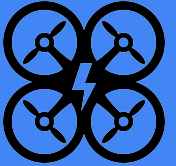# Drone Energy Delivery

Team: DEC1705
Advisors: Dr. Geiger & Dr. Chen

Justin Howe - Team Leader, Avanish Kuntla,
Dustin Reed - Communication, Jeffery Schons - Webmaster,
Chidike Ubani - Key Concept Holder

Outline

- Problem Definition
- Drone Choice/Details
- Design Details
- Testing Methodology
- Implementation
- Conclusions

The Problem

- Powering objects not connected to grid
- Sensor packages (Nodes) that require small amounts of power and may be in very remote locations
- Can we deliver power from a location where power is inexpensive to where it is more valuable?
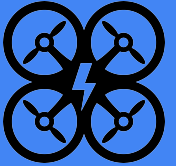
Purpose of Drone Energy Delivery

- To demonstrate ability of a quadcopter to deliver power to a system of nodes
- Bridge the gap between electrical grid and a remote node
- To support wireless sensors or stations that can not easily be connected to a power grid (IE untethered)

Functional Requirements

- Automate the control of the drone
- Autonomously send drone to node using GPS navigation
- Once drone is above target node, use visual guidance for precision landing
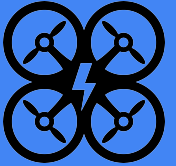- Charger will activate upon landing/connection of contacts
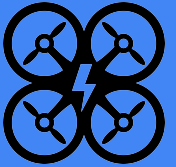
Selecting the Drone

- Manufacturer provided SDK
- On board stabilization and navigation suites
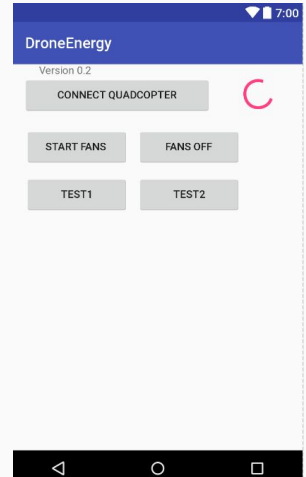- Within budget
- Downward Camera

Decided upon the DJI Phantom 3 Standard

Connecting to the Drone

- The drone is controllable through commands issued using the SDK
- Wireless ad-hoc network produced by drone controller connects to a nearby phone equipped with companion app via WiFi
- Application must authenticate with the manufacturer's servers
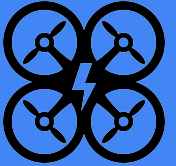
Virtual Stick Flight Controller

- The Virtual Stick class emulates the input of the drone's remote controller
- Allows us to control linear movement in three dimensions as well as rotational movement of the drone
- These controls can be automated to allow autonomous control of flight characteristics
- Commands continuously issued to the quadcopter at 20HZ

OpenCV Desktop

- Using a camera, capture an image and analyze it to determine control inputs
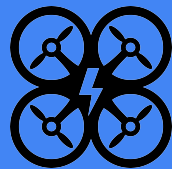- Uses OpenCV to find red circle
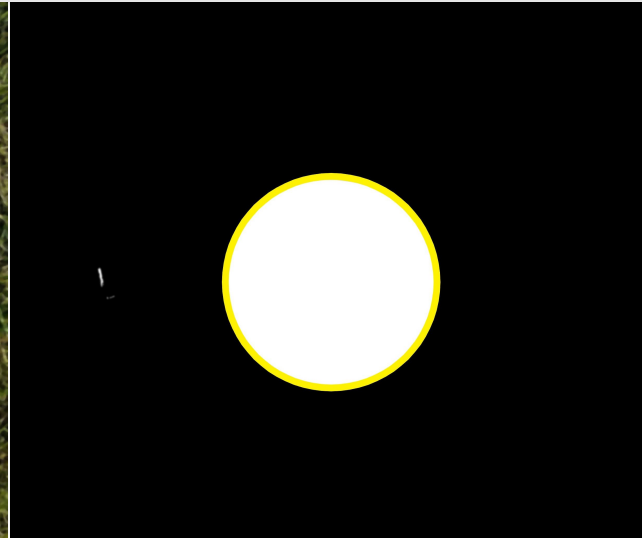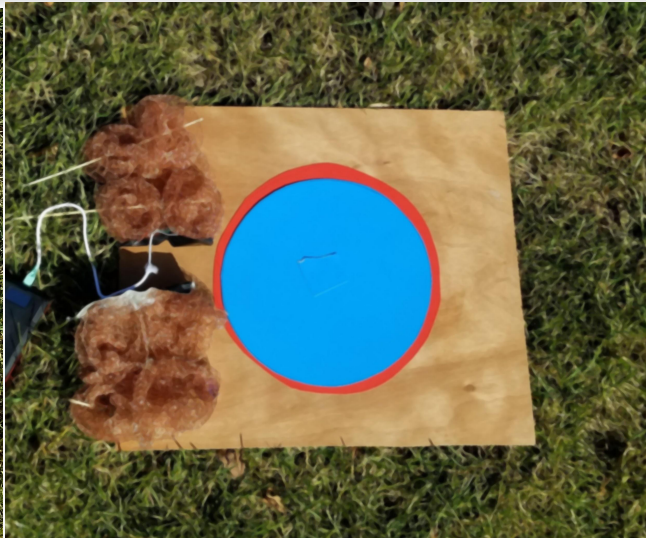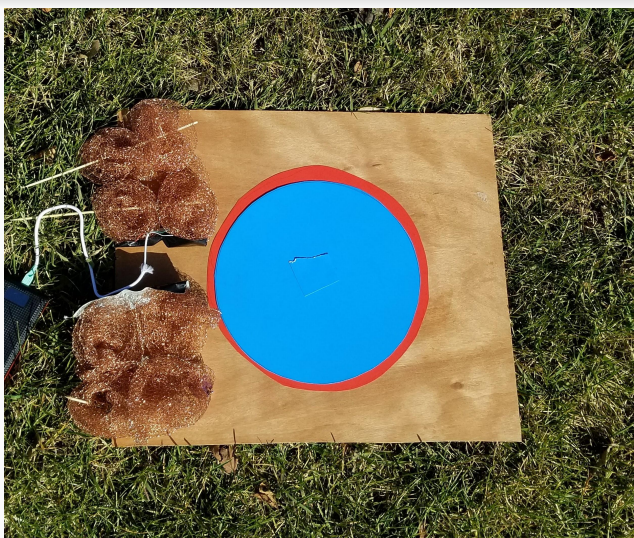- OpenCV provides X and Y position of the circle midpoint
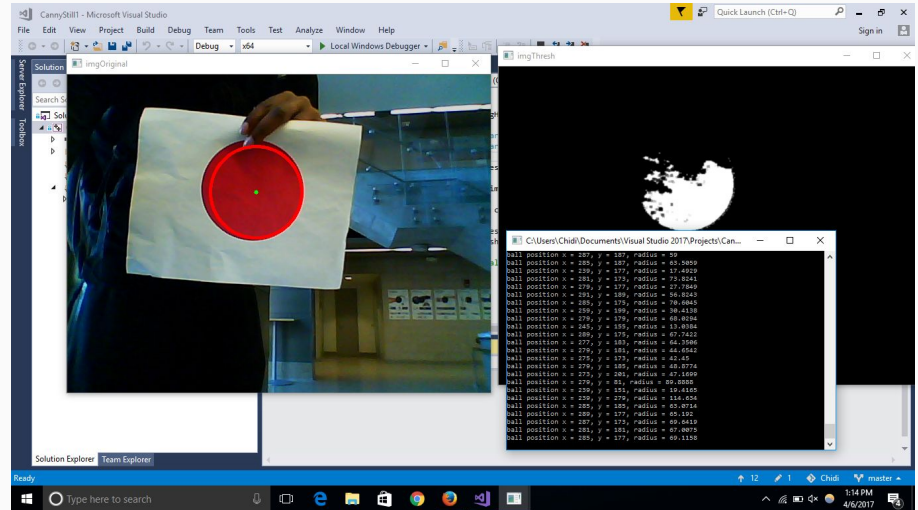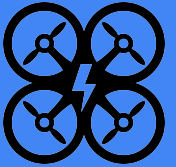
OpenCV Example in Android Environment

- Find largest circular pattern of pixels in the color range
- Return a point (X,Y) and an integer of the radius (in pixels) of the circle found
- Allows us to find location of pad, and estimate altitude
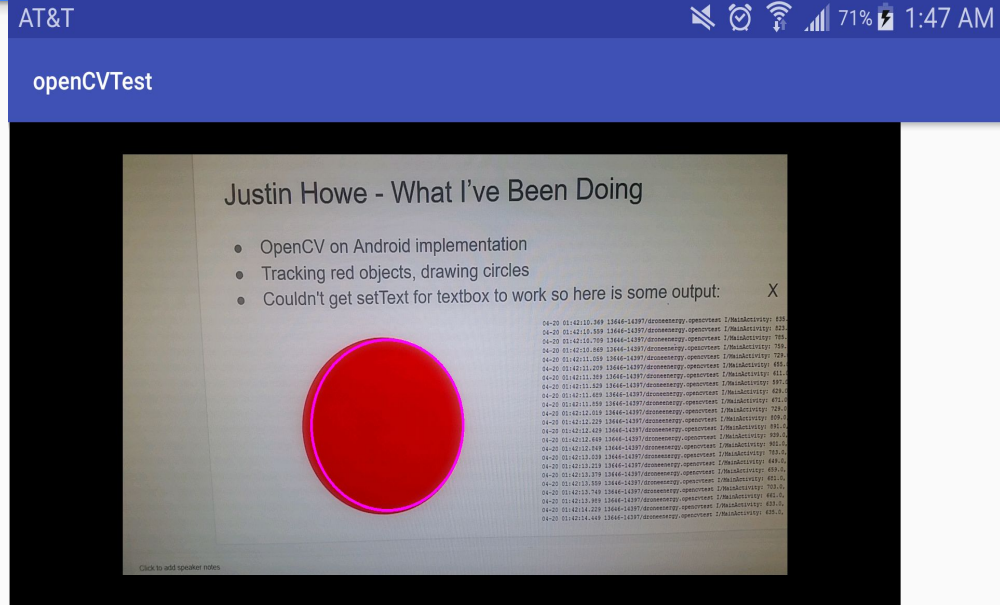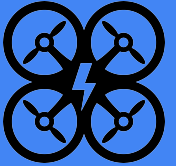
Visual Analysis Pipeline

Integrating the Components

- Using the SDK, start a waypoint mission to coordinates of node
- Once over the node, take video using onboard camera
- Using OpenCV locate the node in image from video and find error in location
- Feed error into PID controller to calculate control input
- Control input is sent to drone and movement is made to correct location error
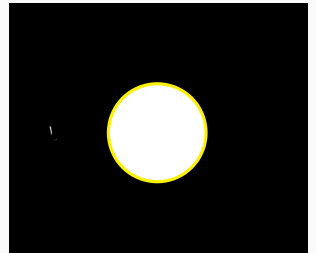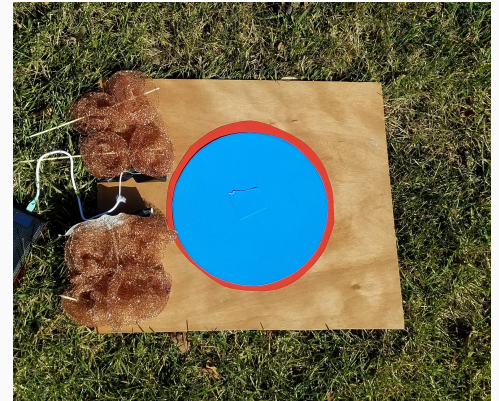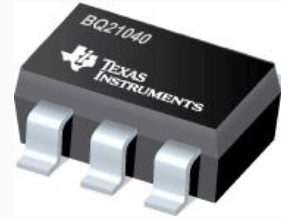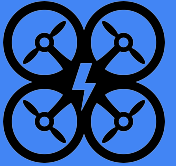
Charging unit

Original design involved creating our own charging unit from scratch.

- Copper wool used as the contacts
- The contacts are attached via broken out USB connectors
- Any device that connects via USB can be used
- USB portable charger receives power from the charger mounted on the drone

Testing Process

Autonomous landing proved much more difficult than anticipated.

- Initial testing simply giving basic command inputs
- GPS guided flights
- Analyzing images during flight
- Handing off input to PID controllers
- Tuning PID controllers during flight
- Implementing a basic landing controller
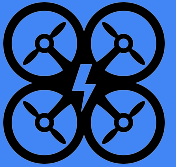
Waypoint Missions

- Waypoint missions are a functionality included in the DJI mobile SDK
- They use a combination of GPS coordinates and altitude to enable the drone to autonomously fly in a set path.

PID Tuning and Tracking

- The error calculated using the position of the tracked object and the center of the frame is sent to PID
- PID output is sent to quadcopter
- Quadcopter makes the motion required to recenter on target
- Analyzing frames at 20 frames per second

Landing

- Changed tracked object to blue for more reliable tracking
- Two stage PID controllers
- Altitude is found using the object size
- Altitude modifies the values given to the PID controller
- 2nd stage PID takes control around 9 ft above the target
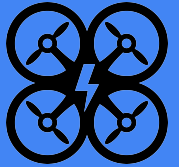
Increasing Accuracy

- Analyzing the last known center of the circle with current image
- Using onboard accelerometers to promote higher stability
- Check difference in GPS location of the drone to enhance stability
- Slower, but more reliable and accurate

What We Have Accomplished

- Utilized SDK to automated drone control
- Drone navigation using GPS
- Retrieving photos from onboard downward facing camera
- Automated Landing with Android based OpenCV and PID controllers
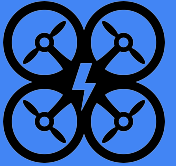- Charging system allows the drone to deliver energy through contact

Challenges

- Location to be able to test
- Dependent on clear, calm weather
- Drone ground effect is substantial
- DJI requiring updated SDK to power on drone
- Errors in Chinese
- None of us are Electrical Engineering majors

What we would do differently

1. Custom/different drone
   - DJI SDK is terribly commented/documented
   - Updates involve moving functionality without documentation
   - Controller must always be connected, even if automating
   - Significant delay between photos being taken and received
2. Purchase product designed to be an automated landing platform
   - Very expensive, but involves less setup and flight control
3. More time to work on the charger
   - Although functional, our charging unit could be better
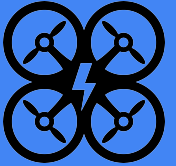
Completed Tasks

Justin Howe: Drone Connectivity, PID Control, OpenCV Android, Landing Control

Avanish Kuntla: GPS Location, Camera Control, Flight Controller

Chidike Ubani: OpenCV Desktop, Image Server

Dustin Reed: Waypoints, Charging Prototype, Camera Control, UAV License

Jeffery Schons: Website, Charging Unit, Control Server

Conclusion

- We have created an energy delivery system that utilizes an automated drone
- In this system we are able to direct a drone to fly to a node and deliver a charge to a connected device
- This project has numerous applications such as charging nodes in remote areas and extending access to the energy grid
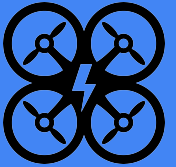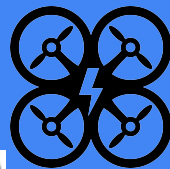
Conclusion cont.

- Is this methodology feasible?
  - It is possible, but not efficient
  - Takes too much money or manpower to implement
- Other Applications
  - A self charging drone/pad would be great for hobbies, or situations where you just need to resupply the drone's internal battery

Questions?
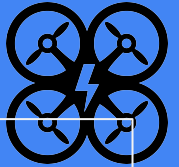
PID Control

$$u(t) = \mathrm{MV}(t) = K_{\mathrm{p}} e(t) + K_{\mathrm{i}} \int_0^t e(\tau)\, d\tau + K_{\mathrm{d}} \frac{de(t)}{dt}$$

Proportional, Integral, Derivative Control

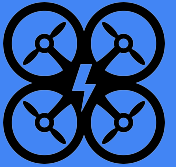Used to dampen error in landing

Error - Difference between our setpoint (center of circle) and current position

K_P - Provide proportional output relative to error value

- Non-zero error required to drive drone leads to oscillation

K_I - Sum of error over time used to dampen steady state error

K_D - Slope of error over time used to improve settling time
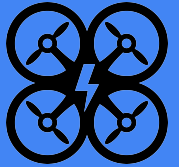
The Drone

DJI Phantom 3 Standard

Priced at $500

Payload carry weight of 800g
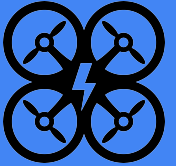
Compatible with Android SDK

Waypoint App

- Locate will place the drones current location on the map (In China by default)
- Add waypoints by clicking app, or by capturing the drone's current location.
- Config allows you to configure flight settings
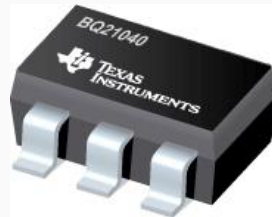- Upload the mission to the drone by clicking upload, then start the mission with start.
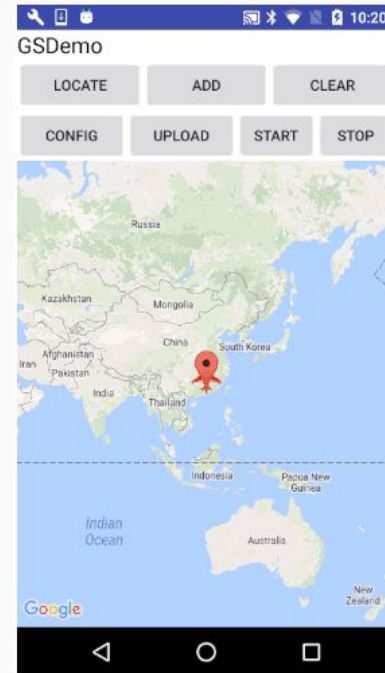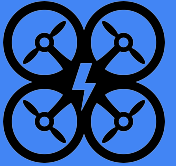
Charging Unit (idea from last year

- Node is powered by a simple lithium ion battery pack
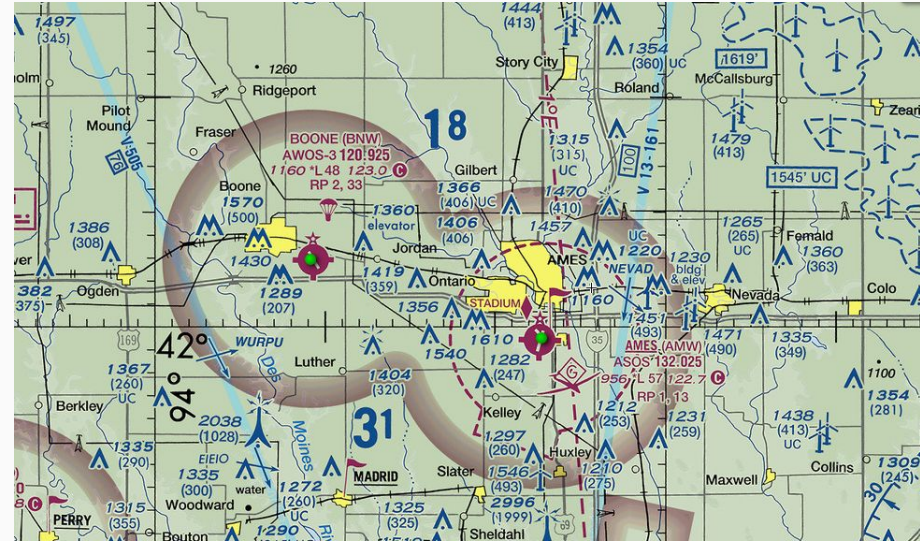- Regulate the charging process with TI BQ21040 control chip
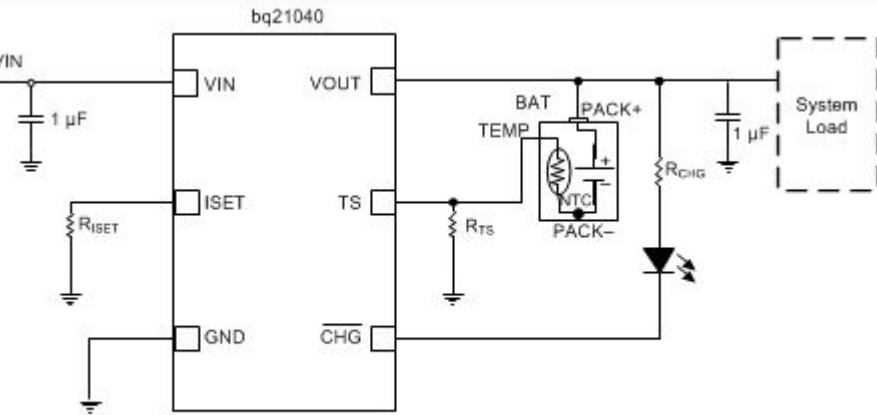
Copyright © 2016, Texas Instruments Incorporated

Flight Regulations

- Drone is owned by university, which is a business.
- Someone needed to become a registered UAV pilot
- Needed to determine acceptable areas to fly drone

Testing Automation of Landing

- This flight test uses our openCV implementation
- It attempts to correct its landing but is influenced by the wind/ground effect.
- The delays in the image analysis caused harsh inputs and an imprecise landing.