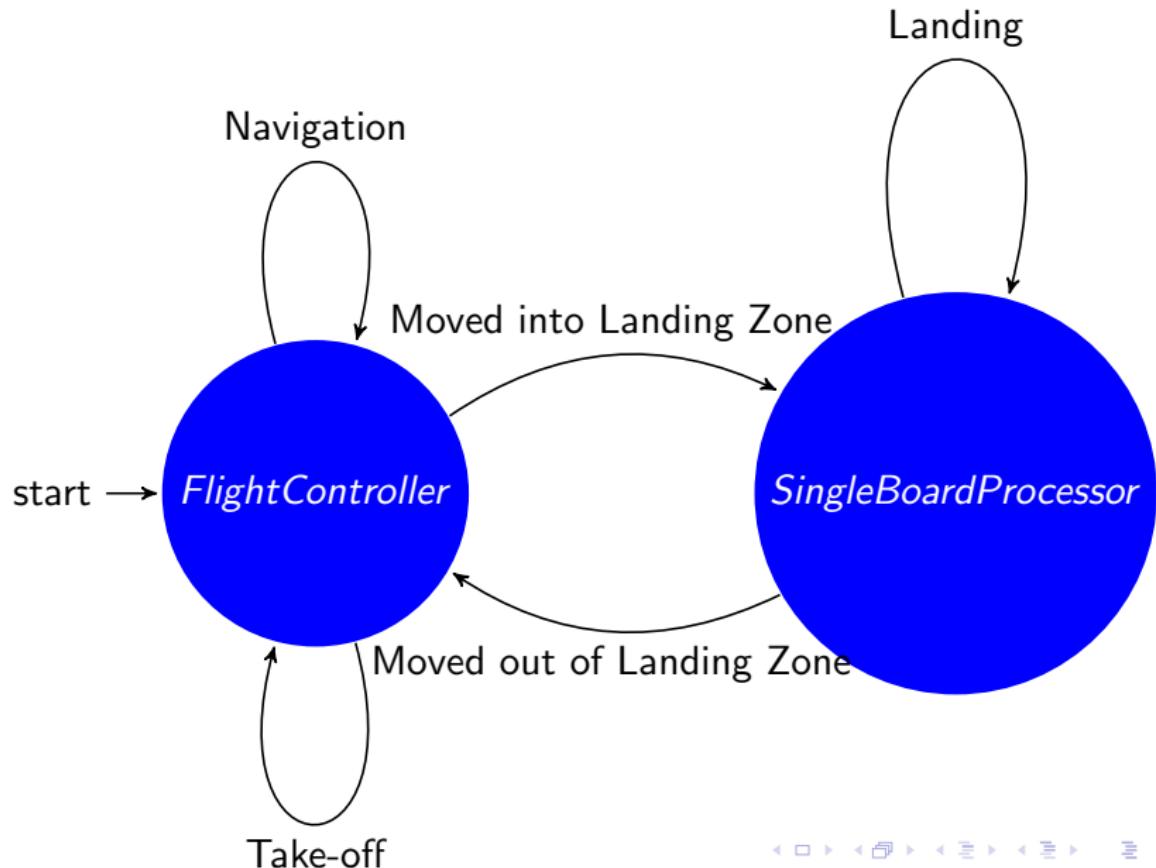
## Sprint 3 - Successes

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

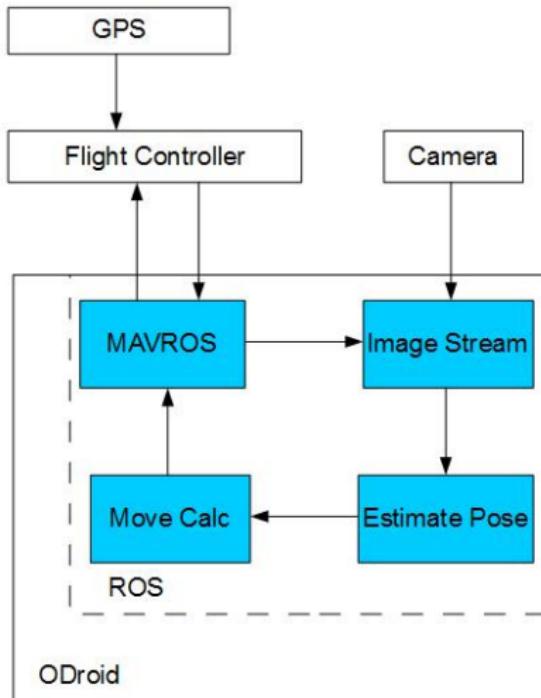
## Sprint 3 - Setbacks

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

# Design



# Architecture



# Hardware Requirements

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

# Software Requirements

- Mavlink
- 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**



# Visual Homography Design & Tech Specs

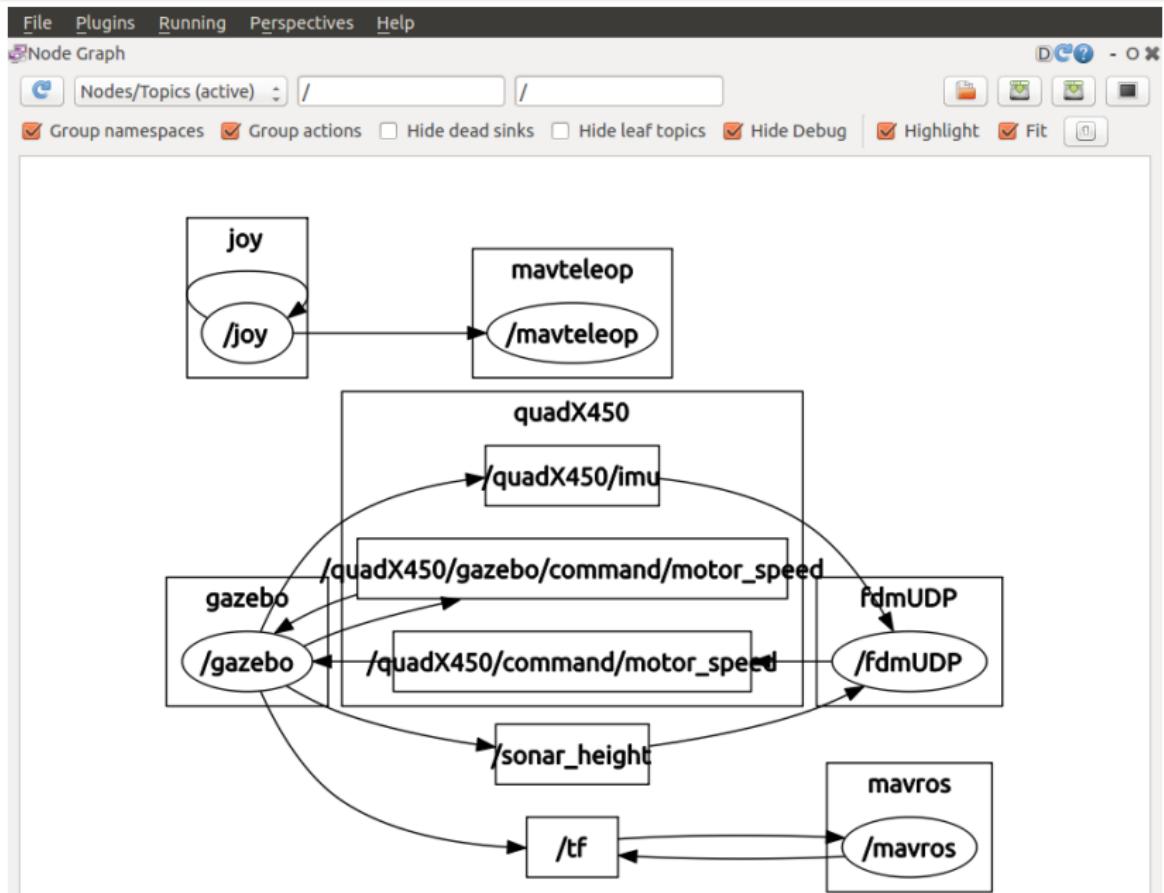
## Design & Tech

- Sarsa
- 2D State Space
- 3D State Space

## Setbacks

- Research and familiarizing
- Simulation Environment

# Simulation Design & Tech Specs



**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.

**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 waypoints 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 override 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	Test to verify
2	Install previous implementation	Test to verify
3	Test previous implementation	Test to verify

# 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	Test to verify
2	Install previous implementation	Test to verify
3	Test previous implementation	Test to verify

**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.	Test to verify
2	Review code that communicates with quadrotor	Test to verify
3	Review code that allows a user to input waypoints	Test to verify

# Remaining Backlog

- **User 1(U-1)**
- **Owner 1(O-1)**
- **Owner 2(O-2)**
- **Owner 3(O-3)**
- **Owner 4(O-4)**
- **Owner 5(O-5)**
- **Common**

# Revised Goals

Project Goals remain fixed

# Risk Analysis

- Simulation: SITL has proven to be problematic
- UAV Build: Borrowing items from UAV Team (Radio and Control)
- UAV Build: One UAV for physical testing and demonstration
- Landing Algorithm: Many approaches, we may pick the wrong one

- Simulation:
  - Attempt HIL as Alternative
- UAV Build(Sharing)
  - Schedule use of tools to prevent conflict
  - Request more funding if schedule is untenable
- UAV Build(One Shot)
  - Integrate manual control override
  - Validate solutions through simulation

## Sprint 3.5 12/16/15 to 1/10/16

- Finish UAV Build(C)
- Manual Flight of UAV(C)
- Autonomous Flight of UAV(C,U-1,O-1,O-2,O-3)
- Resolve Simulation Issues(C)

At the end of break, 3 backlog items will have been completed

## Sprint 4 1/18/16 2/5/16

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

At the end of sprint 4, we should have a landing approach validated by simulation.

# Timeline...continued

## **Sprint 5 2/15/16 3/4/16**

- Integration of Landing Autonomy on UAV(O-4,O-5)

At the end of sprint 5, we should have completed the remainder of backlog items.

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

- Refinement

At the end of sprint 6, project will be complete

# 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

Intellectual Property:  
Project is owned by SDSMT

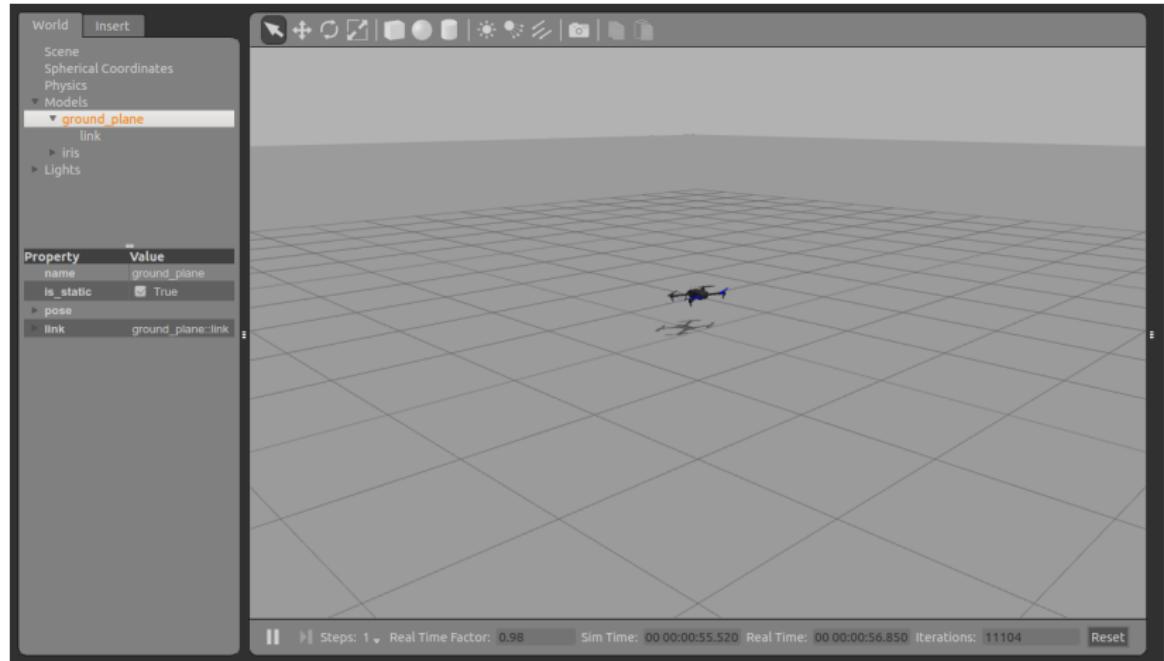
Licensing for Dependencies:

- OpenCV: BSD
- ROS: BSD
- Mavlink: LGPL version 3
- QGroundControl: GPL version 3

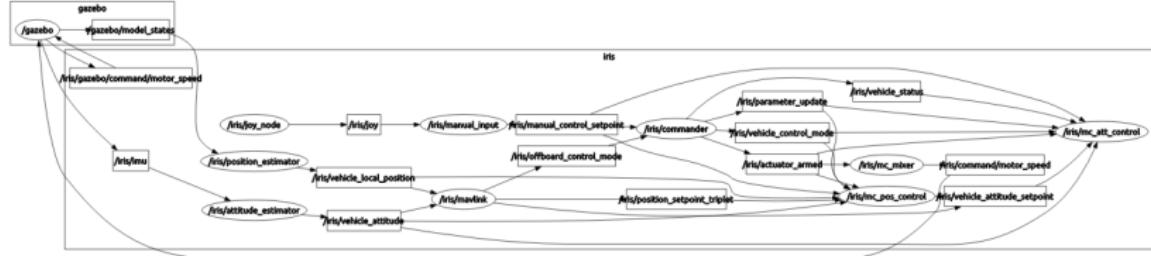
## Demos

- PX4 ROS SITL
- Ground Control Station
- Visual Homography

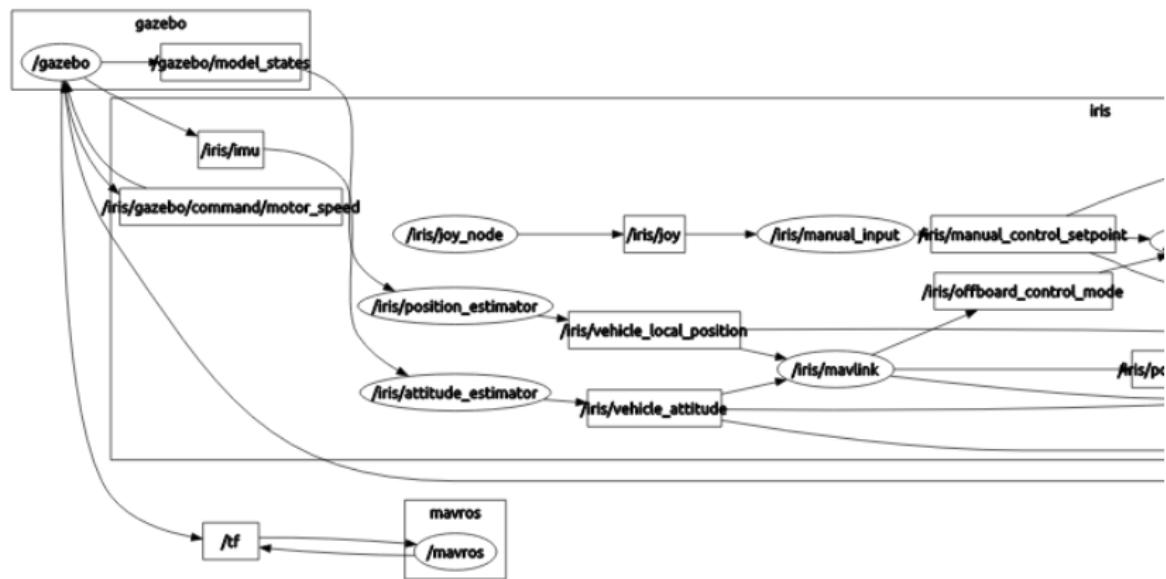
# Simulation - PX4 ROS SITL



# PX4 ROS SITL...continued



# PX4 ROS SITL...continued



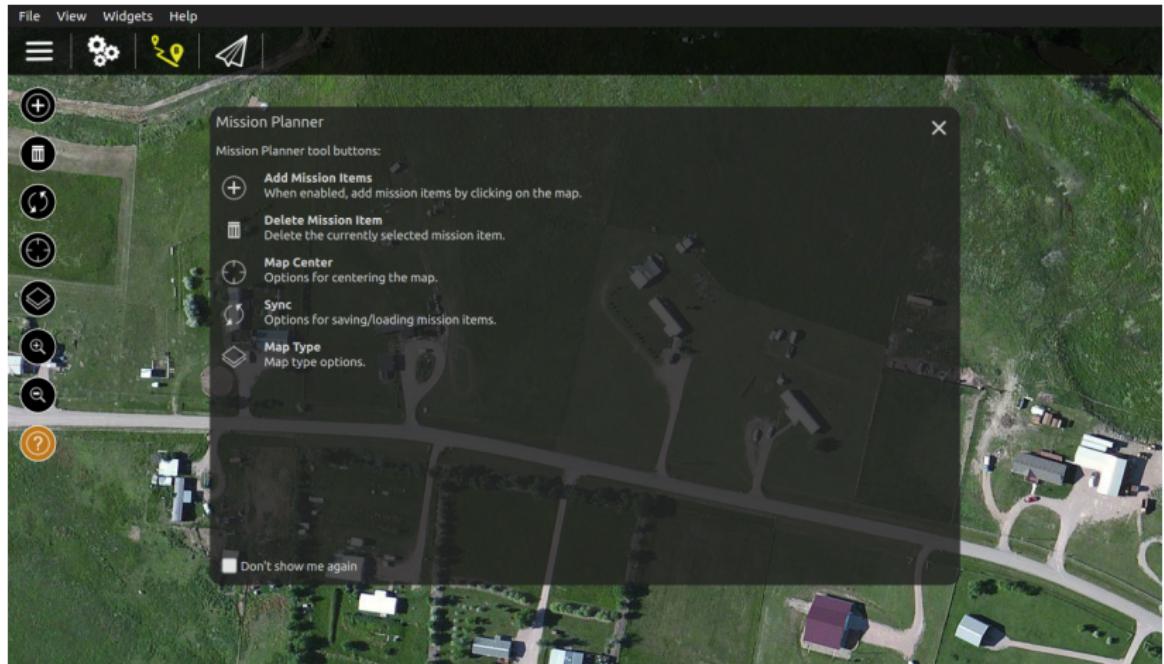
# PX4 ROS SITL...continued

```
device [/rosout] found  
[mavros-1]: started with pid [5760]  
[1449625813.107703184]: FCU URL: udp://localhost:14560@localhost:14565  
[1449625813.108111698]: udp0: Bind address: 127.0.0.1:14560  
[1449625813.108231677]: udp0: Remote address: 127.0.0.1:14565  
[1449625813.108390158]: GCS bridge disabled  
[1449625813.111757989]: udp0: Remote address: 127.0.0.1:14565  
[1449625813.246282225, 415.735000000]: Plugin 3dr_radio loaded and initialized  
[1449625813.264741115, 415.755000000]: Plugin actuator_control loaded and initialized  
[1449625813.427378113, 415.905000000]: Plugin cam_imu_sync loaded and initialized  
[1449625813.446682768, 415.925000000]: Plugin command loaded and initialized  
[1449625813.446866194, 415.925000000]: Plugin distance_sensor blacklisted  
/iris/MP_YAW_FF  
/iris/MP_YAW_P  
/iris/mavlink/mavlink_fcu_url  
/iris/mixer  
/iris/motor_offset_radps  
/iris/motor_scaling_radps  
/iris/vehicle_model  
/robot_description  
/rosdistro  
/roslaunch/uris/host_rivercity_lifebook_t725_4524  
/rosversion  
/run_id  
/tf_prefix  
/use_sim_time  
/iris@rivercity-LIFEBOOK-T725:~/catkin_ws$ roslaunch fcu url  
udp://localhost:14565@localhost:14560
```

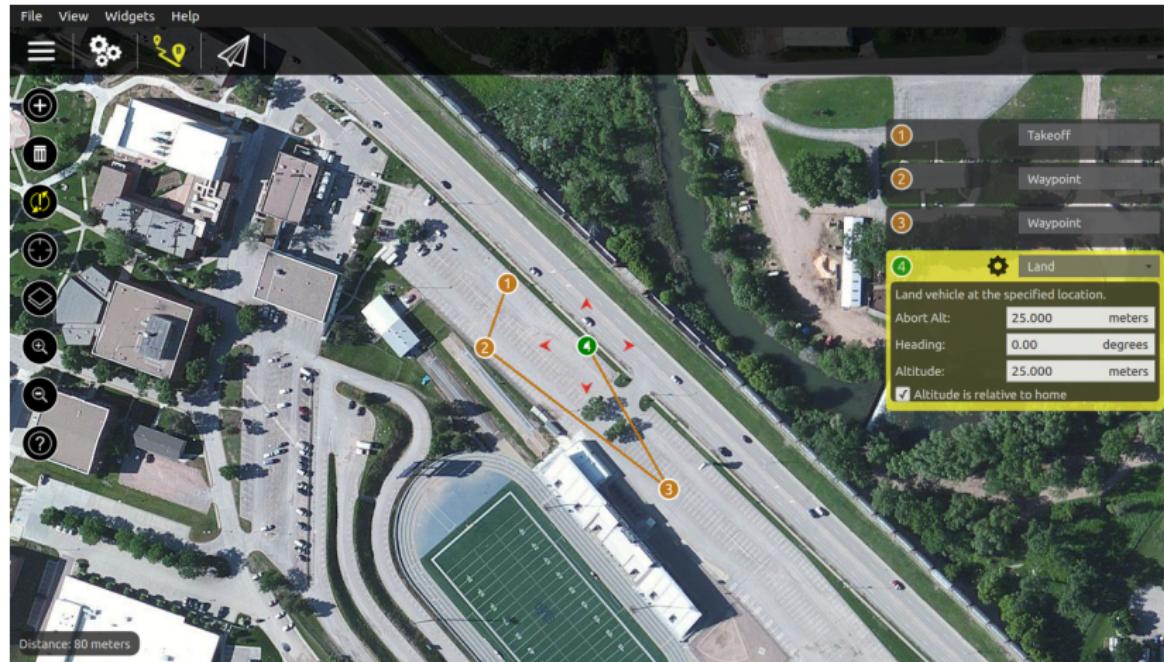
# Ground Control Station - QGroundControl



# Ground Control Station...continued

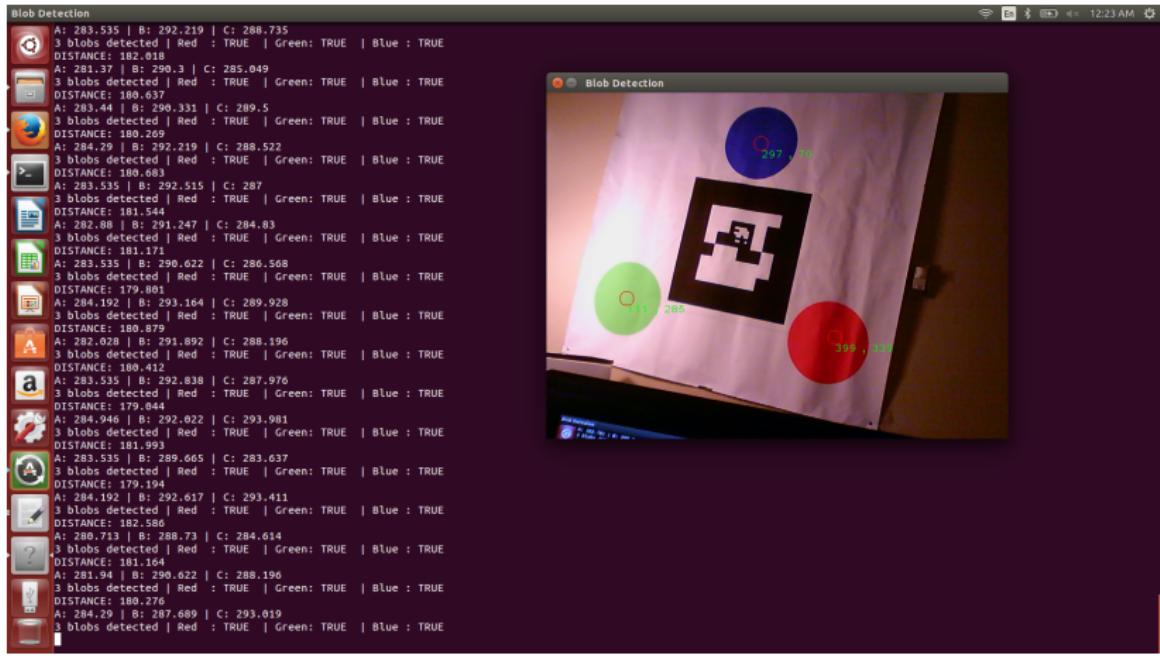


# Ground Control Station...continued



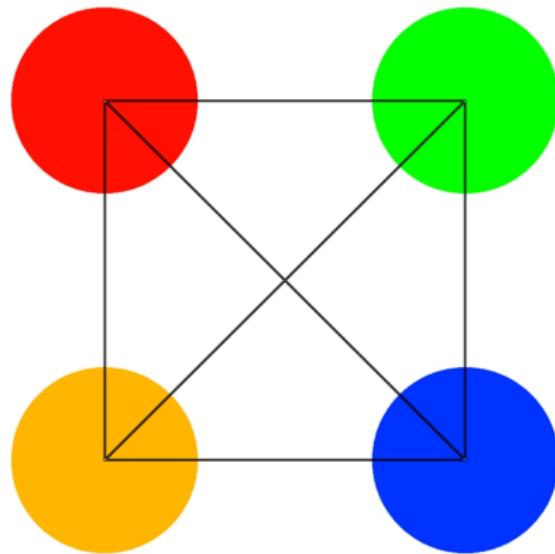
# Vision-Based Landing

- Began playing with last year's code
- Able to detect three blobs and distance to target



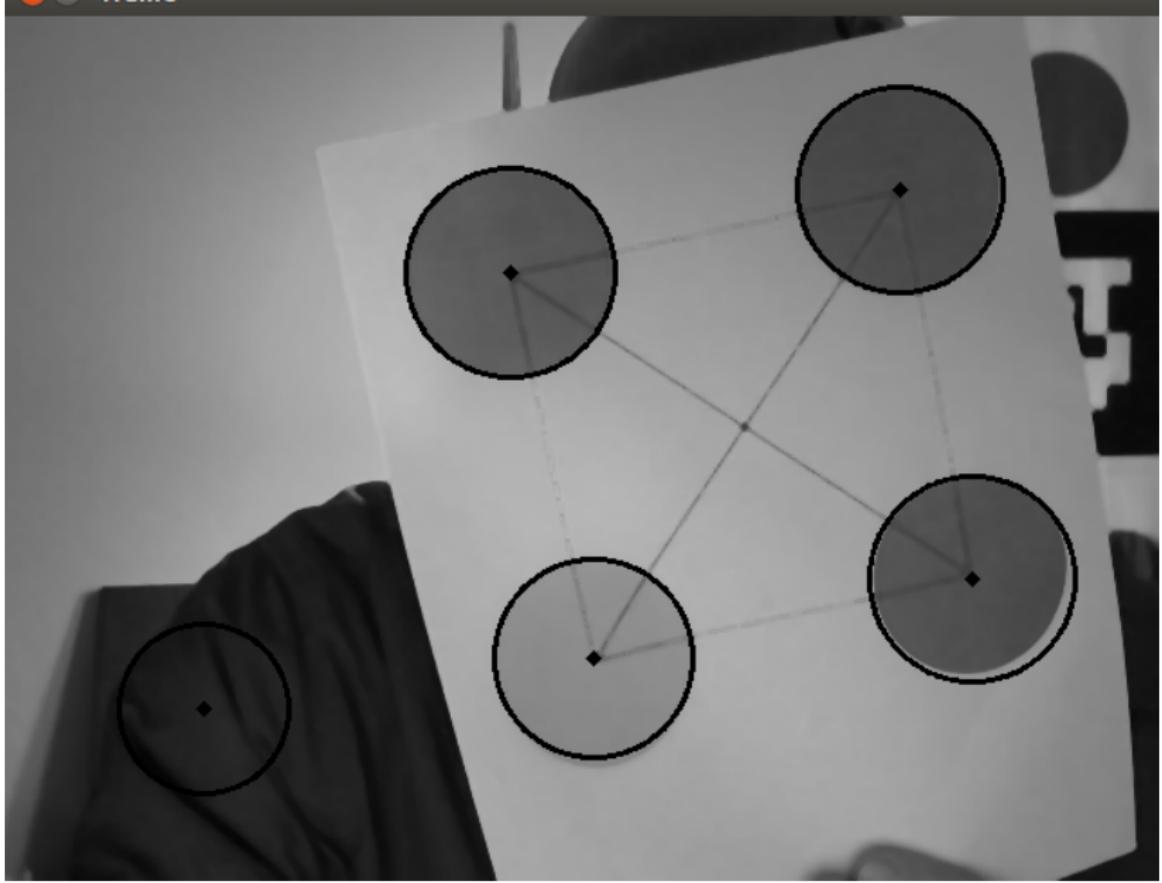
# Vision-Based Landing...continued

- After our computer vision class, we considered adding a fourth circle
- This gives us a square target instead of triangle





frame



END

## Questions