RF Silent Drone Navigation
OUR TEAM

Matthew Dupree
Student Lead

Xihan Liu
Data Analysis

Yingchao Zhu
PCB/Schematic
01
OVERVIEW
- Liftoff from a boat at sea
- Land on a boat at sea
  - Limited-size landing area
  - Moving landing area
  - Miss == wet hardware
- RF-silent: No radio communication
  - No pilot
  - No landing beacon
  - No external processing
    - Computer vision is hard on an energy budget!
Problem Formalization
Camera: Raspberry Pi Camera Module V2
Companion Computer: Raspberry Pi 3B+
LiPo to 5V Voltage Regulator
Flight Controller: Pixhawk 2.4.8
GPS: Micro M8N GPS Module
Motors 3-Phase
Electronic Speed Controllers
Power Management Board
Ethernet
Lipo Battery
LiPo
Parallel
PWM
UART
02 HARDWARE
Three-layer design
- Top: GPS, Flight Controller, Telemetry radio
- Middle: Companion Computer, Camera
- Bottom: 4s Lipo Battery
Pixhawk v2.4.8

- Flight control unit (FCU)
- Uses ArduPilot firmware
- Ensures drone stability in flight
uBlox M8N Micro GPS

- GPS unit
- Provides drone lat/long coordinate reference
- Used until landing target detected
Transceiver Telemetry Radio

- Broadcasts debugging information to GCS
- Used for our testing -- not required for flight
Middle: Companion Computer, Camera
Raspberry Pi 3B+

- Companion computer
- Performs CV tasks to identify the target and communicate target transform to FCU
Raspberry Pi Camera Module v2

- 8 megapixel camera capable of taking photographs of 3280 x 2464 pixels
- Used to detect AprilTags in 10FPS 640x480 mode
- Bottom: 4s Lipo Battery
PRINTED CIRCUIT BOARD (PCB)
- Function as a voltage regulator to convert the input voltage from 14-22V to 5V output voltage
- Diode after the input voltage in order to prevent reverse current
- 2-layer PCB
- 45 * 59 mm
- Holes for thermal dissipation
AprilTag 3

- Landing Detection Target
- Works on resource-constrained platforms (like our RPi 3B+)
- Full transforms from single stills!
Stats
With current onboard processing, we have tested that at 5 meter altitude the craft can search at 10FPS in a 5x6.7 meter box below the vehicle for a 16.5cm square tag.

Could do far better with more processing power!

(Example photo taken at ~2m alt. and has been cropped.)
First-choice:

PX4 Autopilot

- Great droning OS
- Great debug tools (eg MAVLink Shell)
- Doesn’t support our precision landing use!

Old but gold:

ArduPilot Multicopter

- Heavily used
- Many configurations
- Advanced features
- Spaghetti-code
- Difficult configuration
- Missing debugging shells
Ubiquity Robotics’ Ubuntu Mate 18.04 With ROS Kinetic

- All open-source parts
- Free to use
- Spreads tasks among processes to make best use of hardware
- Lots of logging tools!
Processing Pipeline

- A large stack of software packages from the Robot Operating System (ROS) ecosystem
- High-throughput communication between nodes via ROS topics
- Relies on many complex configuration files

Packages in **bold** we had to largely rewrite or write entirely ourselves!
Location Data

- Drone location found relative to the camera position
- Data points are converted to angle and distance and North-East-Down
- For debugging, our `v_sub.py` plots the streaming data onto a Matplotlib 3D plot
05
DEMO
Special Thanks to:

Navsea, Project Sponsor

Alan Jaeger, Navsea Representative

Dr. Yogananda Isukapalli, CE Capstone Project Instructor

Aditya Wadaskar, TA

Kyle Douglas, TA
THANKS!
Questions?