R.A.V.E.N.

Remote Autonomous Vehicle Explorer Network

William Etter, Paul Martin, and Rahul Mangharam

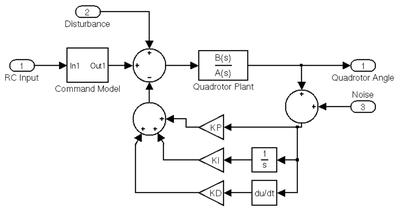
http://www.airhacks.org | http://mlab.seas.upenn.edu/

RAVEN is a leader-follower network of quadrotors capable of human-controlled and autonomous flight navigation for search & rescue operations.

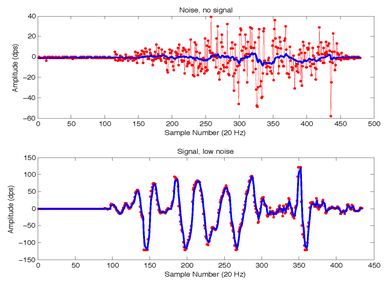
RAVEN uses a combination of on-board vision tracking and wireless communication of attitude measurements to track a leader autonomously.

Stability Control

A PID control loop is used to stabilize the attitude of the quadrotor using Euler angles Φ, Θ, ψ—three control loops in total.



PID Control for a Single Angle



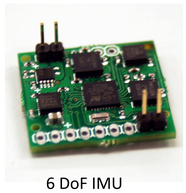
Low-Pass Filter on Rate Gyro Noise for Derivative Feedback

Experimental PID gain values:

- $K_P(\Phi, \Theta) = 7.5$
- $K_I(\Phi, \Theta) = 0.0$
- $K_D(\Phi, \Theta) = 2.15$
- $K_P(\psi) = 9.0$
- $K_I(\psi) = 0.0$
- $K_D(\psi) = 2.6$

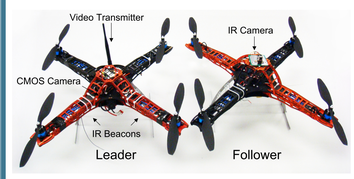
Hardware and software used:

- 6 Degree of Freedom IMU at 255 Hz
- Consumer ESC brushless DC motor controllers (upgraded to 400 Hz)
- Extended Kalman Filter
- ATMega1280 16 MHz microcontroller for main control loop
- First order digital low-pass filter for rate gyro noise reduction in mid-flight

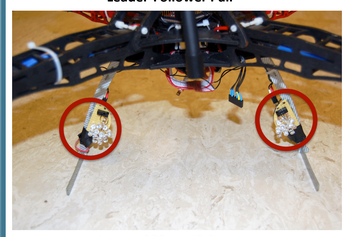


6 DoF IMU

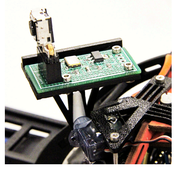
Design



Leader-Follower Pair



Leader 950 nm IR Beacons



Infrared Camera and Servo Mount

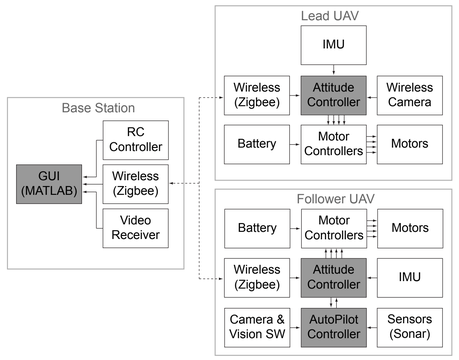
Network Sensing & Control

Wireless Link:

- IEEE 802.15.4 Zigbee 900 MHz link
- Leader to Follower: 20 packets / 60 Bps
- Leader to Base: 5 packets / 15 Bps
- Follower to Base: 20 packets / 480 Bps

Infrared Camera:

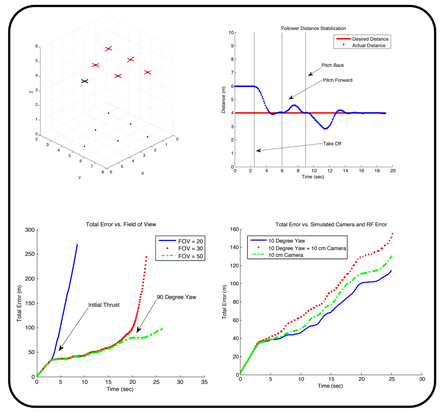
- 500 Hz I²C Camera, 4 point tracking, 1024x768 resolution
- 2 5x2 arrays of 950 nm infrared LED beacons on leader
- Servo-controlled pitch to increase effective FOV



System Overview

Flocking Model

A numerical integration model is used to verify the algorithms and transforms needed to convert IR Camera and wireless attitude information to desired angles.



Top Left: Recursive Flock **Top Right**: PID on relative position
Bottom Left: Simulated camera FOV **Bottom Right**: Response to pseudo-random errors

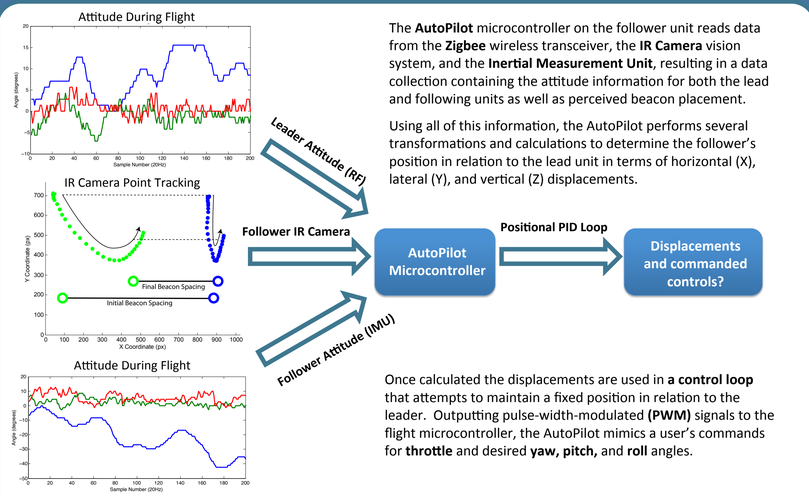
A **Matlab model** is used to verify the algorithms and transforms needed to compute desired angles. The model is characterized by:

- Quadrotor units as dynamic objects
- **Numerical integration** from angular accelerations to angles
- **Simulated on-board camera** and RF link
- **Recursively defined** flocking configurations

Model shows under ideal conditions, system will be:

- **Robust to noise** in both RF and camera
- **Operational with field of views (FOV)** >50 degrees

Sensor Fusion

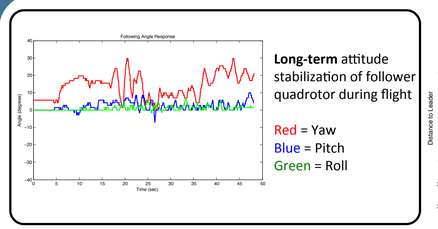


The **AutoPilot** microcontroller on the follower unit reads data from the **Zigbee** wireless transceiver, the **IR Camera** vision system, and the **Inertial Measurement Unit**, resulting in a data collection containing the attitude information for both the lead and following units as well as perceived beacon placement.

Using all of this information, the AutoPilot performs several transformations and calculations to determine the follower's position in relation to the lead unit in terms of horizontal (X), lateral (Y), and vertical (Z) displacements.

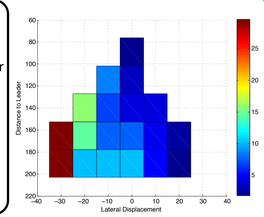
Once calculated the displacements are used in **a control loop** that attempts to maintain a fixed position in relation to the leader. Outputting pulse-width-modulated **(PWM)** signals to the flight microcontroller, the AutoPilot mimics a user's commands for **throttle** and desired **yaw, pitch,** and **roll** angles.

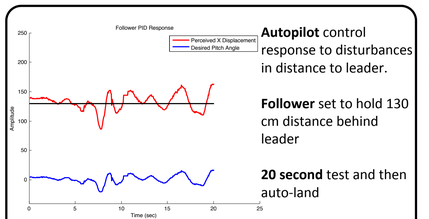
Flight Test Results



Long-term attitude stabilization of follower quadrotor during flight

Red = Yaw
Blue = Pitch
Green = Roll





Autopilot control response to disturbances in distance to leader.

Follower set to hold 130 cm distance behind leader

20 second test and then auto-land

IR Camera Errors vs. Displacement: measured and computed distances vs. actual measured distances. The top represents a location closer to the lead quadrotor.