Field 3D PTV v1.0

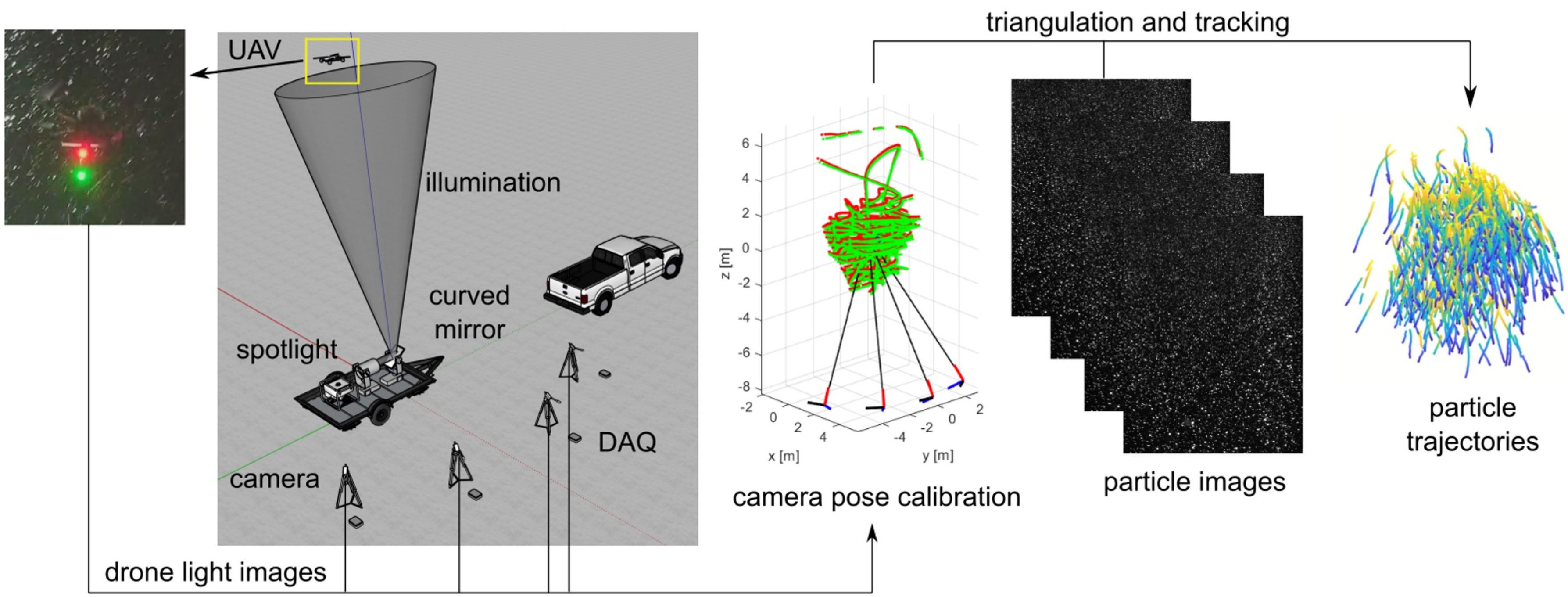
Flow Field Imaging Laboratory, University of Minnesota

Outlines

- Product overview
  - Hardware overview
    - Detailed description of key components
  - Software overview
    - Detailed description of key processing method
- Operational procedure
- Demonstration cases
  - Confetti
  - Snow
  - Pollen
- Troubleshooting

Product Overview

Field 3D PTV System v1.0 Overview



- Hardware: a WiFi-synchronized four-camera multi-view imaging system for data recording and a drone with two-color LED spaced in a fixed distance for calibration
- Software: Easywand code for calibration and OpenLPT for 3D particle reconstruction

Hardware Overview

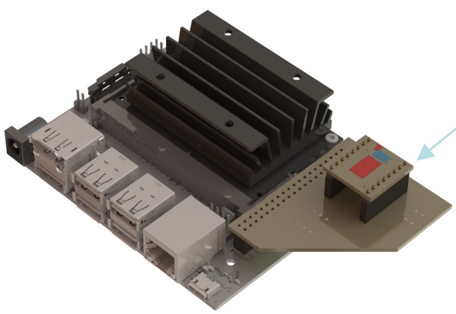
- Main hardware modules
  - Imaging module (four boxed units): cameras, gimbals, processing units, tripods
  - Central computer: laptop control data acquisition and camera position using wifi communication.
  - Calibration module: drone
  - Lighting module (optional)

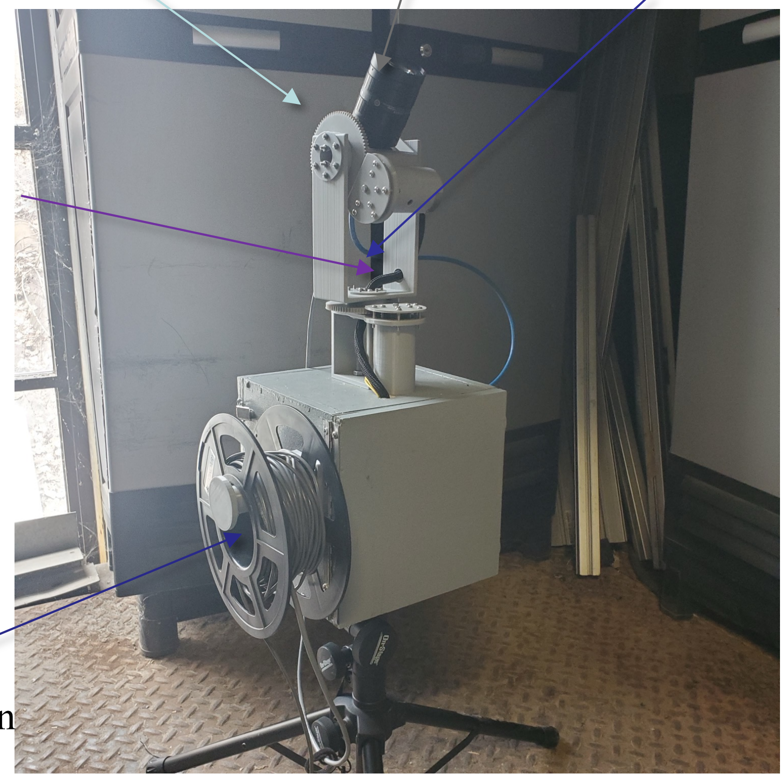
Imaging Module

Four identical boxed imaging units

- Inside the imaging unit
  - Image acquisition system:
    - Camera (2.8mp, 95fps, four cameras are synchronized through cable)
    - Lens (16mm f/1.4)
  - Stabilizing system
    - Two-axis (yaw and pitch) 3D printed gimbal system
    - Position Sensor (GPS)
    - Alignment Sensors (Accelerometer and Magnetometer)
  - Processing and storage system:
    - Jetson Nano (with custom PCB)
    - Wifi adapter
    - 500 GB SSD
- Container and tripod

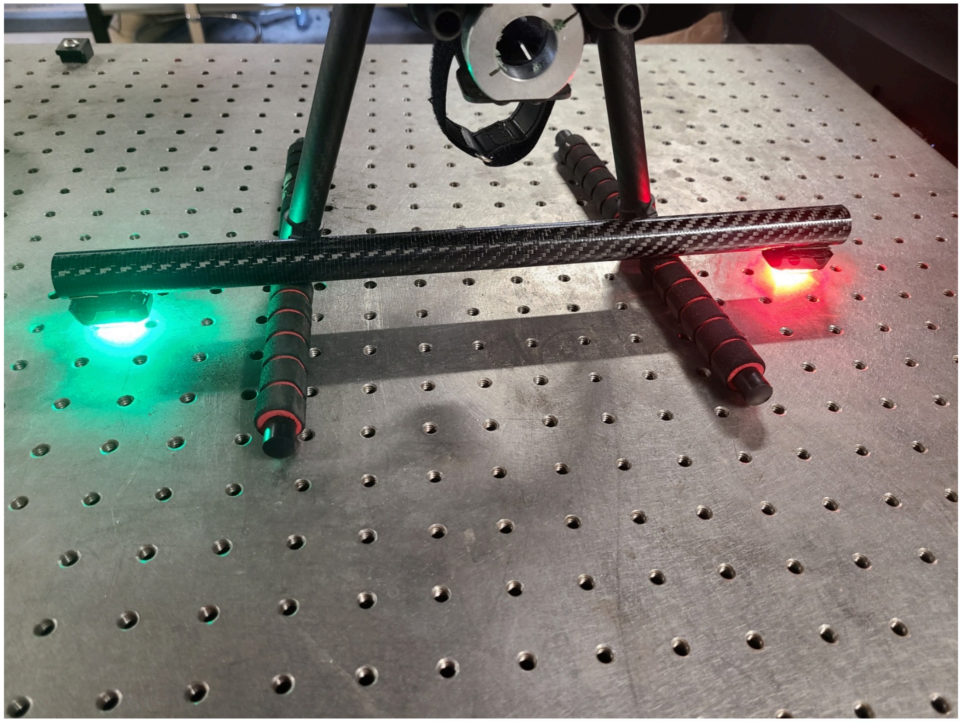
Imaging Module

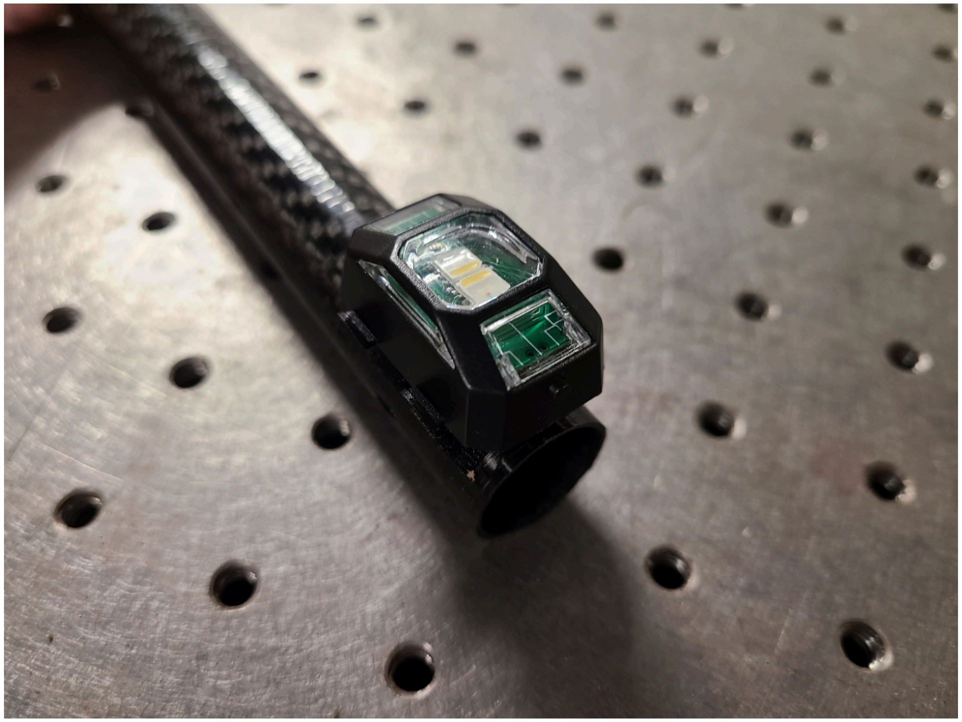




Calibration module

- A bar with red and green lights for calibration
  - Note: bar should be horizontal, and lights should point down
  - Use drone if needed





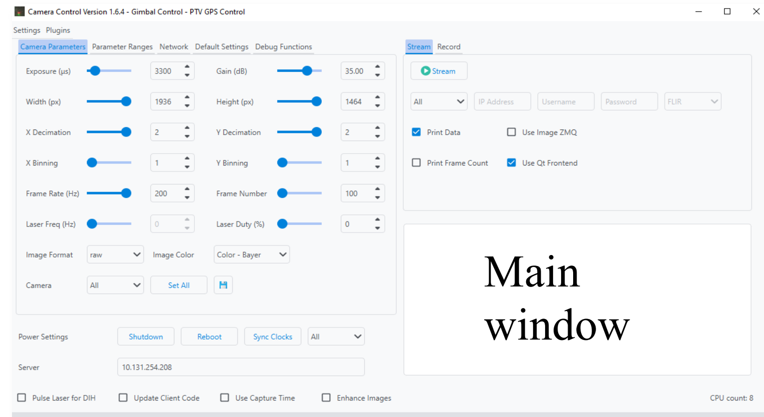
Central computer

- computer

Software Overview

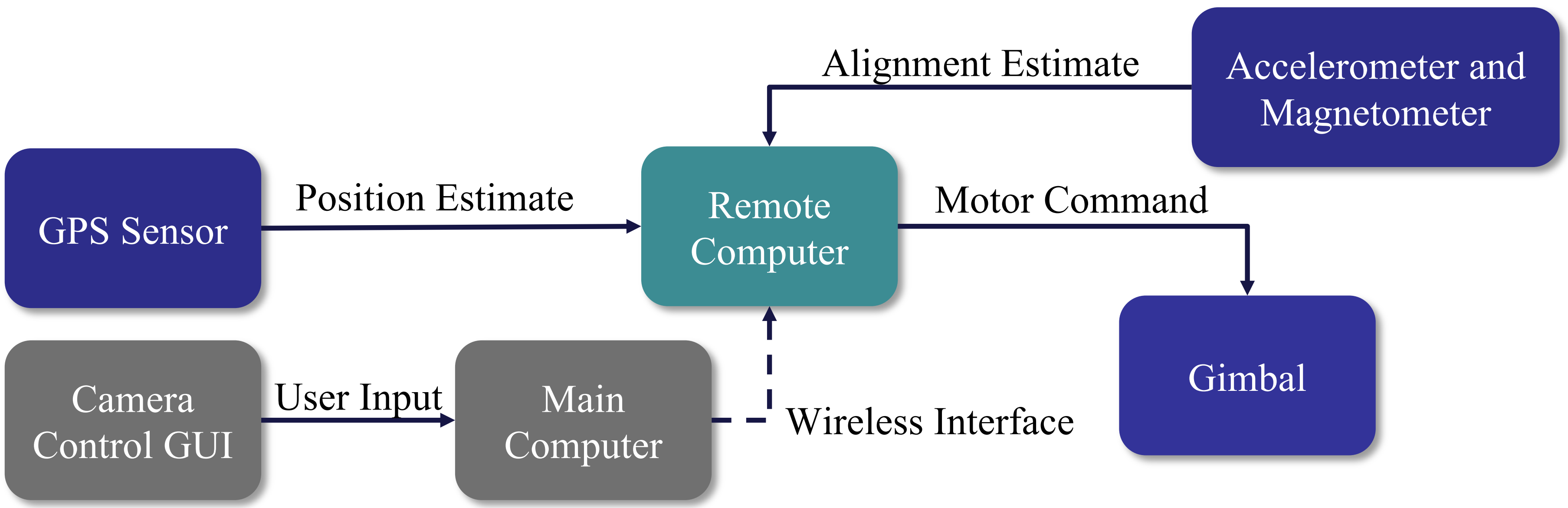
Camera Recording Software

- Custom software designed for controlling camera recording
  - Parameters for recording can be edited and saved easily
    - E.g… Shutter speed, Exposure, etc…
  - Can choose which devices belong to the "network" of camera systems
  - Streaming enables easy testing and viewing of what will be saved to the remote computers
  - Fully supports two different camera types using NVIDIA Jetsons (FLIR and Pi HQ)
- Software design intended to be easy for any user
  - Recent updates include
    - Keyboard shortcuts for easy control in the field
    - New recording progress interface
    - Faster and more reliable code improvements



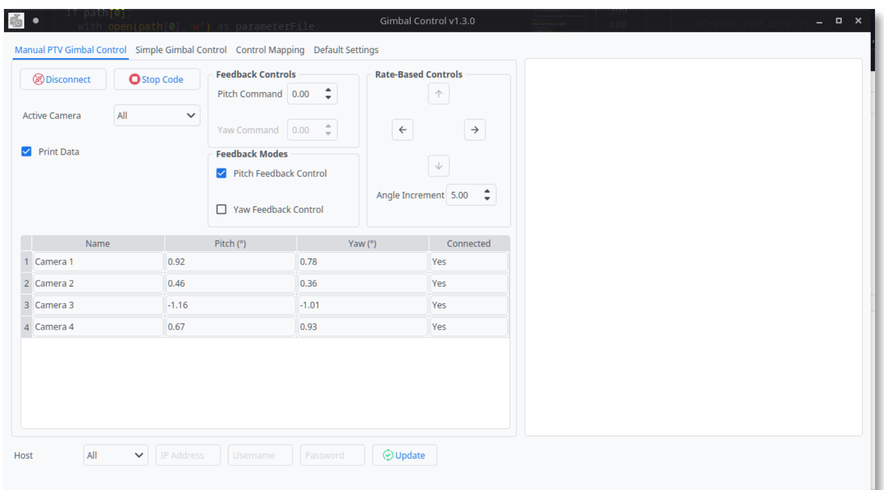
Gimbal Control Software

- Custom software for changing camera alignment
  - Commands to remote computers are sent by the user directly or are calculated from GPS position estimate
  - Feedback controller done using accelerometer (pitch) and magnetometer (yaw) measurements



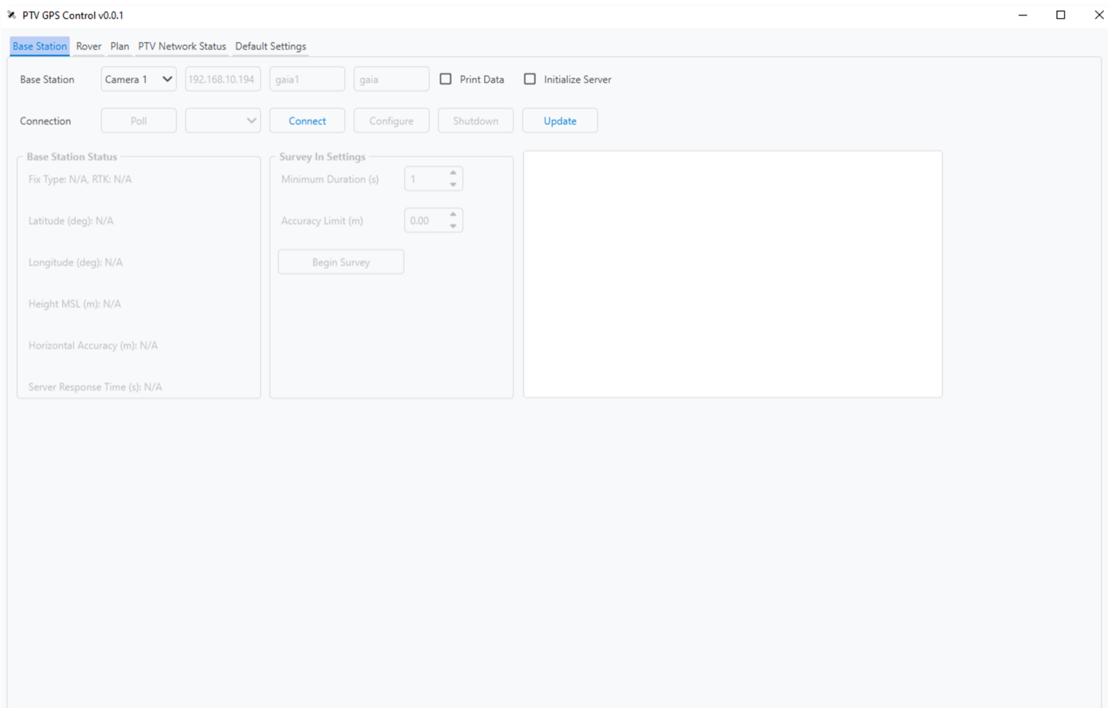
Gimbal Control Software

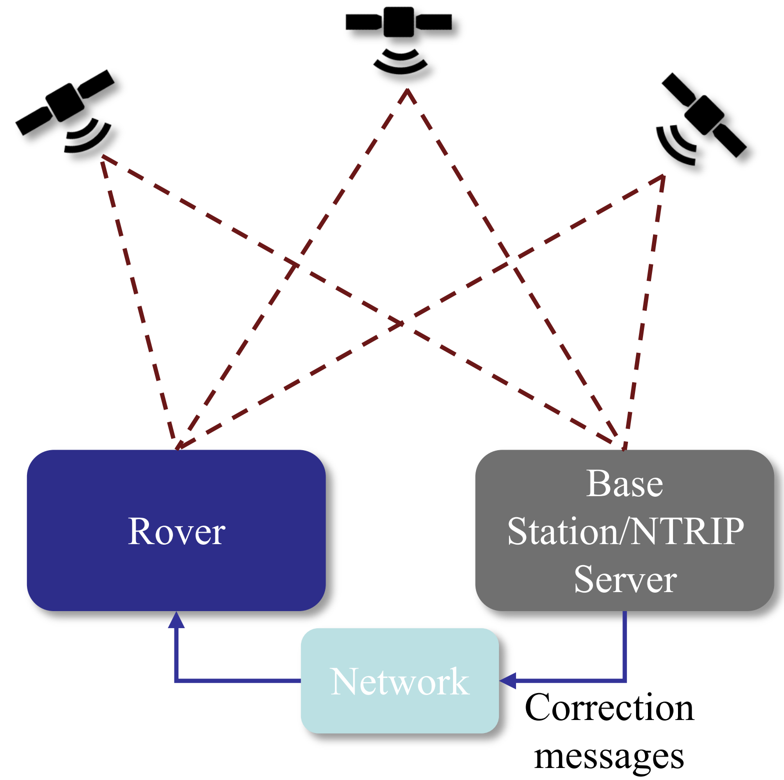
- Completely wireless control of camera alignment
  - Cameras can be controlled independently or all together with minimal lag time
  - Pitch and Yaw can be controlled automatically or adjusted manually with seamless switching of feedback modes
  - Low latency response rate to user inputs
  - Can run with or without the use of motor speed control



GPS Control Software

- Custom software designed for interfacing with GPS receivers
  - Designed to make it easy for the user to interface with the receivers
  - Can be used to automatically provide feedback control to the gimbal systems, depending on the planned positions





Operational procedure

Overview

1. Set up drone and begin RTK GPS process
2. Set up waypoints for drone (if not done in advance)
3. Set up wifi
4. Position camera tripods and focus/aim cameras
5. Connect Jetsons to cameras and power on
6. Connect acquisition laptop to network, check Jetson connectivity
7. Run test acquisition
8. Begin drone flight and begin camera capture at 10Hz simultaneously
9. Run calibration, ensure no issues
10. Capture dataset images

Powering on drone

- Make sure battery is securely strapped in
  - If this flops around during flight, that’s bad...
- Balance the drone
  - Holds fingers under opposing rotor motors, shouldn’t tilt
- Connect battery carefully
  - If it shorts briefly, then just disconnect and do it again
- Should make a few beeps and the lights on top of the GPS will flash
  - When green blinking, that means its ready for flight
- To get green lights
  - Turn on RC controller, make sure all switches are “up” and no errors
  - Might need to hold for 1-2 seconds the “fix” button on the GPS unit
    - This is a safety switch, some of the drones have this
  - Should go green usually in about 10 seconds, maybe longer, once GPS fix is obtained

Drone reminders

- To get out of flight plan
  - Toggle flight mode switch on controller back and forth
  - Drone will exit flight plan
  - Be ready to do this when beginning flight plan in case of unexpected issues
- When manually flying, make sure the drone is in Loiter mode on the controller
  - There are 3 modes:
    - Loiter: keep lat and long, and altitude
    - Altitude hold: only keep altitude
    - Manual control: don’t automate lat, long, or altitude

Connecting to Comfast Wifi from laptop

- Nvidia Jetsons will automatically connect
  - If not connecting, turn off and then on again
- Network name: UMNField_5
- Password: FieldW1F1!
- Website: http://192.168.10.1/computer/login.html
- Website password: admin
- Go to Users
- Will see all devices
  - Only laptop listed here in demo

Acquisition codes

- Located in
  - https://github.umn.edu/HongFlowFieldImagingLab/GAIA-project/tree/master/CameraControl
- record_multicam.py
  - This is the main script to be run from the laptop
  - Sends commands to Jetsons over network in parallel
  - Executes record_singlecam.py with options specified
- record_singlecam.py
  - This runs camera capture
  - Takes several options with argparse
  - Saves images in .raw format, and saves two txt files with metadata
- parameters.py
  - Contains a few options, filepaths, IP addresses
  - The only thing that MAY need to be changed here are the "local" paths at the bottom, depending on the master computer running the codes

How to acquire synchronized multicam images, part I

- In terminal or powershell:
  - python record_multicam.py
- Terminal will prompt to input several options
  - Run name, fps, number of frames, gain, color mode, wireless option
  - Run name will be appended with fps
- After prompts..
  - Automatically will copy record_singlecam.py over to Jetsons (in case of any updates)
  - Automatically will clear remote folders of matching run name
- If wireless sync specified [y]
  - Automatically sends command to Jetsons to synchronize their clocks with master computer IP address
  - Sync offset will be returned
    - If not <0.01 seconds, then redo sync, may simply need to wait a few more seconds
  - Press enter to begin acquisition after 10 second delay

How to acquire synchronized multicam images, part II

- If wired
  - Connect GPIO cables to cameras
  - Join all GPIO cables at center node, connecting red to red, black to black
  - Code will set local computer as "primary" and remote Jetsons as "secondary" which will be triggered to start acquiring when primary does
  - Code will send commands to remote Jetsons to execute their record_singlecam.py code in "secondary" mode
  - Wait at least 5 seconds, then press enter to start acquisition
- After recording complete
  - Should return output message from remote Jetsons with effective fps reported

Image acquisition settings

- Calibration
  - 10 fps, 3000 frames (to be safe)
  - Exposure time: 40 microseconds if f/1.4; 400 microseconds if f/22
  - Gain: 15
  - Color: BayerRG8
- Particle images
  - 80 fps, 2000 frames max
  - Exposure time: <10k microseconds (check this on laptop tho), Gain 15
  - Color: BayerRG8 (do NOT use Mono8, it slows acquisition)
- After complete, do NOT just unplug Jetson
  - This can result in lost images
  - Instead, execute remote_singleCommand.py script
  - This tells Jetson to begin shutdown in 1 minute

Calibration procedure

- (Assuming drone is in at least RTK float mode, ideally RTK fix)
  1. Start record_multicam.py code and input options
  2. Do not press enter to begin recording yet
  3. Turn on drone LED lights
  4. Single press to turn on, additional single presses cycle through modes
  5. Long press to turn off
  6. Manually takeoff using RC controller, and leave in Loiter mode a few meters off the ground
  7. Press enter to begin image acquisition
  8. Click “Auto” in Mission Planner to execute flight plan

Troubleshooting

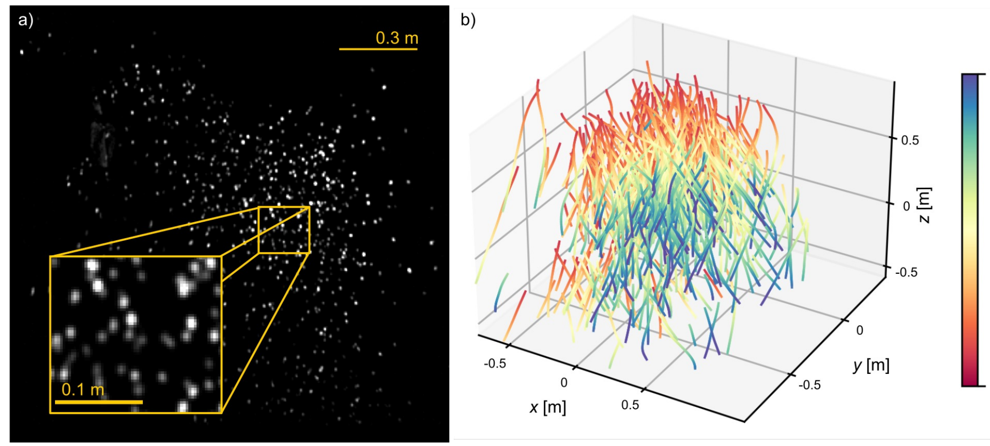
Troubleshooting

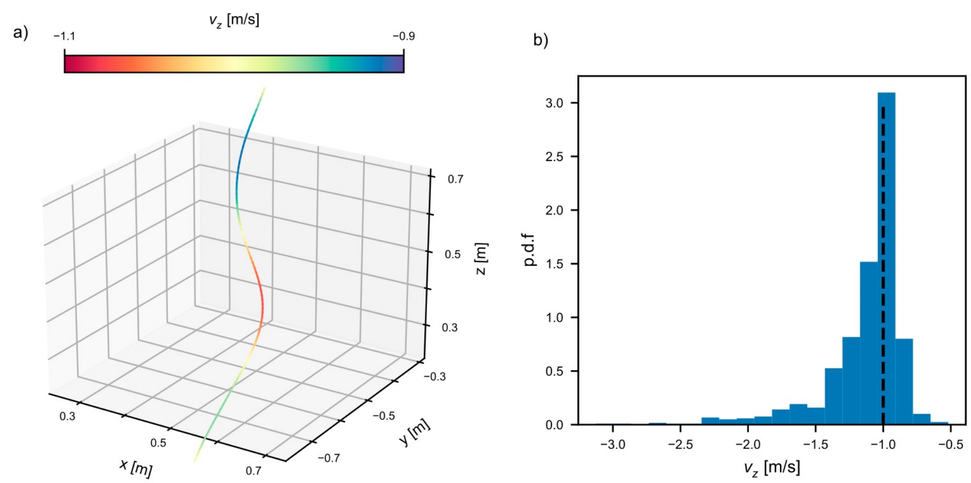
- If error pops up during code execution
  - Easiest thing is to unplug and reconnect each camera
  - This will interrupt any code running on the device and also reset camera
- If error is related to "acquisition frame rate limit exceeded"
  - Unplug and reconnect camera
  - If this doesn't work, connect camera to laptop, and open SpinView
    - Check fps limit
    - If not full 95 fps, then disconnect in SpinView, then unplug and reconnect camera again
    - Fps limit should be back to normal
- Network issues
  - Check wifi network website
  - Reboot the Jetson(s) not connected by unplugging and reconnecting battery

Example Cases

Case I: Confetti Settling

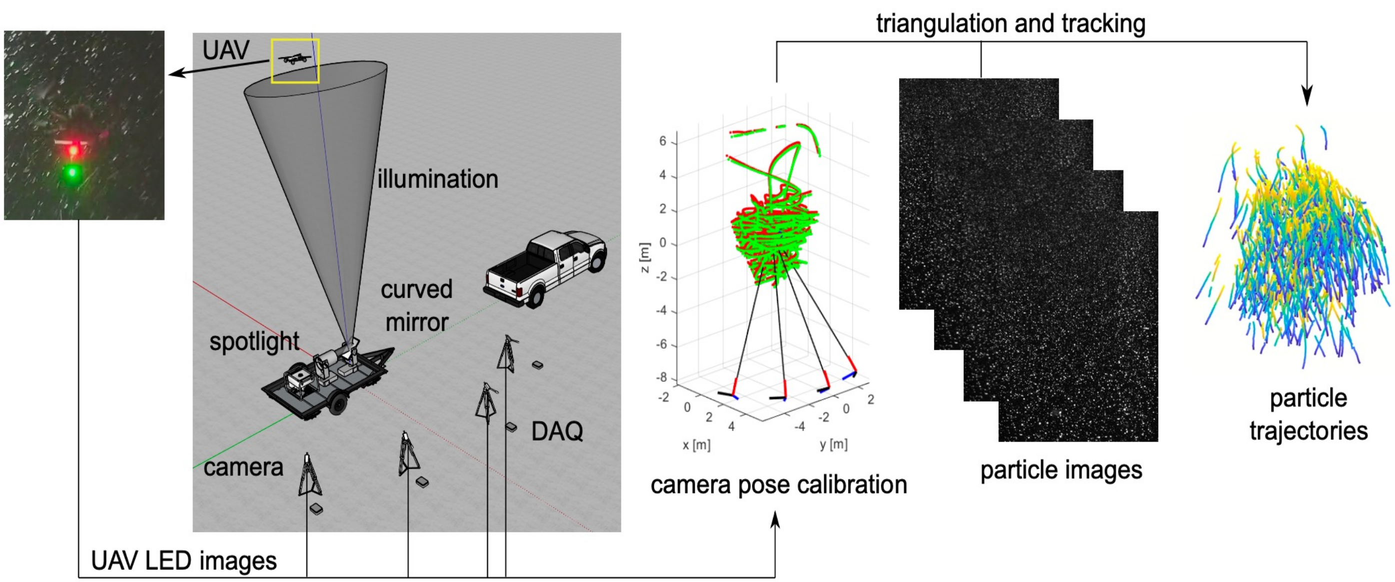
- For concept validation
  - 93% trajectory yield (particle count)
  - Fine scale motions well captured
  - Fall speed agrees with individual measurement

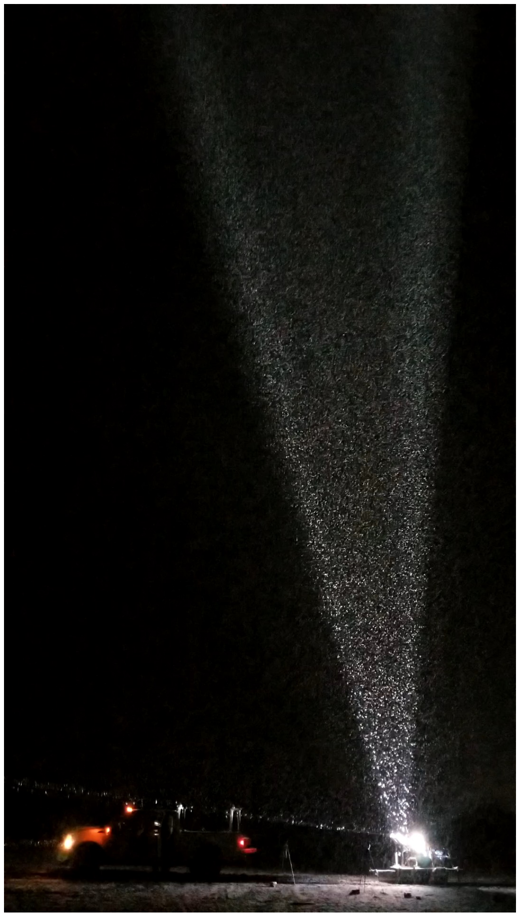




Case II: Snow Settling

- Investigation on snow settling under different atmospheric conditions
  - Low turbulence (snow morphology play a role)
  - Imaging 4 m x 4 m x 6 m volume ~ 10 m above ground
  - Spatial resolution 6.3 mm/voxel; temporal resolution 200 Hz
  - Duration: in total 26 50-second sequences

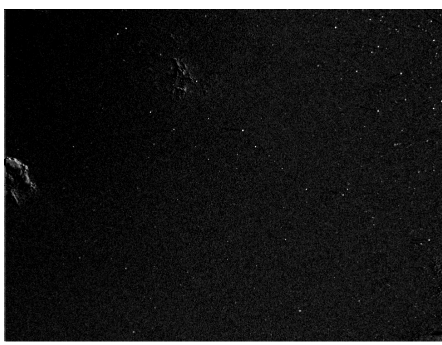




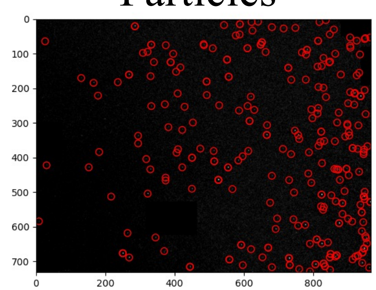
Case III: Pollen emission

- Investigation into pollen particle emission from a cottonwood tree
  - Deployments of the system done over Summer 2022 during a time of high pollen density
  - Used 3D PTV system to obtain particle tracks of cottonwood pollen during the daytime
  - A better understanding of pollen emission can potentially improve forecasting for people with allergies

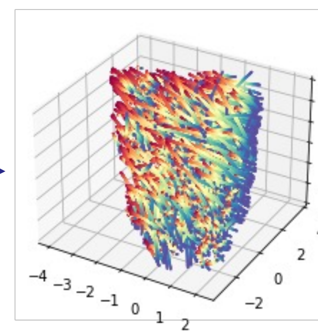
Enhanced Images



2D Recognized Particles



All Particle Tracks



50 Particle Tracks with tree