Kevin Dorosh (Electrical Engineers: Richard Preston, Elyse Cooper, Ryan Stocking)
December 21, 2018
Functional Spec – Team Mahogany

Disclaimer: This is a complete **software-only** functional spec and brief technical spec (technical spec is subject to change). As far as I know, nothing has been omitted, but feel free to contact me at kevin.dorosh@tufts.edu if anything looks awry.

Overview

Team Mahogany is working on developing a black-box hardware attachment and software solution to augment the data-collecting abilities of unmanned vehicles. This attachment will be light, compact, and communicate over radio to a USB device that provides information for our software to process and display, in real-time.

Problem Statement

First responders are often ill-equipped to handle dangerous situations safely. While this has been a given in the field for some time now, it no longer must be. With the advancements made in the casual UAV space, it has become affordable and easy for non-specialists to leverage drones to gather valuable information without risk to themselves.

We will work with drones to augment, hasten, and hopefully altogether replace boots-on-the-ground information gathering in post-disaster scenarios.

Customer

Sponsor: Prof. Karen Panetta.

Customer Requirements:

- Compatible with 3DR Solo Drone
- Supports picture, thermal, and radiation sensing
- Accurate GPS locations
- Real-time transmission and display of the data
- Intuitive UI for data visualization

In addition, we have received helpful customer-style input from Luke Harwath, the Director of Communications for Help.NGO – Global Dirt since 2011

Discussions with Luke Harwath revealed to us the capabilities and limitations of the Geiger counter. We learned that it's vital to be close to the ground to obtain reliable data.

For this reason, we have pivoted from a radiation reporting focus to relying more heavily on using standard and thermal photos for our data gathering, synthetization and reporting, while also supporting radiation tooltip overlays (diagrams to follow). As a stretch goal, we will implement a topographic map-style overlay for radiation in addition to tooltips.

In addition, therefore we are focusing on a black-box attachment system that functions separate from the drone. In scenarios when we can afford to lose speed for accuracy, the same system can be attached to a ground-based unmanned vehicle for data collection.

Scenarios

#1 (Extreme Case) Help.NGO director Luke has received terrible news – the Russians have hacked American nuclear facilities and blown them to bits (like Stuxnet). As a headline news story, Luke is pressured to give prompt responses regarding the scenario, but he doesn't have much information to draw from. Time is of the essence.

To be safe, Luke has evacuated everyone within 50km of the accident, despite the unnecessary inconvenience many will experience as a result. To hasten the response process, Luke has purchased our drone-attachment device and software and given it to his first-responders. They quickly (and safely, from a distance) map out the area, and determine that the damage is not very severe. The danger is limited to within 5km of the accident, and Luke informs the media and evacuated citizens.

#2 (Common Case) California has been struck by a long overdue earthquake. The damage is severe, as many buildings were not constructed to current earthquake code. Help.NGO director Luke is again assigned to respond to the scenario, and needs to quickly locate civilians in danger. Fortunately, he can deploy numerous drones with our thermal/standard imaging solutions to discover trapped people under the buildings in record-time.

Even though there is no radiation or danger to first-responders after the disaster, there is still value to our solution. Increased response time and cheaper/more deployable drones (as opposed to helicopters) will save lives. That cannot be understated.

Non-Goals

We are not trying to invent, modify, or change the way drones or other unmanned vehicles are piloted. Mission Planner and other open-source software is available and growing, and we would like the advancements to seamlessly integrate with our solution. Therefore, our payload must be compact and light.

MVP (Minimum Viable Product)

- Stitches thermal and standard images together with GPS data and overlays onto map
- Real time
- Communications from hardware to laptop over radio
    - Radio signals converted at site and read into machine via serial port

Future Work/Stretch Goals

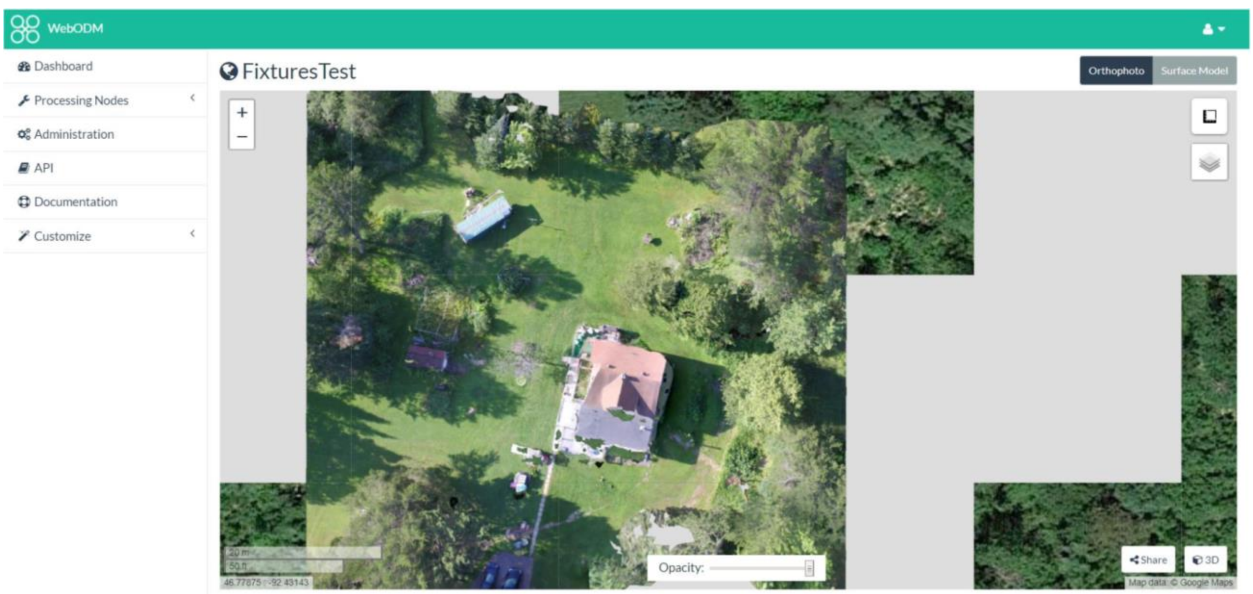
- Topographic style overlay for radiation information
- Package Web UI into native desktop UI using Electron
    - https://github.com/electron/electron

Flowchart

Mission Planner is the drone route-coordinating software. Our software solution, incorporating WebODM, will also run on this laptop.

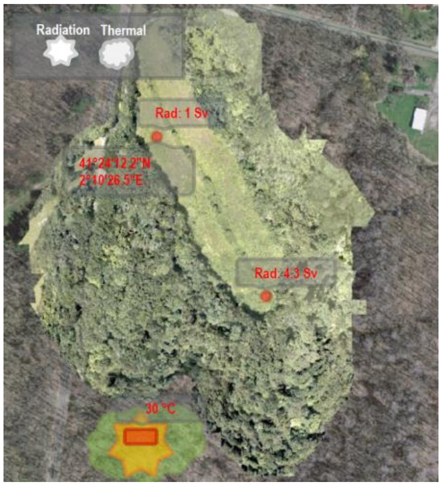
Screen by Screen Specification

Map Overview Page (Updated in real-time)



The other pages in the sidebar on the left will be largely phased out

Sample tooltips overlay for radiation sensing

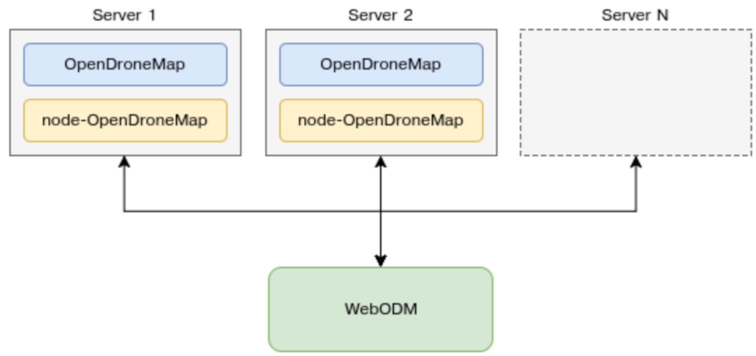


Stretch Goal: Sample/inspiration overlay for radiation based on topographic map

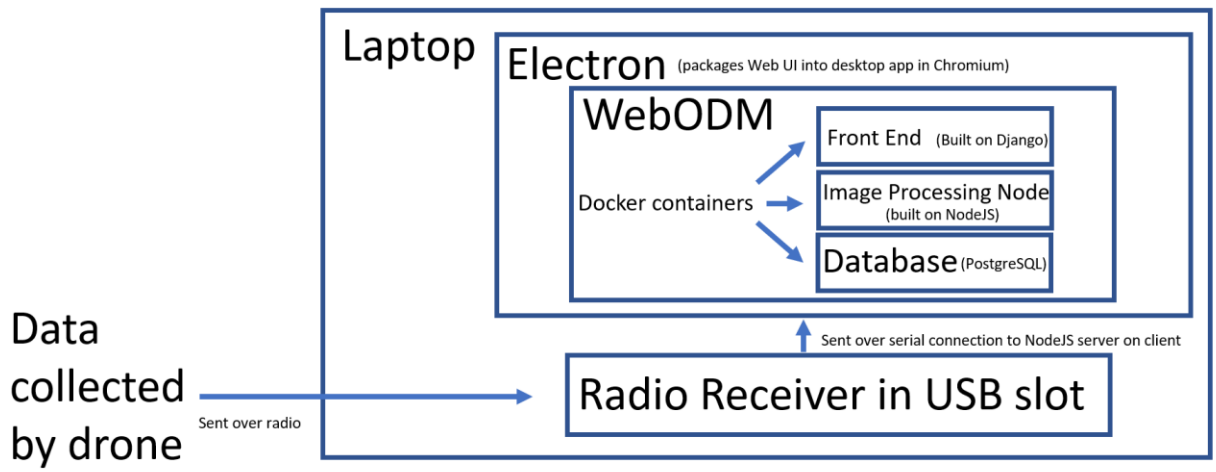
(Brief) Technical Spec

- Hardware
  - Not an important part of this software spec
- Software
  - Ardupilot
    - The firmware that controls the drone
  - MissionPlanner
    - The software that plans “missions” and sets routes for the drone
    - Think “operates” the drone
  - WebODM
    - A web UI and library for post-drone flight image processing
    - Composed of three parts
      - Database (PostgreSQL)
      - Image Processing Node(s)
        - Built off node-OpenDroneMap framework
      - Web UI (Django)

WebODM structure:



Our Modifications and High-Level Design (software-only)



Tools/APIs

- Django
  - Python-based web framework
  - Mostly used for the UI in WebODM, which we will repurpose and augment
- NodeJS
  - Open source server framework based on JavaScript
  - Largely used for image processing in node-OpenDroneMap
- Docker
  - Container software
  - Virtualizes machines at the filesystem-level instead of the entire OS/machine (like a VM)
  - Eases development and deployment of web-based services
- PostgreSQL
  - Object-relational database management system
- Electron
  - Packages Web projects into native-desktop applications
  - You can develop once and deploy to Windows, Mac, and Linux natively!

Risks

- **Risk**
    - Scaling
        - Only 1 person working on the software!
        - Most of the technologies being used are new/foreign
- **Mitigation**
    - Appropriate amount of reach goals
    - Lightened course load for next semester
- **Risk**
    - Unknown performance bottlenecks
        - E.g., We have not yet successfully sent images taken over radio to a serial device plugged into a laptop
- **Mitigation**
    - The is our first goal for next semester
- **Risk**
    - UI is being built top down while hardware is being built bottom up
        - Need to be sure we can "plug and play"
- **Mitigation**
    - We have developed a JSON and binary data format we will adhere to

```
{"packet": {
  "jpgMap": {
    "jpgSize": "1800", //in bytes
    "jpgWidth": "240", //in pixels
    "jpgHeight": "240" //in pixels
  },
  "jpgThermal": {
    "jpgSize": "1800", //in bytes
    "jpgWidth": "240", //in pixels
    "jpgHeight": "240" //in pixels
  },
  "GeigerReading": "800", // counts per unit of time
  "coordsGPS": {
    "lat": "4042.6142,N", //40 degrees, 42.6142 minutes N (DDMM.MMMM)
    "long": "44042.6142,W" //440 degrees 42.6142 minutes W  (DDDMM.MMMM)
  }
}}
```

With the binary/image data to follow. We know how large it is since we say up front.

Open Issues

The way WebODM works now:

- Becomes:

Note: We need to repurpose this to run in real time, and construct the images we started with from our hardware signal. Software is being written to prepare for this, but testing this depends on (must wait on) the hardware.

In addition, the image processing provided by node-OpenDroneMap doesn’t handle the sample thermal images well – this may require further (hopefully minor) modifications on our end.