Final Report for ECE 445, Senior Design, Spring 2020

TA: Jonathan Hoff

01 May 2020

Project No. 46

Anushrav Vatsa (avatsa2)

Colin Sandstrom (colinls2

Ioan Draganov (idraga3)

By

VR Motion Controls for Drones

Abstract

Standard RC Joystick drone controllers are unintuitive and hard to use. Team 22 from Spring 2019 [1] sought to solve that problem by developing a glove to remotely maneuver a drone. The VR Motion Controls discussed in this paper consist of a controller in the form of a VR Headset (for use with a phone screen) with an Inertial Measurement Unit (IMU) mounted on top, and a wireless transmitter using the DJI Ocusync 2.0 technology for data transmission. This project is meant to augment high end drones capable of streaming video by replacing the standard joystick controller with a headset that reads the movements of the user's head and processes them into usable commands which are sent to the drone.

Contents

1. Second Project Motivation

1.1 Problem Statement

Drones are becoming increasingly popular among hobbyists, but using a drone comes with a steep learning curve and a lot of regulations. The FAA (Federal Aviation Administration) requires that anyone piloting a drone must obtain a "*Remote Pilot Certificate"* [2], which entails that the user can speak, read, and write in English, is mentally and physically fit enough to operate a drone, and that the user pass an *Aeronautical Knowledge* exam. However, the certificate then lasts for two years before it must be renewed, showing how difficult it is to start flying a drone. Beyond the legal requirements to operate a drone, the user also has to struggle with the issue of using some sort of joystick controller to maneuver the drone.

Using a joystick may not seem terribly difficult at first glance, but joysticks are not very intuitive, especially as the drone flies further from the user and starts changing its orientation. In particular, when the user and drone are facing different directions, if the user wants the drone to move forward from their perspective, they cannot simply tell the drone to move forward, they have to account for the fact that from the drone's perspective it may need to move backwards or to the side instead.

1.2 Solution

In Spring 2019, Group 22 created an intuitive drone controller that utilized a glove with gyroscopes and accelerometers to control the motion of a drone [1]. They sought to eliminate the issue of using a joystick by having the user control the drone using hand motions. This approach was much more intuitive as they sought to change the dynamic of using a tool to control a drone by treating the drone itself as the tool. They made a glove which allows the user to interact directly with the drone by interpreting gestures and translating them into commands for the drone, and they called it an Intuitive and Ergonomic Gesture-based Drone Controller (which will be referred to as the IEG controller). The IEG controller was very good for precise control over a drone at close range, and they demonstrated its functionality by guiding a drone through an obstacle course. They also made their product compatible with a wide variety of cheaper, indoor-focused drone kits for the everyday consumer.

The IEG controller was very successful, but we noticed that there are some limitations on their implementation. We chose a fundamentally different approach to solve the issue of unintuitive drone controllers. Our solution seeks to eliminate the use of the drone as a tool, and instead treats it as an extension of the user's body. By adapting a VR headset, we strive to replace the user's field of vision with that of a high-end drone capable of streaming video in real-time. Similar to the IEG controller, our device interprets the movements of the user, but our device allows the user to control the drone as if it were their own body.

Our VR Motion controller (which will be referred to as the VRM controller) follows the user's line of sight and directs the camera to show what the user would see if their head were in the same position as the drone. It operates in two modes, "Flight" mode and "Periscope" mode. In "Flight" mode, the VRM moves the drone forward and backwards to show the user what is above or below their initial field of view. In "Periscope" mode, the VRM controls the pitch of the camera instead of moving forward or backwards.

Unlike users of the IEG controller, VRM users cannot observe the drone in detail when it is close to them, but the VRM has a maximum operating range of 5 miles [3] versus 500 meters [4] for the IEG, so it allows the user to maneuver the drone to places that would be inaccessible to the IEG. This shows that consumers who are interested in the IEG would have less use for the VRM, and consumers interested in the VRM would also find the IEG less useful.

1.3 High-Level Requirements

- *Interpretation*:

  The device must be able to sense the orientation of a user's head with an error margin of ± 5 degrees when measuring pitch, yaw, and roll.

- *Transmission:*

  The device must be able to transmit orientation and acceleration data to the drone wirelessly in real time (less than 100ms for our purposes).

- *Execution*:

  The device must be able to control the camera angle and rotation of the drone based on data from the headset while movement mode is enabled.

1.4 Visual Aid

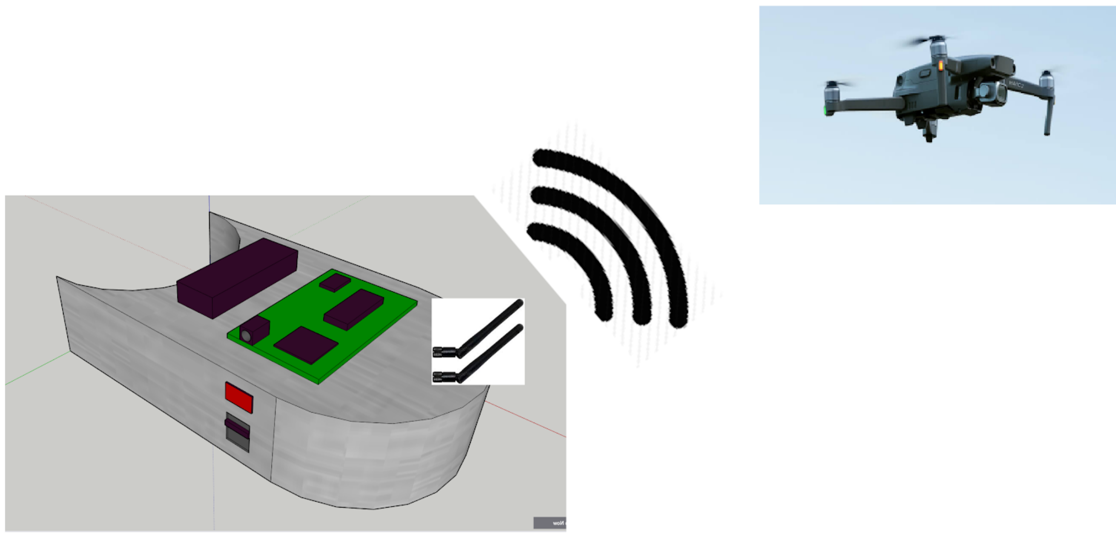


Figure 1. Visual Representation of Project

1.5 Block Diagram

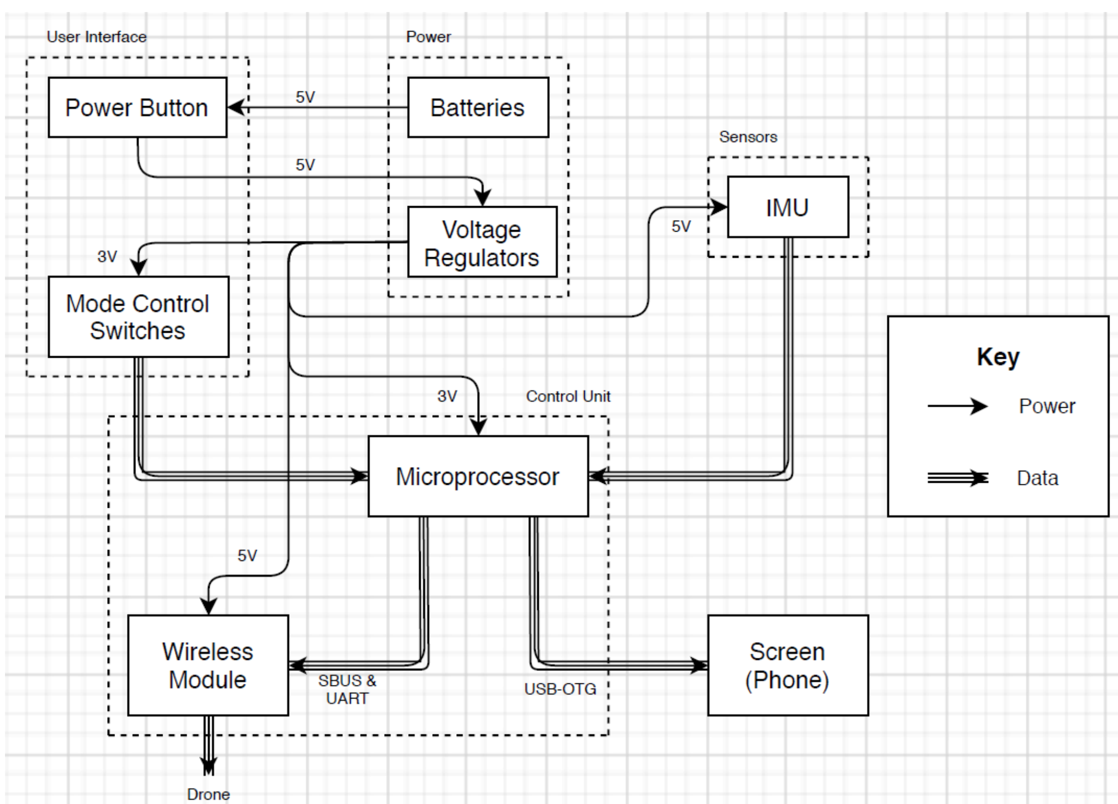


Figure 2. Drone Controller Block Diagram

2. Implementation

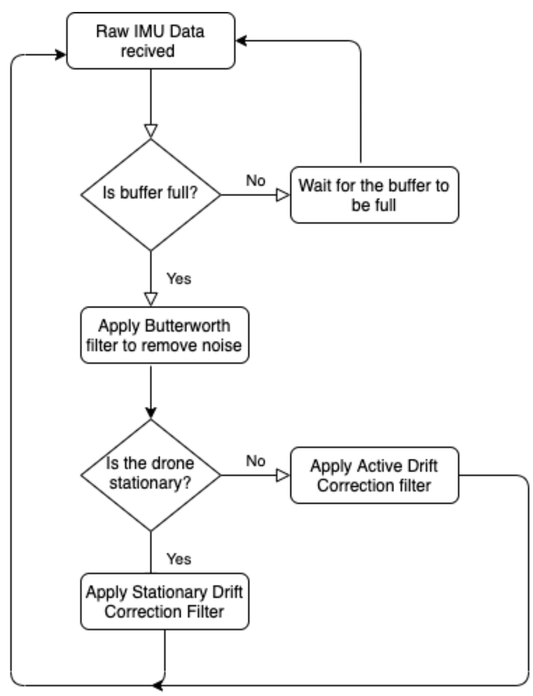
2.1 Implementation Details and Analysis

Interpretation and processing of the IMU data in an accurate manner is critical to the success of this project. We are using the BXM55 IMU from Bosch which is a low power 9 axis IMU with an accelerometer, gyroscope and magnetometer [5]. Low power IMUs are very susceptible to drift caused by integration over their raw input values. Over time, this can significantly skew the output values of the IMU. Our solution to this problem is to de-noise the signal and create an active feedback system that constantly monitors and fixes the data so that it is better aligned with the actual motion of the drone and the user.

Since we do not have access to drones or an IMU unit, we used the data released by the University of Minnesota from their studies on IMU performance on small UAVs [6] to test our algorithm. The data set consists of two different sets of 3 dimensional positional data that we performed our analysis on.

One part of the data set is dedicated to highly accurate elevation (altitude), longitude and latitude collected from a ground station tracking the drone's flight. The other data we have is the combined raw positional values collected from the GPS and IMU unit. After much research we were unable to find any reliable sources of unaltered drone IMU data, so we had to resort to this positional data instead.

The flowchart on the right shows the flow of control for the error correction module. We use a buffer to collect enough samples before we can start processing. Once the buffer is full we remove the noise from the data using a butterworth filter and following that we decide if the drone is stationary or in flight. This is very important because the noise in the IMU data affects the IMU positional data to drift more than it does during flight. After the application of drift correction the data is passed to the flight control unit and the buffer is full again we process the new batch of data.



Flowchart 1: Drift Reduction Flow of control

2.2 Supporting Material

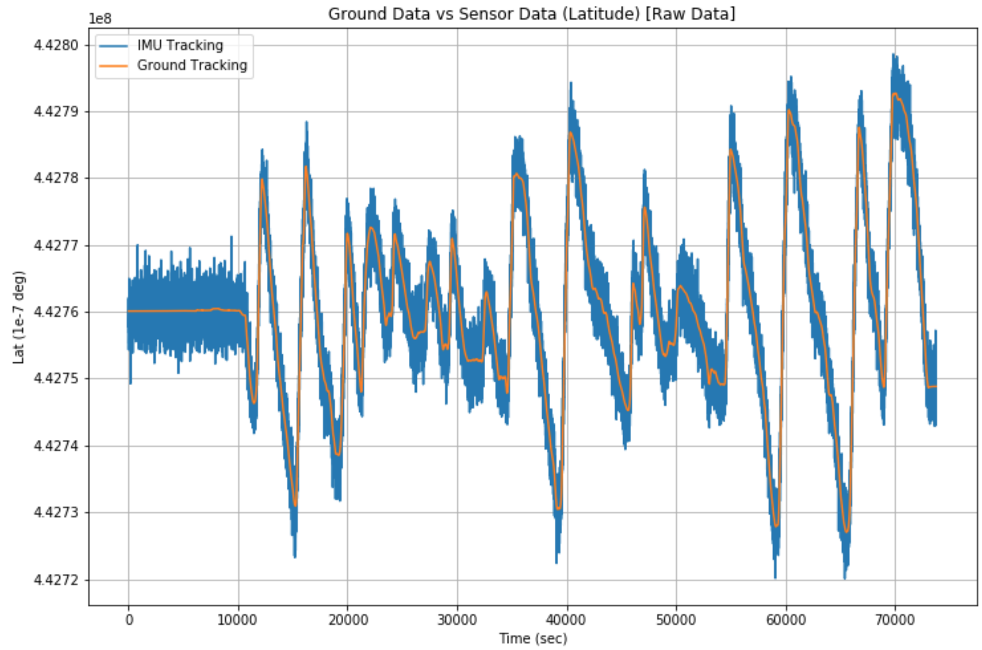


Figure 3.A) Raw Latitudinal Data

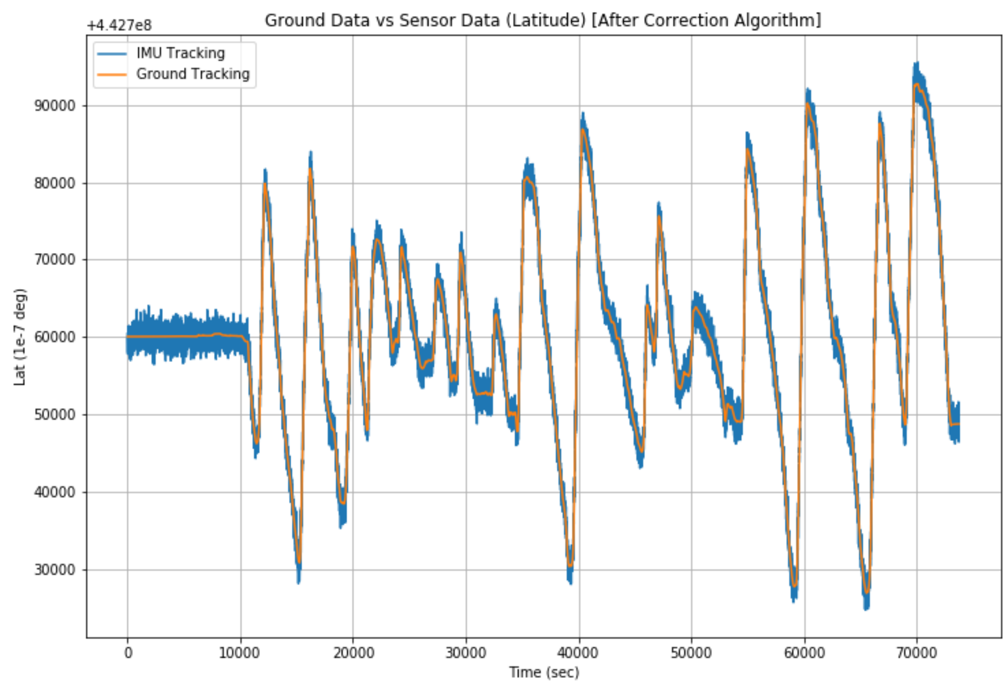


Figure 3.B) Clean Latitudinal Data

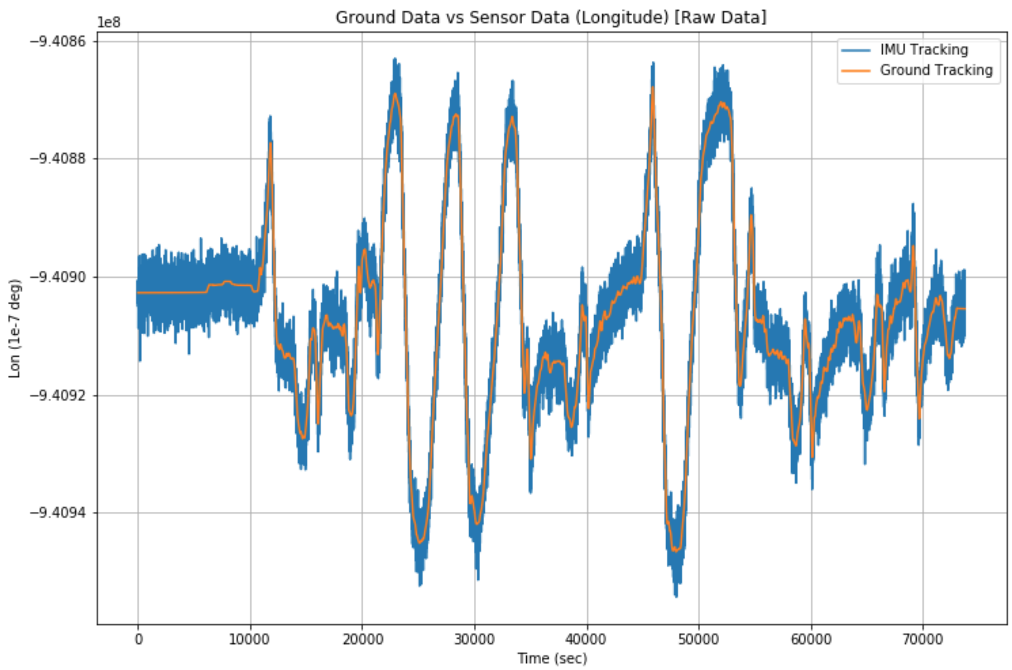


Figure 4.A) Raw Longitudinal Data

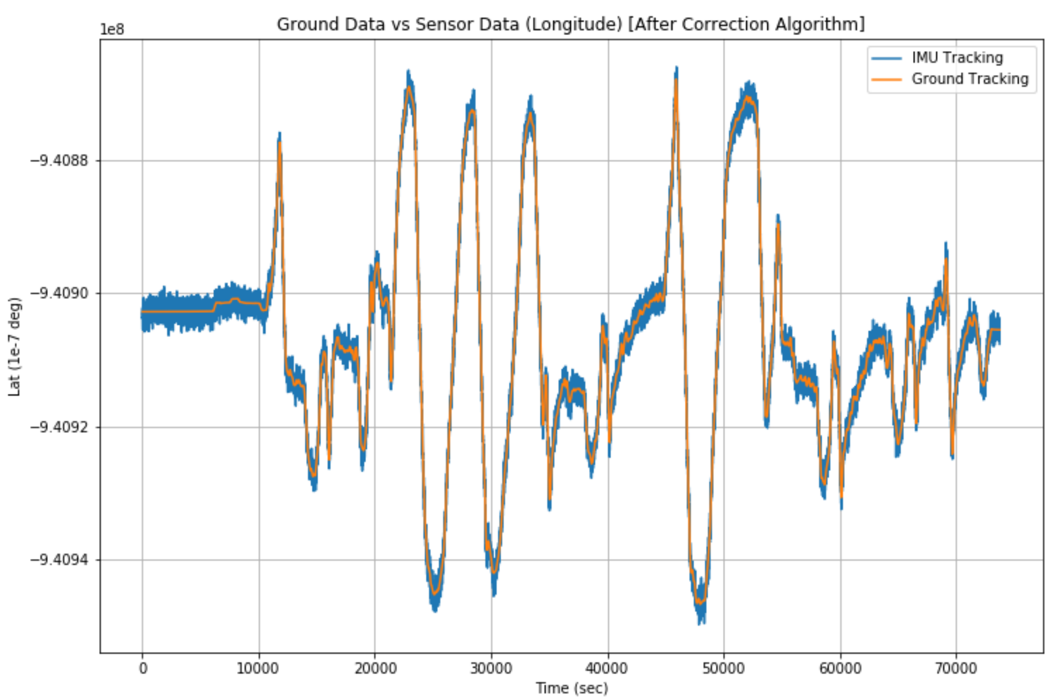


Figure 4.B) Clean Longitudinal Data

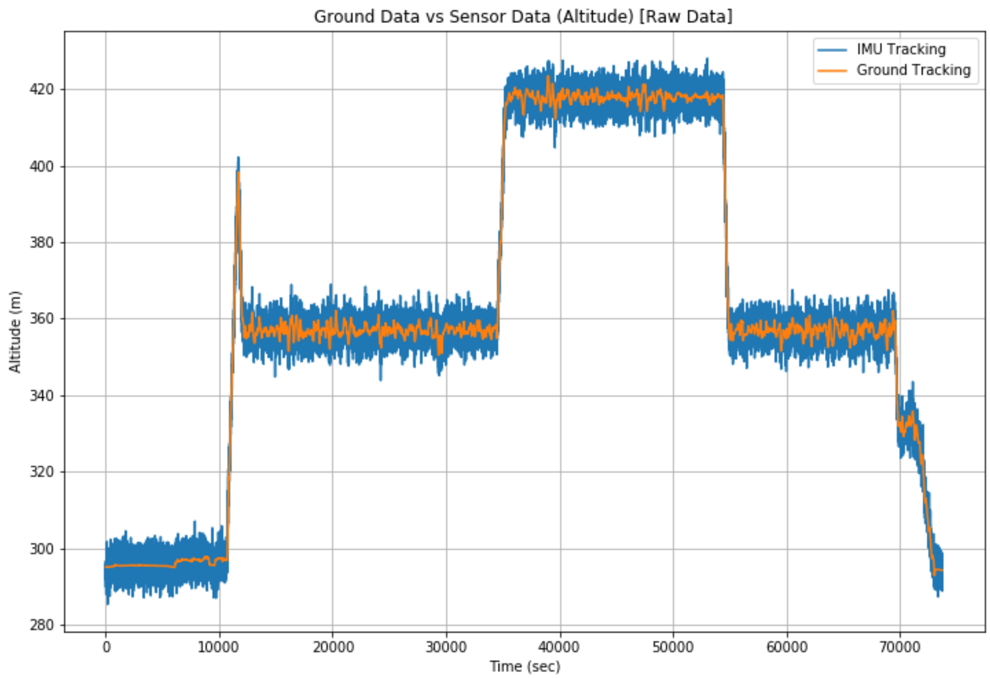


Figure 5.A) Raw Altitudinal Data

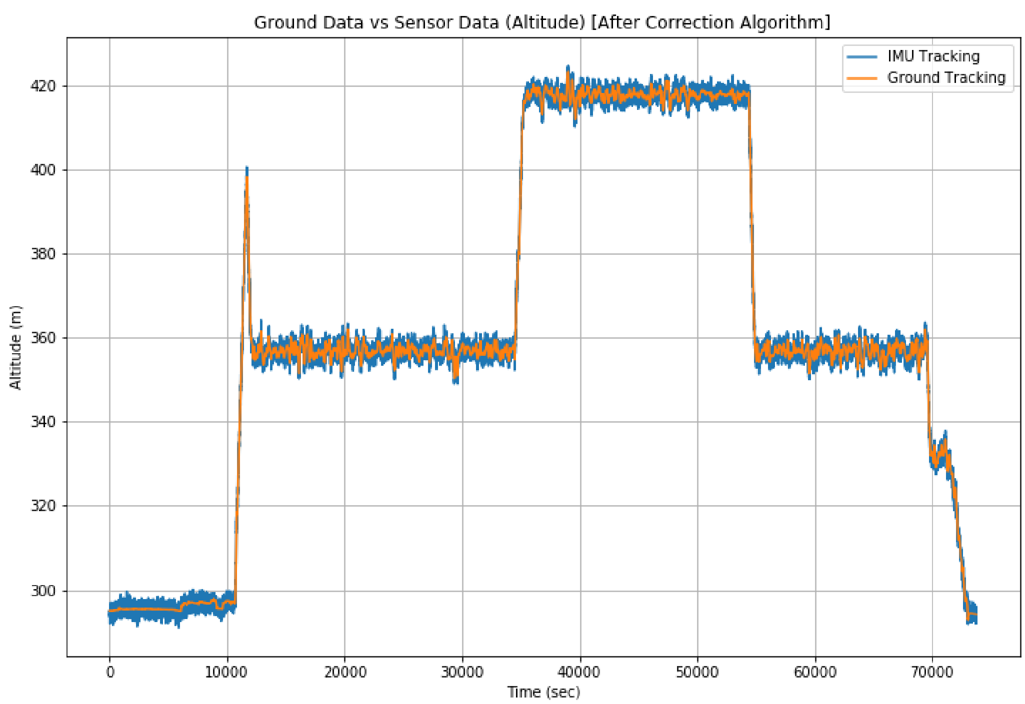


Figure 5.B) Clean Altitudinal Data

2.3 Results and Explanation

As seen in Figures 3-5 shown in the previous section, we use the ground station tracking as a control group of measurements, and the goal of our implementation is to achieve an output as similar to the control data as possible. This is critical since the interpretation and accuracy of the IMU data is integral for our project's success.

Here we implemented a denoising and drift reduction algorithm to make the IMU measuresurements match more closely to the control measurements from that ground station.

We used a *Butterworth* low pass filter to remove the unnecessary noise from the data, this was done experimentally with the latitude values and verified using the altitude and longitude data set to make sure the desired effect was achieved on the measurements on all axes [7].

Lastly, we implemented the stationary drift correction algorithm on the data which uses a *Kalman filter* to help reduce the divergence of the data points [8]. This is important because the raw position values do not factor in the drift added to the actual physical measurements taken by the IMU. This is quite apparent if you take a look at the measurements at the start and the end of the flight where the variance is the highest in the raw data especially when the drone is mostly stationary.

As evidenced in the figures from the previous section, the above implementation is able to generate results with significantly reduced noise and more accurate adherence to the test data provided by the ground station.

3. Conclusions

3.1 Implementation Summary

For the implementation of our second project, we took sample IMU data from a drone and used our denoising algorithm to reduce the amount of noise the IMU introduced into the signal. We were able to see marked improvements in the amount of noise, which would cause the IMU positional data to drift much less when integrating its values. This denoising of IMU data works towards our first high level requirement to sense the orientation of a user's head. Using the denoising algorithm, we are able to take data from the IMU on the VRM controller and reliably use it to generate instructions for the drone's motion.

3.2 Unknowns, Uncertainties, Testing needed

This project relies heavily on the ability of the processor to interpret IMU data and send instructions based on the data to the drone. While we can do simulations of what kind of error margin to expect from the IMU and what kind of computation speed our processor will have, we would need to test the IMU in the lab under real conditions to see how we will need to apply our algorithms to the data. We also would need to test the maneuverability of the drone based on our instructions. We have found drone APIs online that contain instructions for moving in a certain direction, but we would need to calibrate the instructions to account for the specific movements of the drone we buy in response to these instructions.

With lab access, we would take measurements of positional data from the IMU while stationary and while doing different motions to see how much noise to expect in flight. We would use this error margin to be able to accurately compute how much computation power we need to reduce noise to acceptable levels, and ensure the MSP430 or a different processor is powerful enough for real-time processing. Another important test would be to measure the drone's response to instructions from the API. Once we have a set of movements mapped to instructions, we would need to calibrate the drone's speed so that it moves slowly enough to not cause disorientation.

3.3 Ethics and Safety

3.3.1 Concerns

Our project raises several safety concerns involving using batteries and flying drones, and privacy issues related to mounting a camera on a drone. We need to ensure that our project is ethical by following the IEEE Code of Ethics [9] as a guideline to help mitigate all safety and privacy issues.

Because we are using batteries to power our device, we introduce a risk of electrocution. Our battery system will output a maximum of 5.3 V and has the possibility of causing a current that can shock or burn a person. We will need to ensure that our device is safe either by shielding the user or limiting the current output of the battery module.

One more concern is that the drone might collide with a person or someone's personal property. Due to the high rotational velocity of a drone's rotors, this can cause significant injuries and property damage. We need to ensure that we follow the IEEE code of ethics concerning avoiding damage to people and property [9]. To this end, we will ensure that we have a clear area when testing our drone so that we mitigate the risk of any injury or property damage.

Additionally, we must take into consideration how the user will be affected by the movements of the drone that are transmitted through the headset. Overly high velocity or sharp movements can create disconnect between the expected and perceived motion of the drone, and this can cause motion sickness in the user. Noticeable delays in transmission can also unnecessarily strain the user and cause motion sickness.

Finally, we must be aware of the privacy risk introduced when mounting cameras on flying vehicles. A drone with a camera on it could be used to intrude upon the privacy of others by recording private events or flying over private property. We must ensure to the best of our ability that we and any potential users of the product do not violate anyone's privacy while operating the drone.

3.3.2 Risk Mitigation/Resolution

In order to address the risk of electrocution, we will mention that the minimum current to cause a slight sensation in the hands is 0.6 mA and the minimum current to cause pain is 3.5 mA [10]. Our battery system will have a maximum voltage of 5.3 V, which when applied to the worst case scenario of contact with a wet hand (1,000 Ohms) leads to 5.3 mA [10]. This means that the only way someone could experience a shock from our equipment would be to directly touch the positive and negative terminals of the battery with a wet hand. In order to prevent even this small chance of a shock, we will be encasing the battery in silicone and creating a casing for the whole circuit.

In order to avoid colliding with people or property, we will adhere to all of the state and federal laws that are applicable to small UAV in the state of Illinois [11]. This mostly restricts any form of flight in restricted zones and flying in proximity of state-owned infrastructures like transmission lines, public offices, or airports etc. In addition, we can use the built in system statuses and telemetry to determine if there are any potential risks, as the drone is aware of obstructions to its field of vision and height off the ground at any time [12]. In order to avoid the risk of colliding with obstacles outside of the drone's field of vision, we will ensure that any changes in the yaw of the camera are due to changes in the yaw of the drone as a whole so that the camera is always facing the direction of the drone's motion. For our current implementation we will also limit the movement of the drone so that it can only move or turn, but not both at the same time, this way it cannot crash into something that is outside of its line of sight. Future implementations may address this issue in a different way.

In order to minimize nausea in the user due to the VR headset, the drone's movements must match as closely as possible to the user's perception. Research into motion sickness shows that people who suffer from motion sickness can adapt to more intense activity over time, but the strength of the initial stimulus which causes motion sickness varies from person to person[13]. Because of this, we cannot guarantee that there will be no issues with any user, but we can minimize the risk by limiting the maximum velocity of the drone. This can be done through a calibration process in which the user tells the drone what speeds they feel comfortable with, and the drone sets its maximum velocity relative to that, or we can limit the velocity of the drone to a speed that should be relatively safe for a majority of users. As a reference velocity, we can use a bike in order to maximize the likelihood that the user will be comfortable with the drone, so we can limit the maximum velocity of the drone to approximately 10-15 mph.

The final concern raised by our project is that of privacy. This is less of a concern about our system and more of a concern with drone-mounted cameras in general. Laws in many states have been passed to control the usage of drones, including several laws that prohibit flying drones over private property and certain public institutions without permission [11]. We will only be flying on locations where we have express permission from the land owner or an authorized representative. Regarding any end users breaking the law should our project be put on the market, we believe that including a warning that tells users to ensure that they have read local regulations will be sufficient. We can never remove the possibility that someone uses our technology for nefarious purposes, but the responsibility for breaking the law will be on the shoulders of the user.

3.4 Future Work/Project Improvements

Given a full year to work on this project, we would be able to make significant improvements to the design. We would like to be able to make more accurate readings of the user's position, which would likely involve using a different technology from IMUs. One option would be object recognition on several video feeds, which could let us track the user's head motions without the large noise problem that an IMU introduces. This would require much more processing power and more in-depth algorithms, which we would need more time to implement than just one semester and a more powerful processor.

We could also include more features to ease the burden on the user. As we briefly mentioned in section 3.3, the user can experience motion sickness when the perceived motion is different from their expectation. We could include a calibration system that lets the user adjust the drone's speed to what they are comfortable with while watching through VR. The associated fine-tuning of maximum and minimum speed values as well as sensitivity would be outside the scope of a shorter time frame.

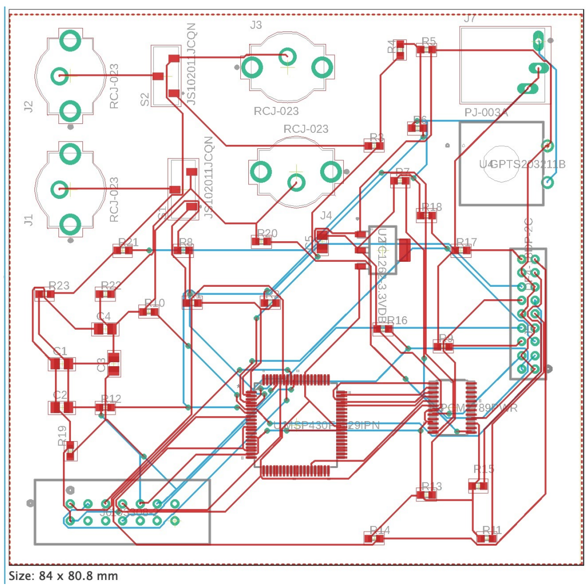
We can further improve the safety of our design by including obstacle detection in the features. Adding side cameras or proximity sensors to the drone would allow us to warn the user or stop the drone when an obstacle is nearby. This would allow us to expand functionality to let the user more freely control the drone (moving while turning and diagonal movement) while continuing to prioritize the safety of the user and their surroundings.

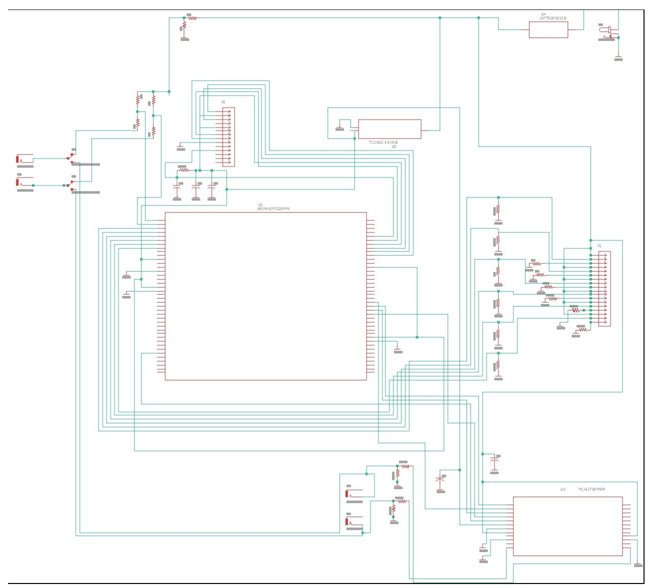
4. Progress Made on First Project

During implementation of the first project, we were able to finalize a PCB design to put in an order before Spring Break as well as bias the sensing circuit to provide the proper maximum voltage input to our microprocessor. For Circuit 1 (test circuit), R_LED = 1 kΩ and R_Out = 10 kΩ. For the final biased Circuit 2, R_LED = 220Ω and R_Out = 10 kΩ. This allowed us to get a maximum voltage of 3.5 V when a reflective surface was just in front of the emitter/receiver combo and a very distinguishable series of voltages for other distances, which could be inputted into the MCU.

Distance to reflective surface	0.05 in	1 in	2 in	3 in	4 in	No surface
Max output voltage, circuit 1	0.61V	0.27 V	0.13 V	0.067 V	0.052 V	0.044 V
Max output voltage, circuit 2	3.5V	1.22 V	0.62 V	0.29 V	0.21 V	0.137 V

Schematics and PCB layout for the HipHop Double Dutch Express:





References

[1] University of Illinois, "ECE 445 Web Board: Projects: Sp2019: Group 22" *University of Illinois.* [Online]. Available:https://courses.engr.illinois.edu/ece445/project.asp?id=7116 [Accessed: April 16, 2020].

[2] Federal Aviation Administration, "Become a Drone Pilot", *faa.gov*, Jan. 28, 2020. [Online]. Available: https://www.faa.gov/uas/commercial_operators/become_a_drone_pilot/ [Accessed: May 6, 2020].

[3] "DJI OCUSYNC 2.0: WHAT YOU NEED TO KNOW ABOUT THIS FPV TRANSMISSION SYSTEM," *djibestdrones.com*, Sept. 4, 2018. [Online]. Available: http://djibestdrones.com/dji-ocusync-2-0/#Compatibility. [Accessed: April 16, 2020].

[4] "Testing the range of NRF24L01+ modules". EmbedBlog. http://embedblog.eu/?p=187 [Accessed: April 16, 2020]

[5] Bosch Sensortec, "Small, versatile 9-axis sensor module," BMX055 datasheet, Revised Nov. 7, 2014. [Online]. Available: https://www.digikey.com/product-detail/en/bosch-sensortec/BMX055/828-1060-2-ND/6136303 [Accessed: April 16, 2020].

[6] H. Mokhtarzadeh, T. Colten. "Small UAV Position and Attitude, Raw Sensor, and Aerial Imagery Data Collected over Farm Field with Surveyed Markers" (Feb 2015). Accessed at: https://conservancy.umn.edu/bitstream/handle/11299/170139/flight_data.csv?sequence=2&isAllowed=y

[7] Erick Macias, Daniel Torres, and Sourabh Ravindran,"Nine-Axis Sensor Fusion Using Direction Cosine Matrix Algorithm on MSP430F5xx," Texas Instruments, Tech. Report. SLAA518A, Feb. 2012. Available: http://www.ti.com/lit/an/slaa518a/slaa518a.pdf [Accessed: April 1, 2020].

[8] Manon Kok, Jeroen D. Hol and Thomas B. Sch¨on (2017), "Using Inertial Sensors for Position and Orientation Estimation", Foundations and Trends in Signal Processing: Vol. 11: No. 1-2, pp 1-153. Available: https://arxiv.org/pdf/1704.06053.pdf [Accessed: April 16, 2020].

[9] "IEEE Code of Ethics". IEEE. https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed: April 3, 2020].

[10] EETech Media. "Ohm's Law (again!)". All About Circuits. https://www.allaboutcircuits.com/textbook/direct-current/chpt-3/ohms-law-again/

[11] "Illinois Drone Laws". StateDroneLaw. https://statedronelaw.com/state/illinois/ [Accessed: April 3, 2020].

[12] DJI Technical Staff, *Mavic 2 Specifications*, DJI, 2019. [Online]. Available: https://www.dji.com/mavic-2/info#specs [Accessed: April 16, 2020.]

[13] Lackner, James R. "Motion sickness: more than nausea and vomiting." Experimental brain research vol. 232,8 (2014): 2493-510. doi:10.1007/s00221-014-4008-8. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4112051/ [Accessed: May 4, 2020].