Autonomous RFID Drone For Inventory Tracking

A Senior Project Report
presented to
the Faculty of California Polytechnic State University
San Luis Obispo

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Science in Computer Engineering

By

Chase Earhart
Bjorn Nelson
Caleb Rabbon

June 2020

Abstract

With the ongoing rise of e-commerce, warehouses are growing larger in size. Increasing demand for goods means warehouse managers need to know precise details about their inventory, including where it is, how much is in their warehouse, and when items need to be shipped out. A traditional solution to manage inventory is counting and recording it manually with humans and barcode readers, but this leads to human errors such as double counting or missing items. Our solution is an autonomous RFID drone which can quickly and efficiently scan all inventory in a warehouse and accurately display locations of items on a three-dimensional map. An autonomous RFID drone will be able to scan tall shelves and follow designated paths to avoid errors in counting inventory. Our solution utilized the Cal Poly drone, the Cal Poly MiniStock RFID reader, Pozyx positioning hardware and software, and the Python programming language for data acquisition and visualization. Our results allowed us to autonomously fly the Cal Poly drone via GPS, scan and read tags with Cal Poly's MiniStock, and integrate the Pozyx system with Python to visualize scanned data points. Unfortunately, we were not able to test fly the drone inside a warehouse due to COVID-19. We were also not able to fully integrate all components of the project (drone, Pozyx, data visualization) into a single product due to closure of the RFID lab. It is hoped that our work can be further built on by university students in the future and will help spur adoption of autonomous RFID drones in inventory management.

Table of Contents

Introduction

Our team has constructed and programmed an autonomous inventory tracking drone at the request of the RFID Journal LIVE conference. This drone, which is designed to be demoed live on the conference floor, is also expected to be used by clients such as cattle ranchers and small-to-medium sized warehouse owners. Our project utilizes a combination of indoor positioning technology, outdoor GPS communication, RFID scanning equipment, a tailored intuitive user interface, and a custom-built drone to autonomously collect positional data of inventory items in a field or warehouse setting.

I. Client

Our team is working under the request of the RFID Journal LIVE president, Mark Roberti. Our project was initially requested to be shown live on the conference floor for the 2020 RFID Journal LIVE conference in late May. Due to the COVID-19 pandemic, the conference was pushed back to September. We are still anticipating our drone to be flown and take inventory on the conference floor once the conference begins.

II. Stakeholders

There are a number of stakeholders for this project. The most prominent stakeholders are the project team members: Chase Earhart, Bjorn Nelson, and Caleb Rabbon; we all wish to make a robust and successful project, and therefore have a vital interest in its success.

Other stakeholders affiliated with Cal Poly include our IME faculty advisor, Tali Freed, and our graduate student advisor, Maxime Jeanneau. These two individuals both have a vested interest in our project, since its success would bring recognition to both our university as a whole and PolyGAIT, Cal Poly's Center for Global Automatic Identification Technologies.

President of RFID Journal LIVE, Mark Roberti, is also a vested stakeholder in our project. Mr. Roberti personally asked Dr. Freed for such an autonomous inventory drone to be shown live on his conference floor. He certainly wishes for our project to meet expectations, for both the success of the conference and future development of RFID technologies.

The final stakeholder is the owners of the convention center where RFID Live 2020 will take place. Although this stakeholder does not play an active role in this project, the demonstration of the project must adhere to the policies of the conference event space. The convention center has strict rules about indoor drones, such as a maximum weight limit and minimum distance required between the drone and people. Thus, we made sure to plan a demo that would abide by all of these requirements.

III. Framed Insights and Opportunities

Our initial email communication with Mr. Roberti was fairly straightforward; he requested that we build and demonstrate an autonomous indoor drone which utilized RFID technology to take inventory of a warehouse. Mr. Roberti was fairly hands-off for the rest of our project, and he was extremely accommodating in providing us with a large enough project space on the show floor and whatever materials we would need at the convention center (tables, monitors, speakers, etc.).

The main difficulty came when Mr. Roberti sent us a list of requirements that the convention center has for indoor drones. The requirement that all indoor drones weigh no more than 7.5 lb was the most difficult to overcome. The first prototype for our drone, without including the RFID reader or battery, was already over 8 lb. It was strenuous redesigning the drone and finding lighter components to get under the weight limit, but we managed to do so successfully.

IV. Project Goals and Objectives

Goals:

- A drone that can fly safely in indoor and outdoor environments
- Autonomous flight capability for the drone
- A robust offline database for inventory management
- An accessible and intuitive user interface
- The ability to plot tags on a 3D map

Objectives:

- Test fly the drone a sufficient amount to verify that it will fly correctly
- Verify that the drone can fly autonomously around an indoor warehouse setting
- Integrate project software together so all code can access whatever is needed
- Have individuals who are less “tech savvy” test the user interface to ensure it would be intuitive to the average warehouse worker
- Plot tags on our mapping software from a variety of heights and locations while testing for edge cases

V. Project Deliverables

The final deliverable for this project is a Cal Poly designed drone which can fly safely both indoors and outdoors. This drone comes equipped with hardware and software which allows the drone to autonomously fly both outdoors and indoors. In addition, the drone is outfitted with an RFID reader which allows the drone to scan RFID tags either on boxes in an indoor warehouse setting or on cattle in an outdoor field setting. To utilize the data from these tags, we have two primary pieces of software: software that creates a visual 3D map of tag locations and software that produces an Excel spreadsheet documenting all known data about each tag. All technology is designed to be automated wherever possible, and the user experience is streamlined wherever user input is required.

VI. Project Outcomes

Although RFID Journal LIVE 2020 is still months away, and although we are not sure if we will be able to present our project in person, we still expect our drone and our software to be shown live on the conference floor. We hope we will be able to attract the interest of conference attendees who are looking to modernize their company through the use of RFID technology. Ideally, we would like Cal Poly's PolyGAIT to be approached with a request to build a similar drone for a company's private use. If this does not happen, we would still be ecstatic if we are able to get conference attendees to see the potential implementation of RFID technology, thus fostering creativity and ingenuity for new ways of applying RFID technology.

Background

Radio-frequency identification (RFID) uses radio transmissions to enable communication between a reader and a tag, typically in order to identify the object attached to the tag. When a tag is in proximity to the reader's antenna, it will send identifying information like its tag ID to the reader. Passive RFID tags do not require a battery to operate, since the power comes from the radio waves sent by the reader's antenna. Active RFID tags have their own radio transmitter and are powered by a battery. Active tags have a much larger read range but are significantly more expensive than passive tags. ("What Is RFID?", n.d.)

Drones have a variety of potential applications in warehouse operations. A white paper by the Swiss Federal Institute of Technology found that the primary use case for a warehouse drone is inventory management. The growth of global e-commerce and improvements in sensor and navigation technologies will lead the market for warehouse drones to grow by $29 billion over the next 7 years. Conducting inventory management with a drone improves accuracy and decreases labor costs. Annual stock taking (physical verification of item quantities stored in a warehouse) can be done on a more frequent and accurate basis if a drone is used. This avoids the disadvantages of using humans to count inventory by scanning barcodes, which is a slow and expensive process that is error-prone due to its repetitiveness. Warehouse drones can also be used for performing intra-logistics and surveillance. Intra-logistics entails using a drone to transport parts from one place to another. As a surveillance method, drones can inspect pallets or surfaces of a warehouse, which can be dangerous and difficult tasks for humans to do. (Wawrla et al., 2019)

Indoor navigation accuracy is considered to be the biggest challenge of achieving fully autonomous operations of drones in warehouses. Various technologies are being developed to improve the precision of indoor location tracking. The visual SLAM algorithm achieves precision of only 5 cm, while Ultrawide-Band radio technology typically has an accuracy of 10-30 cm. In

order for inventory drones to be beneficial indoors, the warehouse should be large in size (at least 10,000 m^2) and have high shelves (at least 5 m) and long corridors (at least 50 m). These characteristics make a drone more suitable than a human for tracking inventory.
(Wawrla et al., 2019)

A research project at MIT aimed to solve the problem of lost inventory in warehouses using a small aerial drone to identify tag locations within 19 cm of accuracy. Since small autonomous drones were incapable of carrying an RFID reader with sufficient range, signal relaying and processing was used to calculate tag locations. Since an RFID tag will transmit multiple signals to an antenna as the drone moves, the phase difference in the electromagnetic waves can be used to calculate the angle of transmission. To obtain an accurate tag location, several signal processing techniques were used to adjust the phase shift to account for simultaneous transmission between the relay/tag and the relay/reader. Some of the potential use cases of this drone include continuous monitoring, prevention of mismatches, and ability to locate individual items quickly to fulfill customer requests. (Hardesty, 2017)

Since our project team did not have a significant background in signal processing, we chose to use a larger drone that would carry its own RFID tag reader. This would make the drone more versatile for usage in both indoor and outdoor environments. The MiniStock RFID reader designed within PolyGAIT was chosen for tag reading, since it had been successfully used on a drone in the past. To perform indoor positioning, we focused on finding a solution that used ultra-wideband technology due to its high accuracy indoors compared to Wi-Fi, Bluetooth, or GPS. The Pozyx system was selected, since it had good online documentation and it could be interfaced with Python and popular microcontrollers such as an Arduino or Raspberry Pi.

Formal Project Definition

I. Customer Requirements

- Stable and flyable drone
    - Autonomous flight capability
        - Outdoors using GPS for navigation
        - Indoors using Pozyx for navigation
    - RFID tag reading
        - Horizontal reading capability for warehouse shelves
        - Vertical reading capability for cattle tracking
    - Must weigh below the conference's maximum weight limit
- Intuitive UI/UX
    - Automated wherever possible, and simple UI whenever human input is required
- Able to be replicated by potential RFID Journal LIVE clients, if interest is sufficient

II. Engineering Requirements

Spec. Number	Parameter Description	Requirement or Target with Units	Toleranc e	Risk	Compliance
1	Drone Weight	7.5 lb	Max	High	Inspection
2	Tag Read Range	10 ft	Max	Medium	Analysis, Test
3	Pozyx Coordinate Position Accuracy	30 cm	Min	Medium	Test
4	Drone Battery Operating Voltage	22.5 V	Min	Medium	Inspection

Table 1: Engineering Requirements

III. Customer Persona and User Stories

One probable customer persona is the owner of a small to medium sized warehouse. This individual has come to RFID Journal LIVE to see how other companies are utilizing RFID technology to make their businesses more efficient. After walking the conference floor for a while, he comes across a project demonstrating a drone effortlessly taking inventory of warehouse shelves and storing relevant data in an easily digestible Excel file. He sees not only the drone in flight, but examples of how the data is shown to the user in an easily accessible way. Floored by how this technology can effortlessly automate his monthly inventory, he inquires with the conference president about how to get such a drone for his warehouse.

Whenever a shipment enters or leaves the warehouse, a warehouse worker will have to record details about the shipment. With our new inventory software system, this individual fills out details about an incoming shipment into our accompanying inventory software: a short description of the item, the item's weight, and the date that the item is scheduled to leave the warehouse. After putting an RFID tag on the shipment, they place the shipment wherever is most convenient in storage. When a shipment is flagged to leave, they will be informed where it is in the warehouse so they can retrieve it, ship it, and remove its information from their system.

Each month, the manager of a warehouse is required to take inventory. A process that once could take an entire day can now be done with the click of a button. Shortly before closing up for the night, the manager will begin the drone's autonomous inventory scanning feature. In the morning, the manager can look at different sets of data about the warehouse's inventory. They can look at a map of the drone's flight path and the general location of all scanned inventory. They can also pull up an Excel spreadsheet that includes up-to-date data about all inventory in the warehouse. They can manually search this spreadsheet using the inventory's RFID tag ID, description, weight, the date it entered the warehouse, or the time it will leave the warehouse. The manager can then see whenever a piece of inventory is scheduled to be shipped out, allowing them to forward the location of the shipment to one of their workers for it to be sent out.

Design

The information below describes the hardware and software technologies and tools that were used to create a data visualization program to display the tag locations.

I. Drone

Introduction

Two drones used for test flights in this project are shown in Figures 1-2: the DJI S900 and the CPPG (a drone designed by PolyGAIT). Having access to two similar drones gives us a backup system for the conference and during development. This proved to be helpful, since one of the drones crashed due to problems with acquiring a GPS signal while in flight.

Figure 1: DJI S900 Drone with Attached MiniStock Reader and Antenna

Figure 2: CPPG Drone

Redesign to Get Drone Under Weight Limit

In order to comply with indoor flight requirements, the CPPG drone had to be redesigned to weigh a maximum of 7.5 lb. Table 2 shows how weight reductions were made in the overall weight by redesigning various components. The landing gear was lowered, wiring was redone, four out of the eight propellers and motors were removed, smaller batteries were used for the drone and MiniStock, and the MiniStock weight could be reduced through using a lighter housing.

Part	Initial Design			Final Design			Weight Delta (lb)
	Qty	Weight (lb)	Subtotal (lb)	Qty	Weight (lb)	Subtotal (lb)	
Landing Gear	2	0.381	0.763	2	0.280	0.560	-0.203
Actuator	2	0.148	0.295	2	0.046	0.093	-0.203
Skid Adapter	2	0.033	0.066	2	0.033	0.066	0.000
Rubber Foot	2	0.044	0.088	2	0.044	0.088	0.000
25x23x349mm 3k CF tube	2	0.086	0.172	2	0.086	0.172	0.000
16x14x445mm 3k CF tube	2	0.068	0.137	2	0.068	0.137	0.000
Rivets	2	0.002	0.004	2	0.002	0.004	0.000
Battery (22.2 V, 16000 mAh)	1	4.302	4.302	1	1.499	1.499	-2.803
Motor w/ Prop	8	0.254	2.029	4	0.254	1.014	-1.014
Motor Mount (dual to single)	8	0.304	2.434	4	0.137	0.547	-1.887
ESC	8	0.084	0.670	4	0.084	0.335	-0.335
Power Distribution Block	1	0.461	0.461	1	0.108	0.108	-0.353
Frame w/ Flight Hardware (no legs)	1	2.364	2.634	1	2.364	2.364	0.000
Antenna w/ Cable	1	1.367	1.367	1	0.628	0.628	-0.739
MiniStock	1	1.940	1.940	1	0.529	0.529	-1.411
MiniStock Battery	1	0.816	0.816	1	0.551	0.551	-0.265
Total		**11.457**	**16.330**		**5.883**	**7.585**	**-8.745**

Table 2: Drone Weights Before and After Redesign

Autonomous Flight Testing

Autonomous test flights were done after the modifications were made to the drone weight. By plotting a series of waypoints in the MissionPlanner software as shown below in Figure 3, the drone was able to successfully navigate autonomously (no controller input) while outdoors. Future testing will require testing the drone's autonomous navigation capabilities while indoors using the Pozyx positioning system.

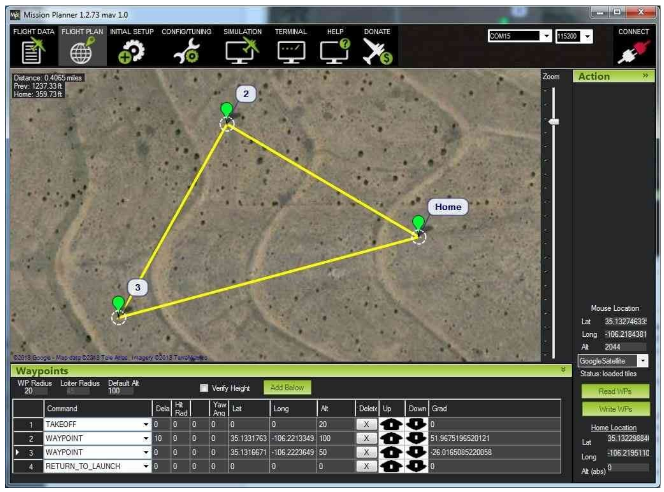

Figure 3: MissionPlanner Waypoint Navigation Setup

II. Pozyx

Introduction

Pozyx is a positioning system that enables precise indoor mapping through ultra-wideband technology. This platform uses fixed anchors placed in known positions and tags placed on objects that move around within a space. Positional data on the x, y, and z axes is computed by taking time of flight measurements between anchors and tags. The Pozyx system shown in Figure 4 was chosen due to the constraints and challenges of tracking location in an indoor space. GPS has an accuracy within 5 meters in open space, but it does not work well to be accurate enough indoors. In contrast, the Pozyx system provides up to 10 cm of accuracy in line of sight conditions and 30 cm of accuracy in environments. (Pozyx, n.d.)

Figure 4: Pozyx Creator Kit

Basic Setup

To obtain data, the master tag and 4 anchors were all connected to a power source. The anchors were placed in 4 different corners of a room, and auto-calibration on the Pozyx software was run as shown in Figure 5. After measuring the actual x, y, and z distances from the anchors to a single point in the room, the calibrated values were adjusted to improve accuracy.

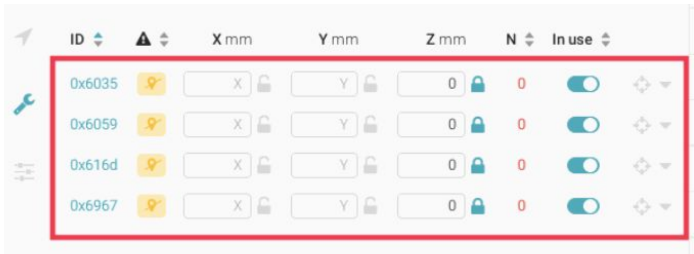

Figure 5: Screenshot of Anchor Calibration Step

When connected to a computer, the master tag's position could be seen in the Pozyx software as shown in Figure 6. Data updates occur at a rate of 60 Hz for usage of a single tag. This fast data acquisition rate is necessary for real-time autonomous navigation of an inventory drone.

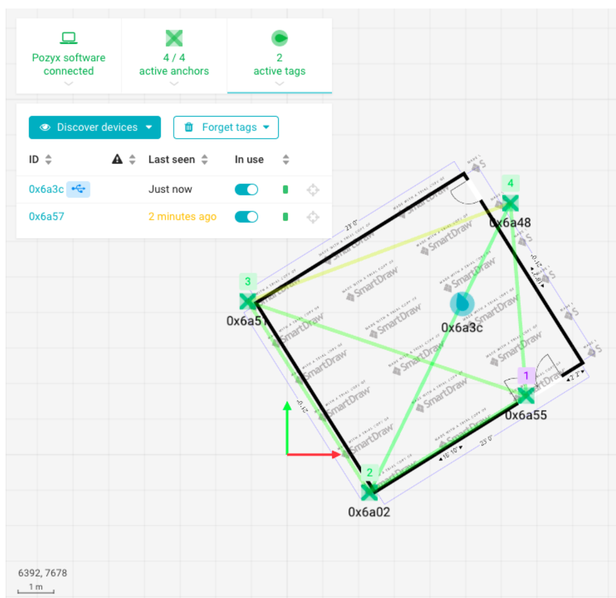

Figure 6: Screenshot of Pozyx Software Showing Tag Position

A Python script was used to save data into a file, for later usage in data analysis and visualization of the locations of RFID tags. Every time data is captured for a given tag, the values are appended to a JSON file. In addition to recording the timestamp and 3-D XYZ coordinates, other potentially useful data values saved include acceleration and yaw/pitch/roll. The Python data acquisition code and an example of output to the JSON file are shown below.

<pre>import ssl
 	
host = "localhost"
port = 1883
topic = "tags"
 	
def on_connect(client, userdata, flags, rc):
    #print(mqtt.connack_string(rc))
    mqtt.connack_string(rc)
 	
# callback triggered by a new Pozyx data 	
packet
def on_message(client, userdata, msg):
    #print("Positioning update:",
msg.payload.decode())
    msg.payload.decode()
 	
def on_subscribe(client, userdata, mid,
granted_qos):
# print("Subscribed to topic!") 	
 	
client = mqtt.Client()
 	
# set callbacks 	
client.on_connect = on_connect
client.on_message = on_message
client.on_subscribe = on_subscribe
client.connect(host, port=port)
client.subscribe(topic)
 	
# works blocking, other, non-blocking, 	
clients are available too.
client.loop_forever()</pre>
	<pre>[{
	  {
	  "version": "1",
	  "tagId": "24576",
	  "success": true,
	  "timestamp": 1524496105.895,
	  "data": {
	    "tagData": {
	      "gyro": {"x": 0, "y": 0, "z": 0},
	      "magnetic": {"x": 0,"y": 0,"z": 0},
	      "quaternion": {"x": 0, "y": 0,
	                     "z": 0, "w": 0},
	      "linearAcceleration": {"x": 0, "y": 0,
	                             "z": 0},
	      "pressure": 0,
	      "maxLinearAcceleration": 0
	    },
	    "anchorData": [],
	    "coordinates": {
	      "x": 1000,
	      "y": 1000,
	      "z": 0
	    },
	    "acceleration": {
	      "x": 0,
	      "y": 0,
	      "z": 0
	    },
	    "orientation": {
	      "yaw": 0,
	      "roll": 0,
	      "pitch": 0
	    },
	    "metrics": {
	      "latency": 2.1,
	      "rates": {
	        "update": 52.89,
	        "success": 52.89
	      }
	    }
	  }
	}
	}]</pre>

Pulling Data from the Hardware

Data can be accessed from the hardware by simply reading the JSON file, which can be done using libraries of various programming languages. Python was used again in this project to determine the positions of RFID tags in a room, by searching for the Pozyx data packet in the JSON data with the closest timestamp to when an RFID tag was read. The associated position data for an RFID tag is then updated in a database. This location information for RFID tags can be represented in a data visualization using Plotly or simply viewed in a spreadsheet by the warehouse manager.

In certain usage scenarios, the measured positions of RFID tags will not be within 10 cm of accuracy. Ideally, a drone would fly overhead, and the x and y coordinates of the drone and piece of inventory would be the same. However, in certain warehouse configurations, the drone would likely fly between shelves and inventory would be on the sides rather than directly below the drone. There are several potential ways of mitigating this issue. Manual adjustments in software can be made to positions, if there is always a fixed distance offset. Another option is to monitor yaw, pitch, and roll data over time, in order to determine when the drone is turning around shelving and thereby calculate the probability of a tag being within a certain area.

III. MiniStock

Introduction

The MiniStock is an RFID tag reader designed within PolyGAIT that connects to an antenna. Inside the housing, there is a Raspberry Pi and other connected peripherals necessary for reading tags. The MiniStock and its drone mounting attachment are shown below in Figure 7.

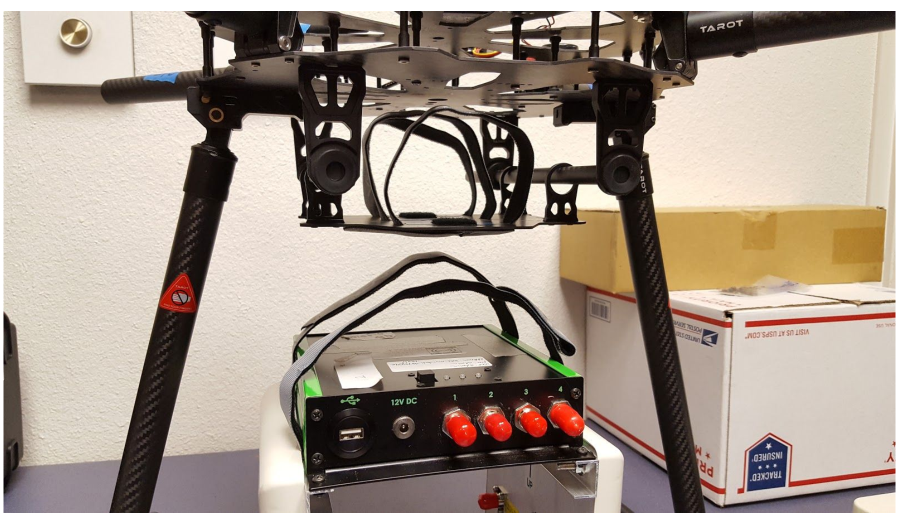

Figure 7: MiniStock Device and Mount

Reading RFID tags

When a tag is read by the MiniStock, the tag ID, antenna ID, date, time, and MiniStock ID are saved in a CSV file on the Raspberry Pi. Some example data is shown below in Figure 8. It is common for tags to be read multiple times, but duplicates can be adjusted for in the software. Since we did not have access to the MiniStock during spring quarter due to the MiniStock, we had to rely on CSV files of sample data rather than reading tags live.

HexTagData	Antenna	Date	Time	MiniStockSerialNumber
300833B2DDD9014035050000	4	2020-06-10	13:51:01	b1951afa
0504300800000000000012A5	4	2020-06-10	13:51:11	b1951afa
030320080000000000000156B	4	2020-06-10	13:51:21	b1951afa
E200322F81B963311307458B	4	2020-06-10	13:51:31	b1951afa
E2003411B802011357281004	4	2020-06-10	13:51:41	b1951afa
050000000000000000003411	4	2020-06-10	13:51:51	b1951afa
E2003411B802011111363469	4	2020-06-10	13:52:01	b1951afa
E2003411B802011111363415	4	2020-06-10	13:52:11	b1951afa
E200322F6A2C7DB1409A41F6	4	2020-06-10	13:52:21	b1951afa
0504300800000000000012A7	4	2020-06-10	13:52:31	b1951afa
000000000000000000003462	4	2020-06-10	13:52:41	b1951afa
4A6F686E0000000000000000	4	2020-06-10	13:52:51	b1951afa
3008048D159E28000000010A	4	2020-06-10	13:53:01	b1951afa
AD82010046A001B05E0000A9	4	2020-06-10	13:53:11	b1951afa
E200322F81CB65B113078D95	4	2020-06-10	13:53:21	b1951afa
E20019C60906AAF1135D1AAA	4	2020-06-10	13:53:31	b1951afa

Figure 8: CSV File Showing Tags Read

IV. Plotly

Introduction

Plotly is a python library that allows users to graph data points on three dimensional graphs. It is used as our visual solution to display the drone and tag locations.

Basic Setup

We used Jupyter Notebook to visually display the Plotly graphs. Jupyter Notebook is an open source web application that can run python code live. This means it can take in data points and graph them inside the Jupyter Notebook's web application. To access the Jupyter Notebook we used Anaconda Navigator, which is a graphical user interface (GUI) package manager. This package manager installs multiple python packages needed to run Plotly and interface between different file types (.json and .csv). This combination of tools allowed us to visually see and interact with the Plotly data points in a web interface.

Figure 9 shows how to open a Jupyter Notebook from Anaconda Navigator by clicking on a virtual environment. Figure 10 shows the file browser in Jupyter Notebook, the .pynb files are the notebooks containing code. By opening a notebook file, the Python code can be edited and tested in a web interface as shown in Figure 11.

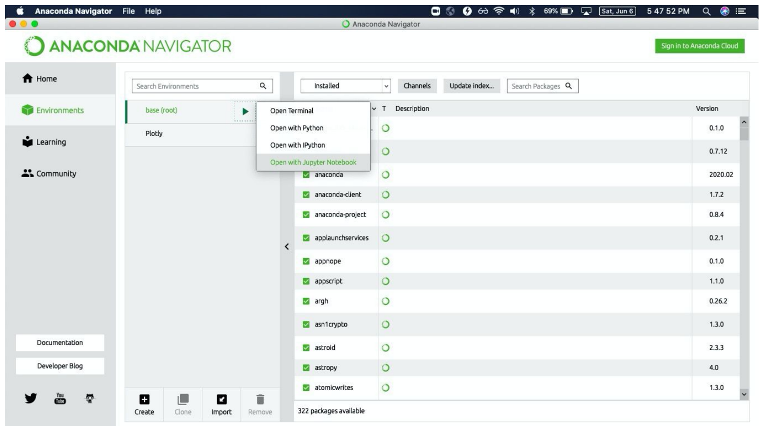

Figure 9: Screenshot of Anaconda Navigator UI

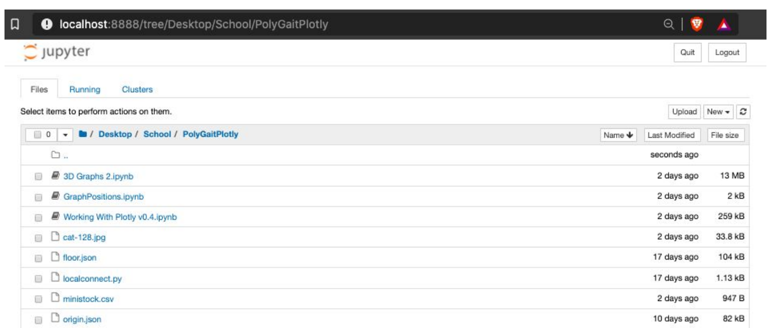

Figure 10: Screenshot of Jupyter Notebook Web Application

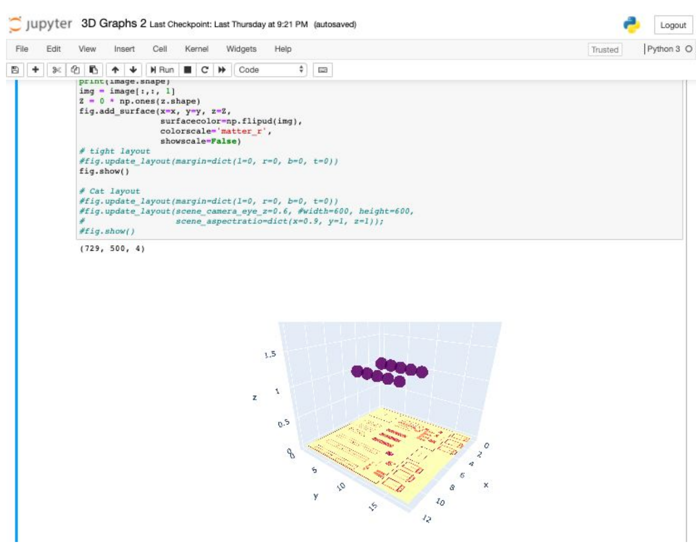

Figure 11: Screenshot of a Jupyter Notebook File

<u>Our Use Case</u>

We used Plotly to visualize the location of the tags read by the drone as well as the drone's position. This was done by using two relational databases. One database held the location, direction, and timestamp of the drone. This data was gathered from the Pozyx positioning system described in the next section. The second database was created from the MiniStock and held the times and tag identifiers of when a tag was read. The one relational element between the two databases is the time. By synchronizing the times, the location of the tag can be predicted. This is done by using the drone's location and direction produced by the Pozyx positioning system. For example, if a tag was read at 3:00 PM, then the tag's location can be determined by searching for the drone's position closest to that time.

The results of the data visualization demo are below. The 2D tag visualization is shown in Figure 12 and the 3D tag visualization is shown in Figure 13.

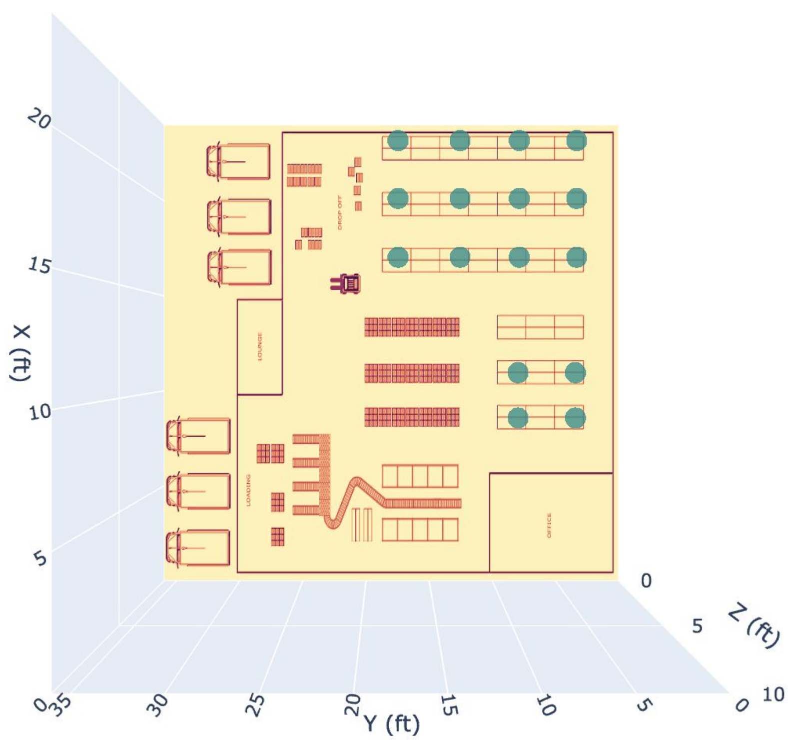

Figure 12: Tag Locations 2-D View

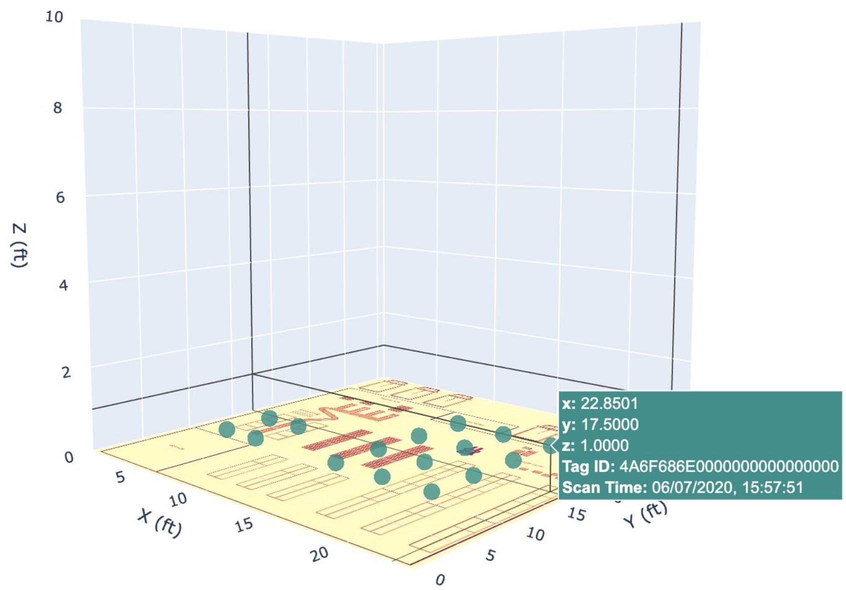

Figure 13: Tag Locations 3-D View

System Testing and Analysis

Various tests of the drone and individual sensors were done during technical development.

I. Drone Testing

Flight and Landing

We flew the drone at the airfield near Cuesta College, to comply with airspace regulations in San Luis Obispo. The takeoff and landing site is shown below in Figure 14.

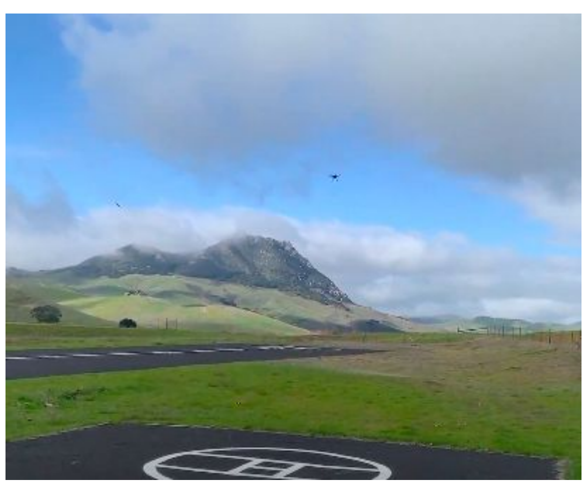

Figure 14: Drone Airfield

Kill Switch and Propellers Breaking Upon Landing

The drone accidentally lost control and some propellers broke upon landing. We then tested backup propellers, but they were cheaply made and broke when they spun up.

Mounting the MiniStock and Antennas to the Drone

The MiniStock and its antennas were mounted to the drone as shown in Figure 15.

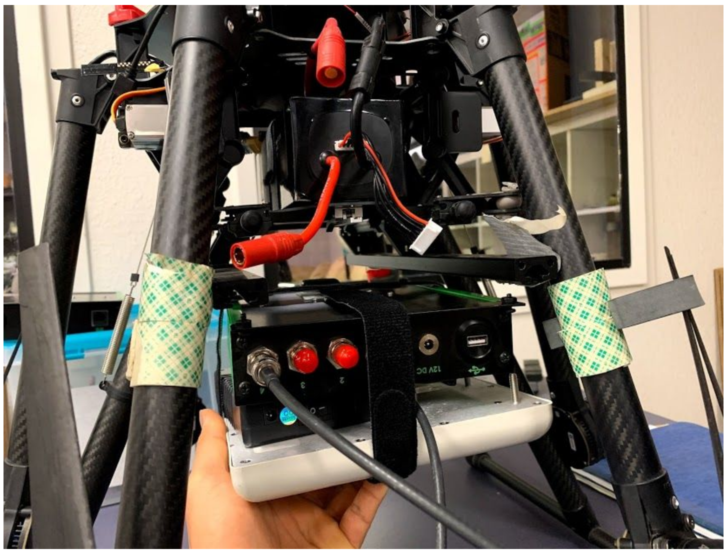

Figure 15: Mounted MiniStock and Antennas

Weight Testing

As shown in Figure 16, we weighed all of the different parts of the drone, MiniStock, antennas, and cables. This was necessary in order to determine the best ways of reducing the weight.

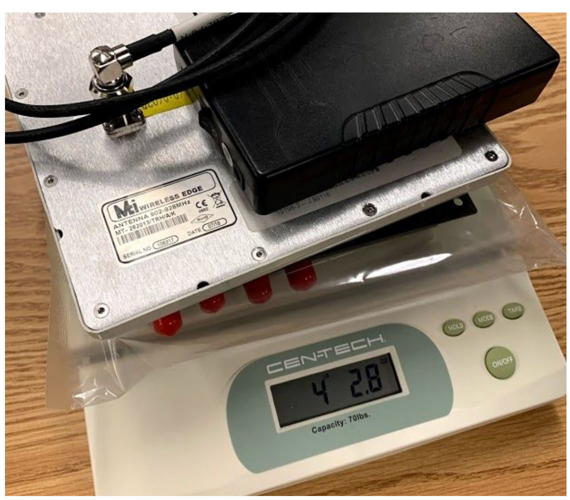

Figure 16: Weight Measurements

II. MiniStock Testing

The maximum distance to read tags needed to be determined. The maximum range was around 12 feet away from the antenna and around 2 feet above and below the antenna. This can be visualized as a cone projected out from the antenna with a radius of 2 feet and height of 12 feet.

III. Total System Testing

Due to the situation of COVID-19, we were not able to integrate all components of the drone, MiniStock, and antenna together since the lab was closed down.

IV. Integration of Pozyx and Plotly

The integration of Pozyx and Plotly was done by manually creating a ministock.csv file and a pozyx.json file. The ministock.csv file contains data of when the MiniStock read RFID tags. The pozyx.json file contains positioning data from the Pozyx positioning system. We then wrote a Python script to create a combined dataset of the ministock.csv and pozyx.json files to find the positions where tags were read.

Conclusion and Future Work

The initial culmination of our work was to present this project at RFID Journal LIVE 2020 conference, which was originally scheduled to take place in late April 2020. Due to COVID-19, our project, along with most of the world, had to adapt to the rapidly changing circumstances. We now have streamlined software for our project, however more testing will be required to verify that the drone itself can run this software efficiently. We have had initial discussions with Dr. Freed about potentially continuing work on this project over the summer, and we would all be eager to pursue that opportunity if it arose. We would also need to verify that the user experience for our project is adequate in an actual warehouse setting. Feedback on whether more information is required and whether the user interface is too convoluted or too simplistic would be helpful. It is our belief that if the project is not yet adequate for a warehouse environment, only small revisions would be required for it to exceed expectations.

Teaming

As a team, we all worked to make the drone stable during flight; despite its simplistic sounding nature, this took many weeks to figure out in its own right. Additionally, we all brainstormed ways to lighten the drone’s weight in order for it to pass the standards set by the conference convention center. Below you will find what each team member did individually or in small groups.

Chase Earhart: I was primarily in charge of user experience for the project; I worked on the Python library which reads/outputs from a .csv file and accommodates user commands to add or edit tags from the terminal. I focused on creating a user experience that both contained all information a warehouse manager would need while also being streamlined and intuitive in its design.

Bjorn Nelson: I primarily worked on integration of the MiniStock and Pozyx systems and collaborated with Caleb on the data visualization in Plotly. This required implementing real-time data processing while considering common usage issues such as tags being read repeatedly and the drone being in a different location than a tag. During the early stages of the project, I led the effort of researching existing inventory drones to incorporate this knowledge into the design and development of our project.

Caleb Rabbon: I was in charge of designing and building a MiniStock and antenna carrying system for the drone. I also worked with Bjorn to integrate the Pozyx software with Plotly. While in quarantine I placed the Pozyx physical tags in my room and tested its accuracy and ability to output live data.

Reflection

Chase Earhart: I have always been fascinated with RFID technology and I find drones very interesting; the ability to combine these two technologies has been very informative and entertaining. There were a number of issues which ended up being much more difficult than anticipated, such as taking weeks to get the drone flying, and there were a few issues which turned out to not be that difficult, such as finding a company which specializes in providing equipment for autonomous indoor flight, a thing I thought would be impossible to code on our own. Due to COVID-19 and the resulting shutdown of Cal Poly, I feel that I was not able to experiment as much with the hardware of the drone as I would have liked. With that said, much of my work was entirely software based and I am quite proud with what I have created. I am confident that this project would be perfect for small to medium sized warehouses to keep better track of their inventory.

Bjorn Nelson: I enjoyed being able to work on integrating so many different hardware and software technologies into a single project. I gained experience with new sensors like passive RFID tags, and I also learned a lot about the usage of existing software libraries like Plotly to enhance the software development process. I gained a more fundamental understanding of how drones are built by learning how to fly one and rebuilding it to account for weight restrictions. I also learned many virtual project management skills and observed how a technical project evolves over time as the team gains a clearer understanding of what the final product will be. I appreciate working with everyone on this team: Chase and Caleb were very committed and positive teammates, and Tali and Maxime provided helpful guidance and support to achieve our goals. I feel like we made a lot of progress despite the challenging circumstances of COVID-19. Combining a drone and RFID technology provides a novel solution to inventory management, and I believe our final product would be a useful tool in medium sized warehouses.

Caleb Rabbon: It was a pleasure to work with all members of this team and to create a software program that can be used for future teams and individuals. I thank Tali for providing us access to the RFID lab before COVID-19 hit and Maxime for guiding us in flying the Cal Poly drone. I thank the rest of the team members for being flexible and understanding. After working on this project, we all learned valuable skills such as virtual project coordination, creative testing, drone flying, integration of python libraries, and finally utilization of third-party software to create a cohesive final product. This product pulls data from the MiniStock and Pozyx positioning system and then visually displays it on a floor plan.

Bibliography

Cal Poly PolyGAIT. (n.d.). What Is RFID? Retrieved June 12, 2020, from https://polygait.calpoly.edu/what-rfid

Hardesty, L. (2017, August 25). Drones Relay RFID Signals for Inventory Control. Retrieved June 12, 2020, from http://news.mit.edu/2017/drones-relay-rfid-signals-inventory-control-0825

Pozyx. (n.d.). Creator Kit. Retrieved June 12, 2020, from https://www.pozyx.io/shop/product/creator-kit-65

Wawrla, L., Maghazei, O., and Netland, T. (2019, August). Applications of Drones in Warehouse Operations. Retrieved June 12, 2020, from https://ethz.ch/content/dam/ethz/special-interest/mtec/pom-dam/documents/Drones%20in%20warehouse%20opeations_POM%20whitepaper%202019_Final.pdf

Appendices

Source Code: https://github.com/bjornhnelson/RFID-Inventory-Drone

Project Video: https://youtu.be/OQxa1KQnF4o