CS CAPSTONE TECHNOLOGY REVIEW
THE AUTONOMOUS ANIMAL LOCATOR
GROUP 33

PREPARED BY

PREPARED BY

AIDAN CARSON, PROJECT ENGINEER

Abstract

This project involves a partnership between a group of six Senior Capstone students and the Department of Fisheries and Wildlife. Professor Taal Levi is the client for the project, and he has tasked the group with a solution to the problem of small animal tracking within the forests near Oregon State University. This project involves the research, construction, and programming of remotely controlled, preprogrammed, and/or partially autonomous drones to triangulate small animals' position when equipped with VHF transmitters. In this document, there will be a discussion of the technologies needed for several aspects of the project. This includes: The platform on which the software will be built, the methodology and technology with which we will be transmitting data back to the researcher, and a look at the different options for fixed wing drones.

CONTENTS

1 INTRODUCTION

This document will discuss several components of The Autonomous Animal Locator with respect to their practicality in expense as well as timeline. The main topics that will be covered in this paper are: Platform, data transmission, and fixed wing drones. Each of these categories are critical to the project's success, and in each category there will be a recommendation as to which software is best suited for our endeavor.

2 PLATFORM

The platform chosen to build the UI on top of will have a large influence over the viability of the product produced. The platform will host the UI that allows for all of the interaction between researchers and the programming of the drone as well as the visualization of the data it produces. The platform needs to be easy for researchers to access, as well as viable in remote locations. There are many factors that influence what would be useful for the client, and each will be broken down along with the benefits and drawbacks of each platform with respect with each factor.

2.1 Website

A website is the most straight-forward solution to the problem of where to host the UI. This allows for researchers to easily access data anywhere they have a computer or phone; however, this means that the client will need to have some sort of internet access in order to be able to use the UI. Whether this means programming the drone, or receiving and visualizing the data. This isn't a viable solution because the researchers will often be outside of easy internet access and will need something that works in any situation.

2.2 Mobile Application

A mobile application is a more viable solution because it could allow for offline receipt, translation, and visualization of the data. It could allow perhaps for offline programming of the drone flight path. However, this needs a map to be able to program on top of and in order to get accurate data of the area there needs to be internet access. This is a separate problem that needs to be addressed regardless of platform. Any platform will use the same kind of mapping technology and so will require either internet access or an already downloaded copy of the area.

The client wants to be able to see the data immediately, so it would make sense to create a mobile app which can take a set of coordinates and translate them into a general area in which the animal resides. Because of a restricted screen size the mobile application might not be the most optimal solution to programming the drone and visualizing data. If a UI is created to work exclusively with the phone, we would need to make whatever software we choose for programming the autonomous flight of the drone work with the mobile application in a user-friendly manner.

2.3 Desktop Application

A desktop application is the culmination of all positive factors from previous solutions with almost none of the drawbacks. It allows for offline access when needed, but can upload the data it collects to a central server for saving history and giving access to third parties. It also has the added processing power to be able to do offline translation of the data it receives into useful information about the animal. The only problem is that when a user is offline the desktop app won't be able to visualize the data on top of a GPS map without connecting to Google Maps, or perhaps downloading the map of the area beforehand. A mobile application would be more likely able to do this, but again would need to have

internet access. With regards to building the software, Electron is a widely used software that allows for cross-platform building of a desktop application [1]. With the offline processing power and screen size that the desktop provides, a native desktop application would be the best option in providing a user interface. This will allow the researchers in the field easy access to both the programming of the drone as well as the processing and visualization of the data once it is received.

3 DATA TRANSMISSION

Data transmission is a vital part of the project, as it is the process that will get the data acquired by the plane back to the server or computer running our translation and visualization software. Any option needs to fit into the life cycle of the drone flight nicely, making the transition for programming to flight to visualization as seamless as possible. There are three different methods that could possibly allow data transmission back to the ground station set up by the researcher. These are: Storing on the plane for physical access once the plane lands, uploading the data to a central server, and real time transmission back to the researcher. Each section will be talked about along with the benefits and drawbacks of each method.

3.1 Store on Plane

This is the simplest to implement approach of the three that will be talked about, but may prove to be the least useful. By storing the data onto the plane and not transmitting any data back to the researcher there won't be any visualization of the data until the plane has landed. Beyond the data not being transmitted for visualization, it also means that the drone's flight would not be transmitted back. This means that if the drone were to have a problem mid-flight and crash then the researcher would not be able to recover it.

This doesn't make sense for many reasons. However, a software that we are investigating, ArduPilot, has a feature that allows for connectivity using radio telemetry [2]. There are many radios that could work well with the plane, and allows for long distance telemetry, such as the RFD 900MHz Antenna [3]. If the problem of tracking the drone is completely solved, then perhaps storing the actual processing locally is reasonable. This would only require a simple SD card or other memory device, which has a low power, weight, and size footprint

3.2 Upload to Server

This would also be a relatively simple to implement mechanic, allowing for the pushing of raw data up to a central server so that our code hosted somewhere in the cloud can process it and return it back to the researcher at the ground station. This, however, would require satellite access in order to be effective. This could also paired with the last section in order to be have a higher quality of life for the researcher - optionally viewing the data in real time if connectivity allows.

While satellite internet is prevalent, there are few devices that fit the goals of this project. They are mostly designed for hotspotting multiple devices, and often times required a paid subscription plan as well as a high up front cost [4].

3.3 Real Time

This option would be the most ideal of the three if configured correctly. This would allow for real-time data collection and processing at the researcher's location, and would offer the highest quality of life of the three. There are several

products that could allow for this, including the LoRa project that uses an Arduino for long range data transmission [5]. This allows for data transmission of up to 20 miles. Additionally, the project wouldn't require large bandwidth because it will only be transmitting strings every couple of seconds - something any one of these solutions would be able to handle.

With the existence of hardware that makes long range transmission possible, this is the most practical solution to the section. It will allow data to be transmitted as it is acquired by the drone and therefore allow for translation and visualization to occur as soon as possible.

4 FIXED WING DRONE

There are three approaches to the drone's body that will be discussed by the group outside the scope of this paper: multi-rotor helicopter, fixed wing, and VTOL. This section will focus on a single element: the fixed wing drone. A fixed wing drone needs to have the distance to cover large distances (perhaps dozens of miles at a time) while carrying payload of receiver and antennae. This section will discuss the fixed wing drone, as well as the benefits that each of the various technologies will provide. Each subsection will range from least customized to entirely custom built.

4.1 Prebuilt

This is the easiest to implement section of the three that will be talked about. Using a prebuilt drone will offset the workload to other aspects of the project, and seeing that none of the members are aerospace engineers building something completely from scratch is unrealistic. The drone that is picked will need to have a weight capacity to support the custom receiver and antenna that the ECE team will be building and supplying us.

An example of a fixed wing drone that could be used is the Penguin B. With a weight specification of 10kg and a 20+ hour flight time, this would be a very nice solution to the project [6]. This solution, However, would not allow for easy addition of the receiver antenna setup, and would force us to make adjustments probably to the main hull. Many of the prebuilt solutions are very expensive, perhaps prohibitively so, and would force the team have little leeway when purchasing other components of the drone. Making this the best option depends on the true budget and the diligence of the team to figure out where the money will be allocated. If there is excess funding, then it could make sense to choose a prebuilt option as our base for its reliability and the fact that it involves no additional manpower to create.

4.2 Customized Prebuilt

This section allows for the modification of a prebuilt drone discussed in the previous section. While buying a prebuilt drone would inevitably involve modifying it to include our hardware, this option talks more about building on top of an already purchased solution to make it more powerful or flexible. The drone must have a relatively large wingspan so that it will be able to generate lift, and ideally would have multiple motors so that each motor modification would have more of an impact.

With customization, this would allow the team to outfit the drone with better motors, propellers, battery, and perhaps wings in order to get better performance and therefore lift and weight capacity. The TMotor UAV Brushless Motor would be a great upgrade option that allows for much higher propeller rotation than many prebuilt drones, including continuous power of up to 430W and specifically designed for multi-rotor aircrafts [7].

4.3 Custom Built

The custom built drone is where the team gets the most flexibility. This would involve customizing every bit of the drone, from hull body to motors and propeller(s). This will allow for the cheapest drone that is most tailor-made for our project specifications in every way; however, this will also require the most intensive implementation.

The biggest part in building a custom drone is picking a drone body that suits the project's needs. A good option for the drone body is the RMRC Anaconda - a large cargo plane that will have plenty of room for carrying all of the electronics our project will require [8]. Another good option is the Skywalker Black X8, which offers a smaller form factor but is significantly less expensive [9].

Any hull option will provide a much more customizable and inexpensive route to building the drone for the project. This is of utmost importance if the team hopes to stay within the budget allocated for the project. This will also allow us to research and choose the best flight controller, battery and transmitter technologies, making a custom built drone the most viable solution of the three introduced. The flight controller and battery are discussed in a separate tech review, but the team has strong interest in PixHawk and ArduPilot for the autonomous control and programming of the plane. The transmitter was talked about in a previous section of this document.

5 CONCLUSION

This document covered three aspects of the project that will be needed to complete the minimum requirements of the project. These were: Platform, data transmission, and fixed wing drones. Each subsection talked about the benefits and drawbacks with respect to the project, and there was generally a trend within each category - there is the easiest to acquire option which involves buying prebuilt solutions, but allows for little modification. This means there are no options that fits our specific use case. Then there is the custom built solution that involves more hands on work but will allow the group to deliver a product for less cost.

In general, a less expensive hands-on approach will lead to a better finished product as we have a lot of pieces that must interface with each other. It is not realistic to be able to buy prebuilt solutions for every piece and combine them together and expect them to work flawlessly. It makes sense to build our own drone and use the antenna discussed to manually configure and send data back from the drone. With regards to platform, a desktop solution would make the most sense for many use cases in this project. Overall, there are a number of technologies for each piece of this project so the group should have no trouble figuring out *how* to accomplish a certain piece of the project. The trouble comes when configuring each piece to work with one another.

REFERENCES

[1] "Electron." [Online]. Available: https://electronjs.org/

[2] "Connect mission planner to autopilot," accessed: 2018-11-2. [Online]. Available: http://ardupilot.org/plane/docs/common-connect-mission-planner-autopilot.html#setting-up-the-connection

[3] "2x rfd 900mhz ultra long range radio telemetry modem with ftdi & antenna apm pix," accessed: 2018-11-2. [Online]. Available: https://www.amazon.com/900MHz-Ultra-Range-Telemetry-Antenna/dp/B017997YKS

[4] "Bgan - portable broadband & phone," accessed: 2018-11-2. [Online]. Available: http://www.groundcontrol.com/BGAN.htm

[5] "end data at extreme long range using lora with arduino, raspberry pi and intel galileo / cooking hacks blog," accessed: 2018-11-2. [Online]. Available: https://www.cooking-hacks.com/blog/send-data-at-extreme-long-range-using-lora-with-arduino-raspberry-pi-and-intel-galileo/

[6] "Uav factory unmanned platforms and subsystems," accessed: 2018-11-2. [Online]. Available: http://www.uavfactory.com/product/46

[7] "Tmotor uav brushless motor ms2814 770kv," accessed: 2018-11-2. [Online]. Available: https://www.robotshop.com/en/tmotor-uav-brushless-motor-ms2814-770kv.html

[8] "Rmrc anaconda - pnp," accessed: 2018-11-2. [Online]. Available: https://www.readymaderc.com/products/details/rmrc-anaconda-pnp-with-installed-motor-esc-servos

[9] "Latest version skywalker black x8 flying wing," accessed: 2018-11-2. [Online]. Available: https://www.fpvmodel.com/latest-version-skywalker-black-x8-flying-wing_g632.html