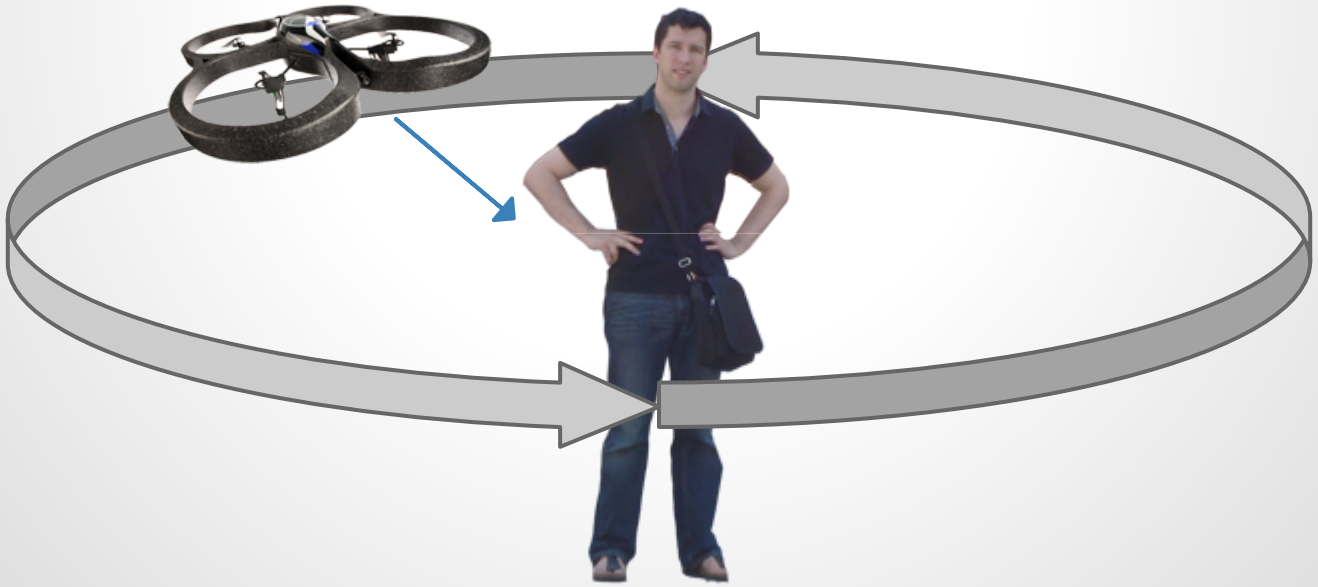
Dragon Sheep

Project Status
Autonomous Circle Flying

Michael Shelley - Tom Rothoerl - Oliver Dunkley

Motivation

Task: Autonomously circle a human with variable distance, altitude and speed.



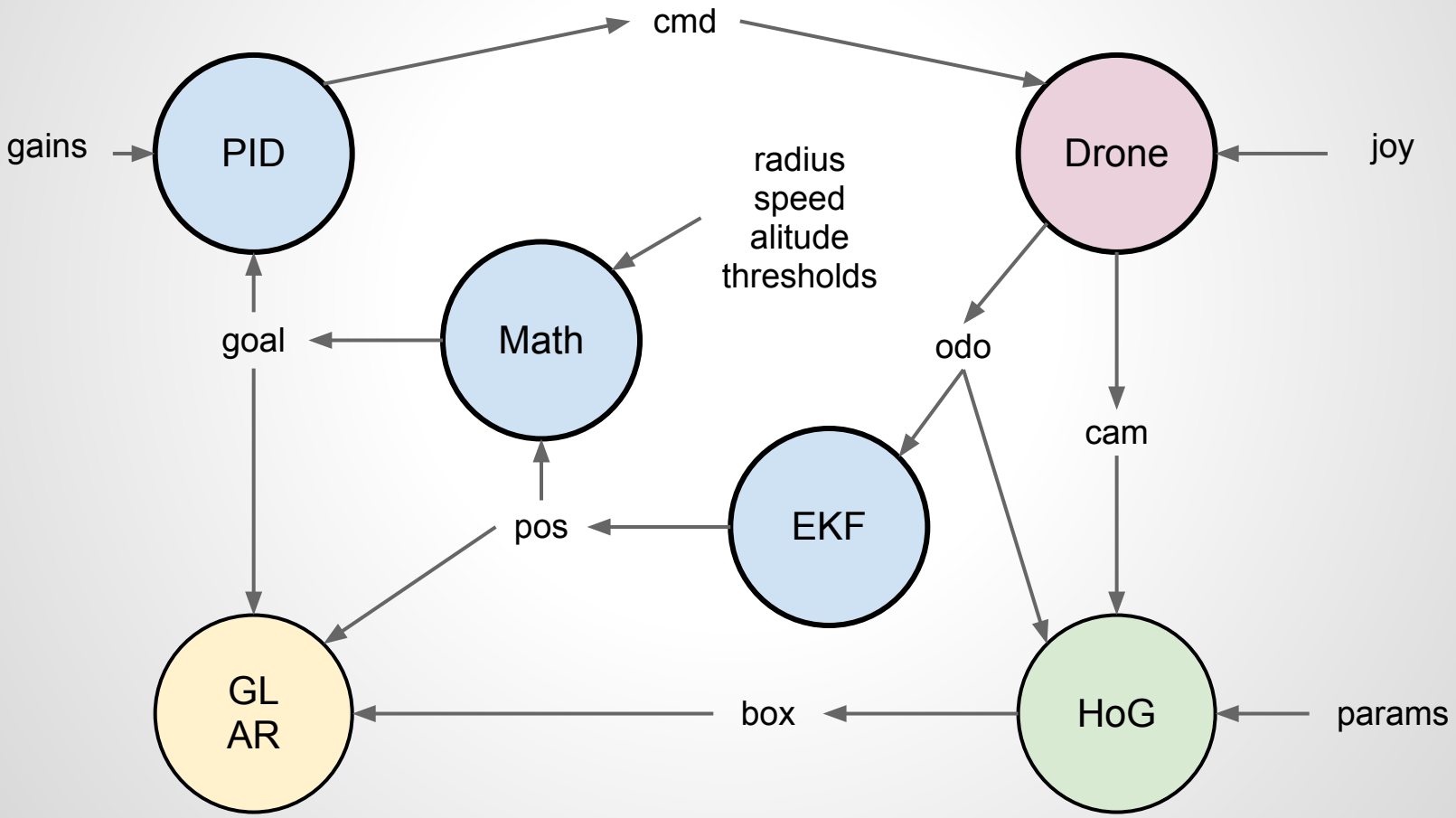
Problem Specification

- Use opencv to detect the person (HoG)
- Determine location of quadrocopter with bounding box of human
  - Treat bounding box as a marker with a known size in the real world.
  - Calculate pose estimation from virtual marker.
- Specify a goal position and orientation relative to the current location and the human
- Recover automatically when human is lost

Approach

- Use opencv HoG detector to find the person
- Calculate pose from bounding box
- When no human is detected uncertainty rises
- When uncertainty crosses a threshold switch to search mode
  - Stop moving
  - Rotate until the human is found again
  - If enough time passes, land the copter

ROS Nodes and Msgs



Implementation Plan

1. Create bag files with a human at varying distances and locations
2. Find HoG bounding box of human and figure out the size of the box in the real world
3. Use a 4 point algorithm to get the quadrocopter location
4. Plug it into the Kalman Filter and create a goal at distance r from the human.
5. Vary the goal so the copter goes in a circle
6. Implement search mode for when the human is lost for a long period.

Demo

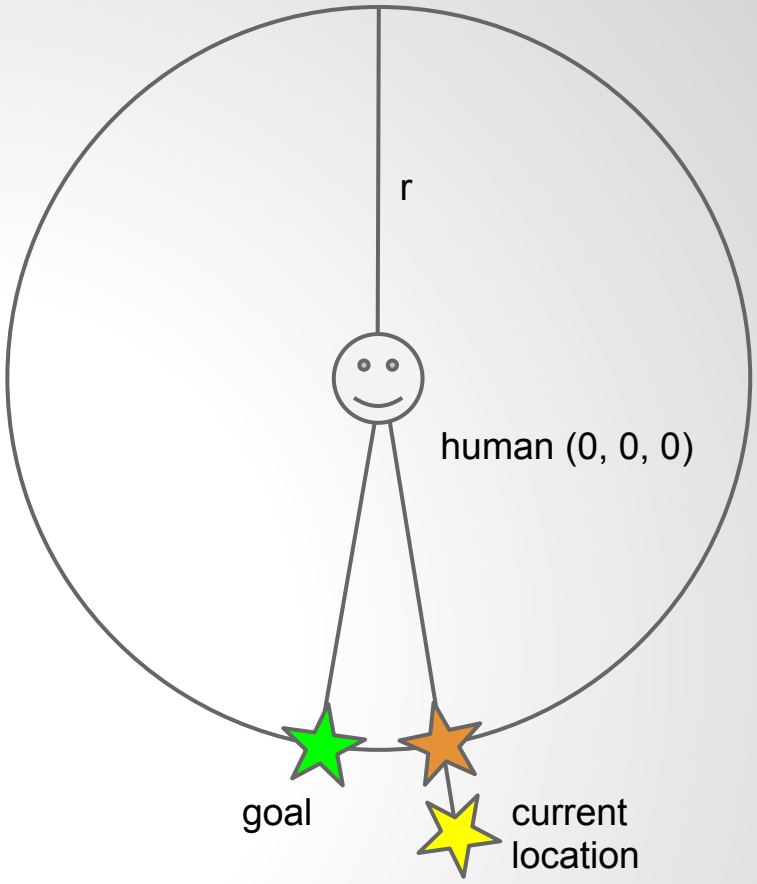
HoG with image rotation

Questions?

Future Work

- A quadrocopter rotating around the user would be useful in a number of applications such as sports, dancing, amateur movies etc.
- Using something other than HoG would be a more general approach to circling a target.

Goal Positioning



Rotation from current angle to angle facing origin

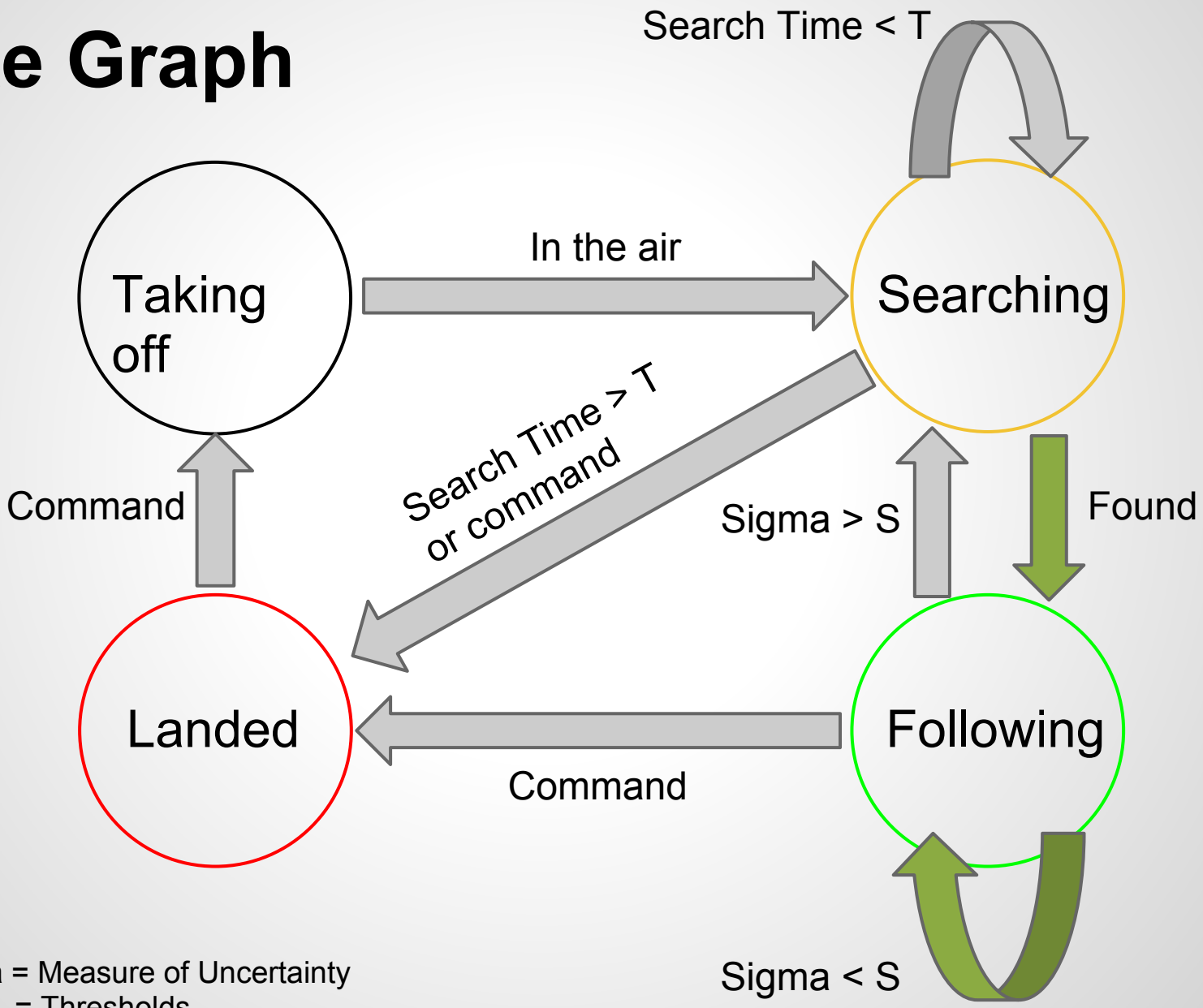
Rotation specifying goal distance

$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix}_{goal} = T_O^{-1} R_\psi T_O T_C R_\phi \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}_{copter}$$

Transform from edge of circle to origin

Translation to closest point on circle of radius r

State Graph



Sigma = Measure of Uncertainty
S, T = Thresholds