University of Colorado
Department of Aerospace Engineering Sciences
Senior Projects - ASEN 4018

Project RiBBIT

River **B**athymetry **B**ased **I**ntegrated **T**echnology

Project Final Report (PFR)

May 3, 2021

Information

0.1 Project Customers

Name:	R. Steven Nerem	Name:	Toby Minear
Email:	nerem@colorado.edu	Email:	tminear@colorado.edu

0.2 Team Members

Project Manager: Megan Jones Email: Megan.Jones-1@colorado.edu.com Phone: (720) 302-3353	Lead Systems Engineer: Mikaela Dobbin Email: Mikaela.Dobbin@colorado.edu Phone: (303) 990-0316
Name: Abdullah Almugairin Email: abal3405@colorado.edu Phone: (310) 648-4952	Name: Paul Andler Email: paul.andler@colorado.edu Phone: (925) 214-8647
Name: Andrew Benham Email: andrew.benham@colorado.edu Phone: (970) 690-2975	Name: Daniel Crook Email: Daniel.M.Crook@colorado.edu Phone: (650) 421-1130
Name: Courtney Gilliam Email: cogi5884@colorado.edu Phone: (832)-385-7014	Name: Jessica Knoblock Email: Jessica.Knoblock@colorado.edu Phone: (702) 755-9603
Name: Phil Miceli Email: philip.miceli@colorado.edu Phone: (303) 875-7278	Name: Samuel Razumovskiy Email: sara6569@colorado.edu Phone: (360) 281-7503

Contents

List of Figures

List of Tables

List of Acronyms and Definitions

Acronyms

- GNSS - Global Navigation Satellite System
- RTK - Real-Time Kinematic
- PPK - Post-Processed Kinematic
- SONAR - Sound Navigation and Ranging
- SLAM - Simultaneous Localization and Mapping
- FPS - Frames per Second

Definitions

- Bathymetry - The study of the beds or floors of water bodies, including oceans, rivers, streams, and lakes.
- Ideal Conditions - According to the USGS, ideal river surveying conditions are defined as "a smooth, mirror-like water surface with steady, uniform, non-varying flow conditions in the stream reach where the discharge measurements will be taken." [6]
  More specifically, the river needs to:
  - be a "reasonably straight channel with streamlines parallel to each other."[6]
  - have a "stable streambed free of large rocks, weeds, and obstructions that would create eddies, slack water, and turbulence."[6]
- Payload - Includes any attachment to the drone including the instrument suite and any additional mounting materials.
- Instrument system - Refers to the instrument and any associated mechanisms being used to capture the measurement defined by the row category.
- Instrument suite - Includes all instruments, electronics, any associated mechanisms, and housing.
- Data - Defined by bathymetric data, velocity capture images, and positional data from GNSS receivers.
- AstraLite - Company which specializes in drone-mounted LiDAR systems used for bathymetric modeling.

Nomenclature

ρ_w	=	Density of Water
V_w	=	Volume of water displace by float
m_f	=	Mass of Float
M	=	Lateral Metacenter
M_L	=	Lateral Metacenter
G	=	Center of Gravity of Float
B	=	Center of Buoyancy
θ_{error}	=	Measured Float Displacement Angle with Error

1 Project Purpose

Rivers are a critical resource to monitor due to their contributions to agriculture, urban development, hazard monitoring, and environmental monitoring. As global warming persists, the water cycle is a key indicator to track as it links the atmospheric, terrestrial, and oceanic processes [1]. One vital measurement to effectively track this resource is river discharge, or the units of volume per unit time. To calculate this, the river cross-sectional area must be multiplied by the river's velocity [6]. This measurement enables the amount of water available for human consumption or risk mitigation to be quantified. Currently, there is a lack of updated and accurate global data for river discharge, especially in hard-to-access environments. Existing Earth-orbiting satellites simply do not provide the accuracy and precision that in-situ, and UAV-mounted systems can enable [18].

To collect these river measurements, scientists from the United States Geological Survey (USGS) travel the world to collect in-situ river measurements. In the past, these measurements have been collected utilizing a tagline system, pulling a boat across the water with an acoustic depth-sensing instrument and velocity tracker [6]. Alternatively, helicopters towing radar systems have been utilized. However, both of these methods have proven to be expensive and dangerous in certain environments. To solve this problem, a river surveying device shall be designed to enable maximum science output for a portion of the cost of traditional methods.

In correspondence with Steve Nerem, Toby Minear, and the Ann and H.J. Smead Aerospace Engineering Sciences department at CU Boulder, this team shall design, manufacture, and test a drone-mounted sensor system to gather river depth profile and velocity data in hard-to-access areas for the purpose of monitoring river discharge.

2 Project Objectives and Functional Requirements

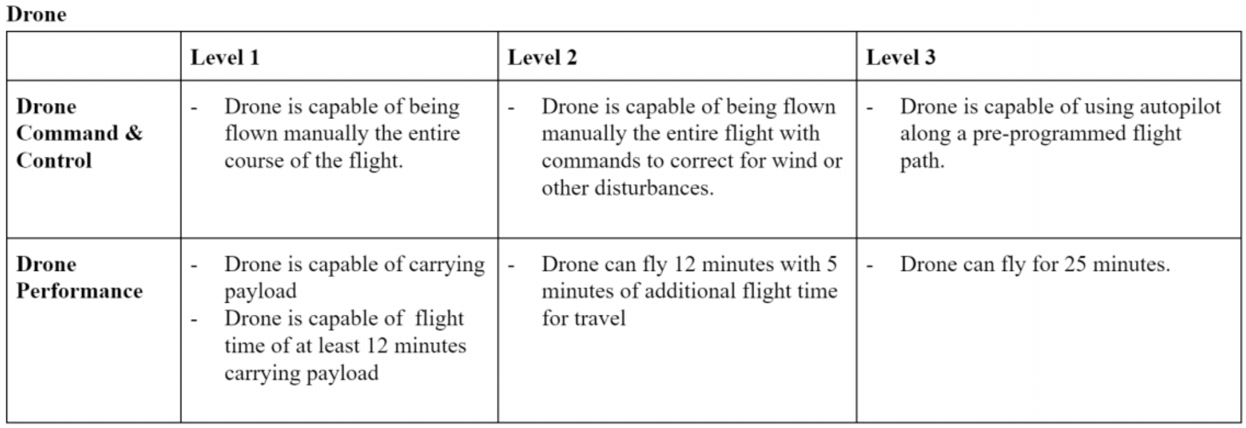
Project RiBBIT has a number of different project elements that must work in unison to enable a functional and successful project. The levels of success depict the various categories of the project and what constitutes success for each criteria. The levels of success range from Level 1 to Level 3:

Level 1 Minimum level of success for the project element. If all components reach Level 1 success, the RiBBIT system will be baseline functional.

Level 2 Further development and complexity of project elements, building upon the Level 1 levels of success.

Level 3 Encompasses both Level 1 and Level 2 success criteria. If the project achieves Level 3 success, the project will be fully functional and has achieved all requirements and objectives.

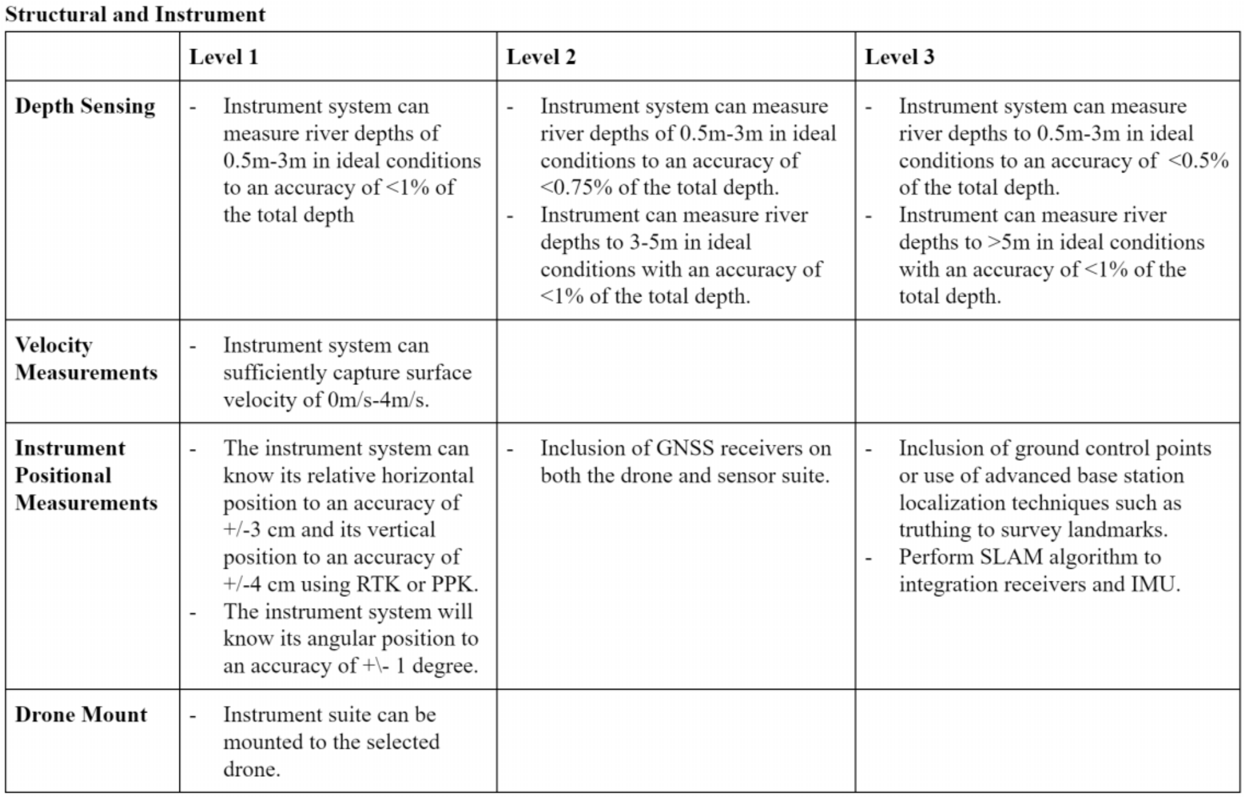
Levels of success are built upon one another, with the mission reaching baseline success at Level 1. In the event of time or budgetary constraints, the higher levels of success may not be accomplished. Each table contains the levels of success for the project subsystems. The rows establish the identified categories for defining levels of success. The columns define the conditions required in order to reach each specified level of success.



Drone

	Level 1	Level 2	Level 3
Drone Command & Control	- Drone is capable of being flown manually the entire course of the flight.	- Drone is capable of being flown manually the entire flight with commands to correct for wind or other disturbances.	- Drone is capable of using autopilot along a pre-programmed flight path.
Drone Performance	- Drone is capable of carrying payload - Drone is capable of flight time of at least 12 minutes carrying payload	- Drone can fly 12 minutes with 5 minutes of additional flight time for travel	- Drone can fly for 25 minutes.

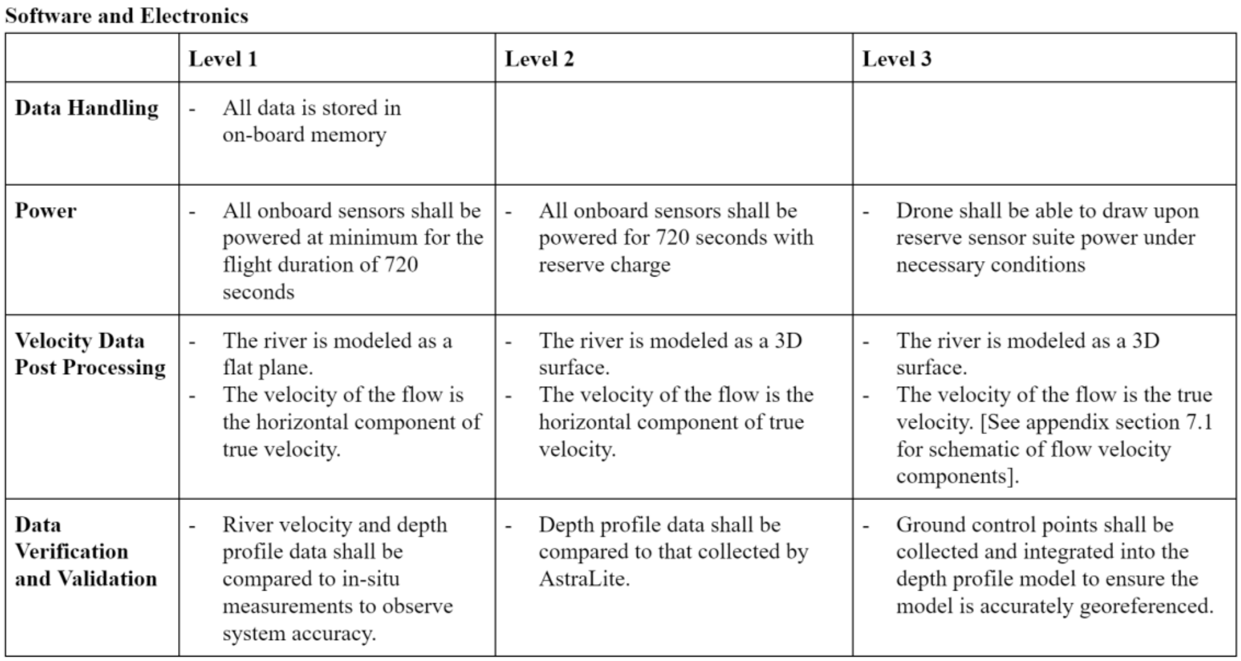
Figure 1: Drone Levels of Success



Structural and Instrument

	Level 1	Level 2	Level 3
Depth Sensing	- Instrument system can measure river depths of 0.5m-3m in ideal conditions to an accuracy of <1% of the total depth	- Instrument system can measure river depths of 0.5m-3m in ideal conditions to an accuracy of <0.75% of the total depth. - Instrument can measure river depths to 3-5m in ideal conditions with an accuracy of <1% of the total depth.	- Instrument system can measure river depths to 0.5m-3m in ideal conditions to an accuracy of <0.5% of the total depth. - Instrument can measure river depths to >5m in ideal conditions with an accuracy of <1% of the total depth.
Velocity Measurements	- Instrument system can sufficiently capture surface velocity of 0m/s-4m/s.		
Instrument Positional Measurements	- The instrument system can know its relative horizontal position to an accuracy of +/-3 cm and its vertical position to an accuracy of +/-4 cm using RTK or PPK. - The instrument system will know its angular position to an accuracy of +\- 1 degree.	- Inclusion of GNSS receivers on both the drone and sensor suite.	- Inclusion of ground control points or use of advanced base station localization techniques such as truthing to survey landmarks. - Perform SLAM algorithm to integration receivers and IMU.
Drone Mount	- Instrument suite can be mounted to the selected drone.		

Figure 2: Structural and Instrument Levels of Success



Software and Electronics

	Level 1	Level 2	Level 3
Data Handling	- All data is stored in on-board memory		
Power	- All onboard sensors shall be powered at minimum for the flight duration of 720 seconds	- All onboard sensors shall be powered for 720 seconds with reserve charge	- Drone shall be able to draw upon reserve sensor suite power under necessary conditions
Velocity Data Post Processing	- The river is modeled as a flat plane. - The velocity of the flow is the horizontal component of true velocity.	- The river is modeled as a 3D surface. - The velocity of the flow is the horizontal component of true velocity.	- The river is modeled as a 3D surface. - The velocity of the flow is the true velocity. [See appendix section 7.1 for schematic of flow velocity components].
Data Verification and Validation	- River velocity and depth profile data shall be compared to in-situ measurements to observe system accuracy.	- Depth profile data shall be compared to that collected by AstraLite.	- Ground control points shall be collected and integrated into the depth profile model to ensure the model is accurately georeferenced.

Figure 3: Software and Electronics Levels of Success

2.1 CONOPS

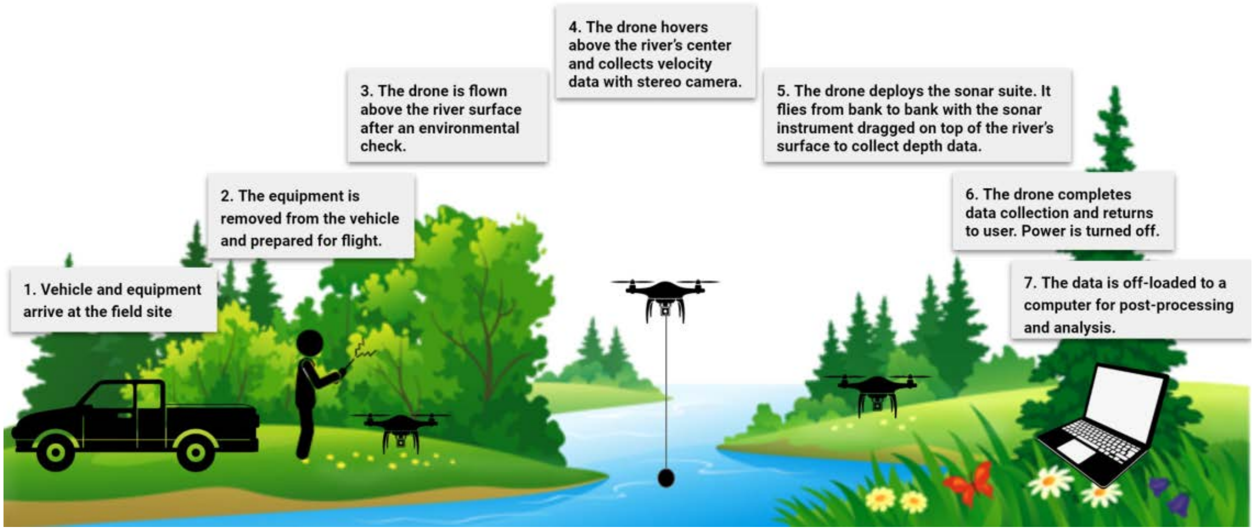

1. Vehicle and equipment arrive at the field site
2. The equipment is removed from the vehicle and prepared for flight.
3. The drone is flown above the river surface after an environmental check.
4. The drone hovers above the river's center and collects velocity data with stereo camera.
5. The drone deploys the sonar suite. It flies from bank to bank with the sonar instrument dragged on top of the river's surface to collect depth data.
6. The drone completes data collection and returns to user. Power is turned off.
7. The data is off-loaded to a computer for post-processing and analysis.

Figure 4: RiBBIT Concept of Operations

The purpose of this mission is to provide a low-cost, light, and long-range drone with mounted sensors to enable river discharge calculation through velocity and bathymetric data collection. Figure 4 shows the mission procedures of the RiBBIT system. The mission begins with a set-up period where the vehicle and equipment arrive at the river site. The test site will be chosen to ensure the environmental conditions are suitable and non-hazardous. The drone and sensor system are then prepared for flight by qualified personnel. The drone shall be elevated to a pre-determined altitude (dependent on the river's width) and video data of the river will be collected from a nadir pointed stereo camera. This data will be utilized to compute river velocity through particle image velicometry. Next, the drone will deploy a float holding a sonar device and be flown from riverbank to riverbank, perpendicular to the stream flow. While in flight, the sensor package shall collect bathymetric data. The collected data will be saved on-board. Once the system has completed the data collection phase, the deployable sonar unit will be rappelled up and the drone will be flown back to the user, and the system will be powered off. After the data collection has been conducted, the data shall be off-loaded to a computer. Post-processing of the data is then carried out to produce a cross sectional depth profile, mean velocity field, and cross-sectional river discharge.

2.2 Functional Block Diagram

The functional block diagram in Figure 5 below depicts the connections between the ground segment, drone, and the instrument suite.

The ground segment consists of a stationary GNSS Base Station, a visual observer, and the pilot. The stationary GNSS Base Station will receive positional data from the GNSS satellites. The Visual Observer will have their eyes on the environment surrounding the drone to mitigate the risk of damaging the drone or instrument suite such as hidden tree branches or river debris. The pilot will be operating the drone through the survey campaign. To do this, the pilot will use Mission Planner software which comes with the drone. Next, the drone itself will be purchased ready to fly. Shown in this category are the components built in to the drone that the team may need to interface with. There will be a rigid mechanical connection for the instrument suite to mount with the drone.

The instrument suite is broken up into two sections: the On-Board Sensor Unit, and the Deployable Sensor Unit. The On-Board Sensor unit includes a Jetson Nano as the main computer, which is connected to a battery for power supply. The Jetson Nano is connected to the ZED 2 Stereo Camera capture river velocity as well as a the U-Blox GPS Receiver for positional data. The Jetson Nano will store the stereo camera and positional data to an on-board SD card. Additionally, the On-Board Sensor Unit is mechanically attached to the Deployable Sensor Unit which is controlled by a servo motor system that is triggered by the pilot's remote. The Deployable Sensor Unit will be below the drone in order to safely make contact with the water surface.

The purpose of the Deployable Sensor Unit is to make contact with the river surface in order to collect sonar measurements of the river depth. The design of this unit is a stable boat design with the sonar mounted underneath for data collection. By pilot's command, it is lowered using a servo-motor mechanism which is powered through the on-board unit. There is a fishing line run through the motor spool and running to the deployable float to lower the payload with this command. The Deployable Unit is controlled through the Arduino Uno and is powered via on-board battery (independent of the On-Board Sensor Unit). Finally, the Deployable Unit also includes an IMU in order to track the angular displacements of the unit for sonar measurement corrections in depth profile post-processing.

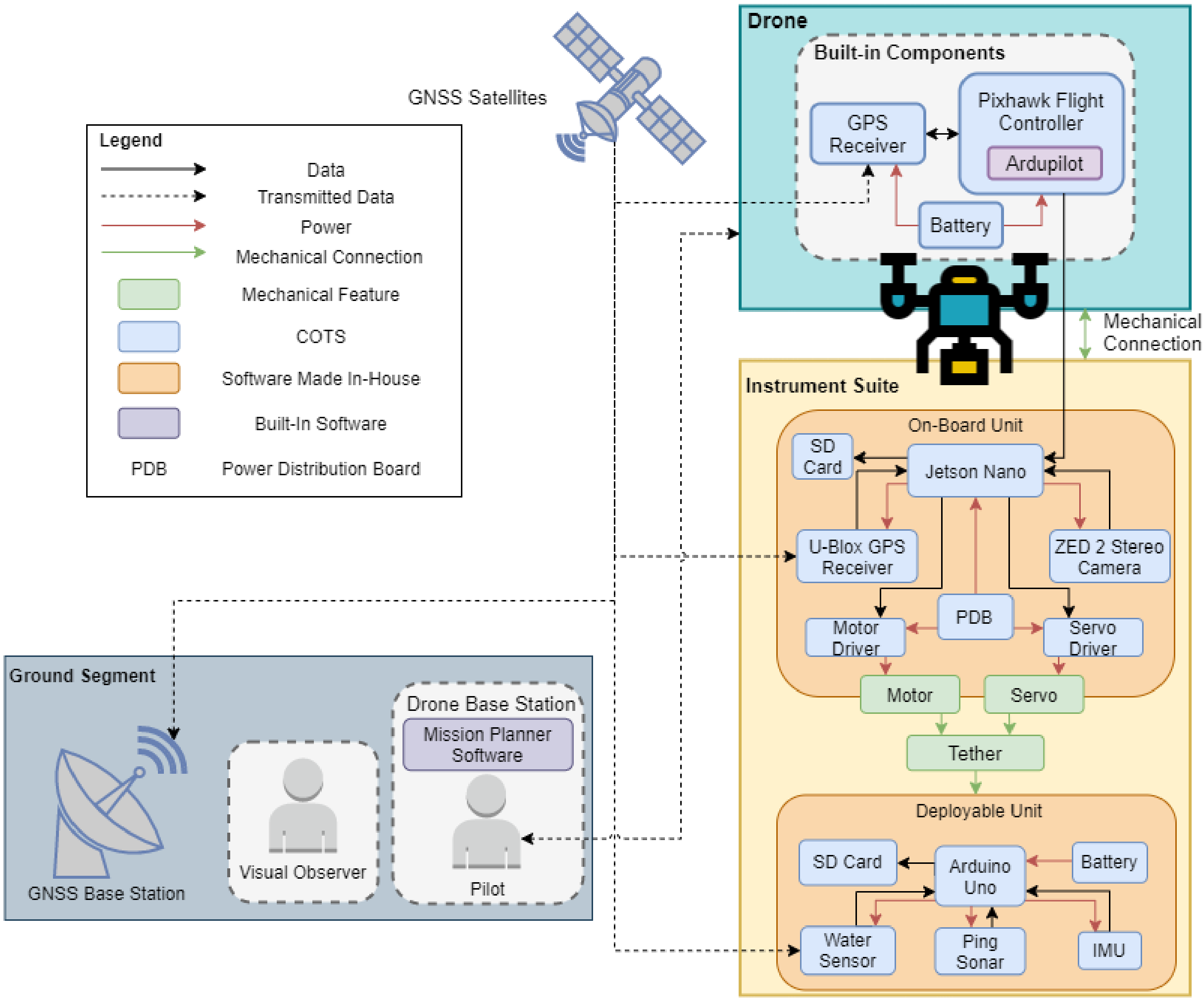


Figure 5: Functional Block Diagram

2.3 Mission (Functional) Requirements

The following outlines the high level functional requirements that were derived from the project objectives. These functional requirements were the basis of the requirement flow down, and aided in what trade studies were initially conducted. These requirements additionally enabled the team to structure necessary testing to be completed.

FR1. RiBBIT shall be an unmanned aerial vehicle (UAV) system.

Motivation: The system must be able to reach to locations difficult to access by foot such as canyons and areas with restricted human access. The drone would avoid these complications by being able to fly over/around obstacles.

FR2. RiBBIT shall be able to operate in customer specified river conditions.

Motivation: There are many rivers with various conditions. In order to narrow the scope of the project, the customer has specified river conditions comparable to Boulder Creek.

FR3. RiBBIT shall include an instrument suite payload that is compatible with the Tarot 680.

Motivation: This requirement is driven by the fact the instrument suite must be able to be mounted to the selected drone.

FR4. The instrument suite shall be capable of measuring the bathymetric profile of a river cross section from one bank to the other, perpendicular to the current.

Motivation: In order to measure river discharge, RiBBIT must profile the cross section of the river by measuring water surface level, width, and river depth perpendicular to the current as well as the water flow rate.

FR5. The instrument suite shall be capable of measuring the stream flow of a river cross section.

Motivation: See motivation for FR4.

FR6. RiBBIT shall be able to power and command all instruments and sensors.

Motivation: This requirement is driven by the need to be able to power, turn off and on instruments and sensors as well as store acquired data on board.

FR7. The collected data shall be post-processed after data acquisition.

Motivation: This requirement is specifies that the on-board computer will not be calculating river discharge simultaneously with the data acquisition.

FR8. The UAV shall comply with all FAA requirements.

Motivation: This requirement ensures that the UAV will be safe to fly for both the community and the team.

3 Final Design

In this section, a detailed design of all of Team RiBBIT's project elements and how they satisfy design requirements will be discussed. How design requirements were developed from the functional requirements will be overviewed. Given these requirements structured our project's design priorities, it is important to emphasize them before leading into the design choices. The instrument choices enabling bathymetric and velocity measurements will be described, leading into the uncertainty quantification of these measurements in data post-processing. As collecting and post-processing this scientific data is the central objective of the mission, design emphasis will be given to the sonar and stereo camera data collection, error quantification, and post-processing. Then, the structural design components will be described to outline how the team successfully housed, powered, controlled, and deployed the deployable instrument suite while the UAV was in flight.

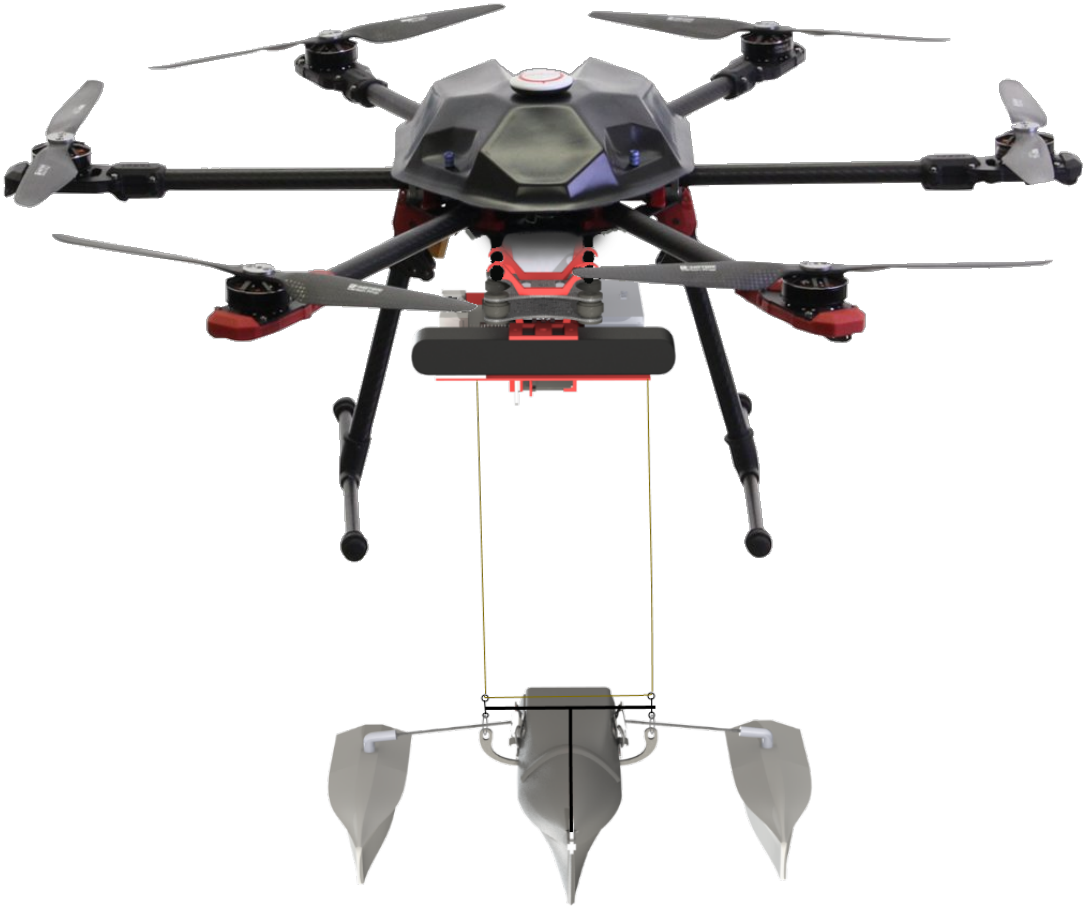
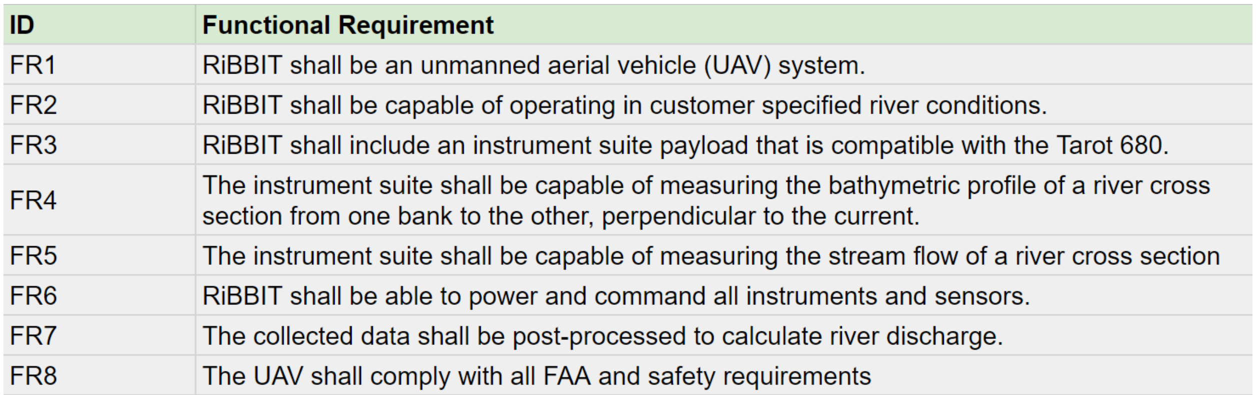


Figure 6: Full System Model

3.1 Requirements Development

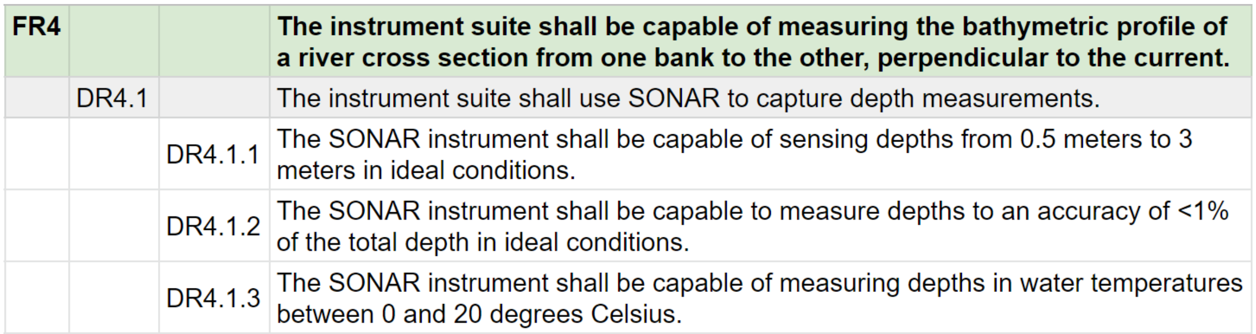
Figure 7 below specifies the functional requirements for the project. Each functional requirement has an ID which is used to map the functional requirements to the derived design requirements. The rationale for each of these requirements can be found in section 2.3.



ID	Functional Requirement
FR1	RiBBIT shall be an unmanned aerial vehicle (UAV) system.
FR2	RiBBIT shall be capable of operating in customer specified river conditions.
FR3	RiBBIT shall include an instrument suite payload that is compatible with the Tarot 680.
FR4	The instrument suite shall be capable of measuring the bathymetric profile of a river cross section from one bank to the other, perpendicular to the current.
FR5	The instrument suite shall be capable of measuring the stream flow of a river cross section
FR6	RiBBIT shall be able to power and command all instruments and sensors.
FR7	The collected data shall be post-processed to calculate river discharge.
FR8	The UAV shall comply with all FAA and safety requirements

Figure 7: Function Requirements

Each of the functional requirements includes several child requirements (which sometimes have their own children requirements) that were developed as the project progressed over the semester. For brevity, an example of a three-tiered parent-child relationship for FR4 are 8 and the full table of requirements are included in Appendix Section 10.15. These requirements will be referenced throughout the remainder of this report, specifically in the verification and validation section, in order to evaluate the project's success. The nested structure depict the flow down from the functional requirement and the associated children requirements.



FR4			**The instrument suite shall be capable of measuring the bathymetric profile of a river cross section from one bank to the other, perpendicular to the current.**
	DR4.1		The instrument suite shall use SONAR to capture depth measurements.
		DR4.1.1	The SONAR instrument shall be capable of sensing depths from 0.5 meters to 3 meters in ideal conditions.
		DR4.1.2	The SONAR instrument shall be capable to measure depths to an accuracy of <1% of the total depth in ideal conditions.
		DR4.1.3	The SONAR instrument shall be capable of measuring depths in water temperatures between 0 and 20 degrees Celsius.

Figure 8: Functional Requirement Flow Down

The logical flow-down from function requirement to design requirements in this example come from the project objectives and chosen instrument. FR4 specifies that the instrument suite needs to measure the bathymetric profile or the river depth of a river cross section perpendicular to the current. The flow-down from this is seen in DR4.1, specifies how FR4 will be achieved through the project design, i.e. by using sonar which was the result of the trade study process, The following requirement flow down set up how well we need to be able to measure the depth in order to have useful data. DR4.1.1 specifies the depths which we need to confidently measure, DR4.1.2 specifies the accuracy which we need to measure the depth using sonar, and finally, DR4.1.3 specifies the operational temperature of the sonar instrument. The child requirements of DR4.1 all expand on what is require by using sonar to capture successful depth measurements. This logic is applied all of the functional requirements, resulting in the nested requirements list as a result of the design evolution from the first semester.

3.2 Bathymetry Technique

Based on the trade studies conducted the first semester, the most optimal bathymetric survey sensor to meet the project objectives and constraints is the Ping SONAR by BlueRobotics. This sensor excelled in all categories with exception of the contact vs. non-contact section. This sensor decision led to the planning and formation of the deployable float unit held beneath the UAV. The only other sensor system that was considered was the LIDAR system which was outside of the group's budget and also under performed in the mass, size and water performance sections. The Ping SONAR is a sensor that was able to fulfill the needs of the project in a cost effective manner ($309 including shipping and taxes), provide great range and accuracy in the measurements it takes and also has the lowest mass of all sensors considered. All of these factors improved the flight time of the UAV due to it having to carry less weight.

Figure 9: Ping SONAR Device

3.2.1 Depth Profile Error Uncertainty Quantification

When it comes to the uncertainty associated with depth measurements, it becomes crucial for us to know what angular displacements the SONAR float can experience while still producing data that falls within our accuracy requirement of <1% error of the true total depth. So to identify this angular displacement limit the group started by considering the accuracies to which the two on board instruments could measure. These being the Ping echosounder (±0.5%) and the BNO055 IMU (±1 degree). From here a test scenario was created which, for this analysis, assumed a flat river bottom with constant depth. To then perform the error analysis on the total system a true depth was calculated for each angle on which analysis would be performed (-30 to +30 degrees). Following this a random erroneous data set was created using a uniform distribution of errors with these errors being applied to both the measured depth and angle. The process of which is shown below.

$$\theta_{error} = \theta_{true} + rand(-1, 1) \tag{1}$$

$$Depth_{error} = (Depth_{measured} + Depth_{measured} * rand(-1, 1) * 0.005) * cos(\theta_{error}) \tag{2}$$

The depth being multiplied by the cosine of our error adjusted displacement angle as we are interested in the adjacent side of the triangle (depth along the gravity vector) that is created when taking data at an angular displacement that is non-zero. Given this we now have our "measured" depth from our test data as well as our true depth. Given these two values it then becomes possible to calculate the total percentage error.
Given that this process was random, the program would produce a slightly different result every time it was run. That being said, for the number of times that it was run the group did not receive a result in which the 1% error bound was broken with less than a 2 degree displacement. However, to give an extra safety margin the decision was made to set our acceptable data threshold to be less than or equal to 20 degrees to account for possible environmental factors that may not be accounted for in the current analysis. For a plot of the described results please refer to figure 92 in the appendix below.

3.2.2 SONAR Float Angular Displacement Correction Technique

Based on feedback the group received, a decision was made to change from the previously selected 3-axis gyroscope to a 9 degree of freedom inertial measurement unit, the BNO055. The reason as to why the decision was made to switch these two components was multifaceted. Firstly, the group did not take the drift of the gyroscope into account which would make taking accurate angular position measurements almost impossible after a short period of time. Furthermore, the gyroscope only took in one stream of data. This being the instantaneous acceleration in the X,Y and Z directions. The improvement with regards to the IMU being that it is not only has a three axis gyroscope but also a a magnetic field heading sensor and gravity vector sensor. The latter being especially usefully to us due to purpose of this sensor being to measure the relative angular displacement between it and the gravity vector. Lastly, the reason why this specific IMU was chosen to fill this role was not only for its sensor package but also the on board sensor fusion and openly available arduino libraries making its implementation as simple as possible.

3.3 Velocity Technique

Given the results of an initial stereo-camera trade study and research on Particle Image Velocimetry (PIV), the team selected the Stereo Lab ZED 2 for river surface data video collection to enable velocity post-processing. It's 2.2k video resolution, high frame rate, and in flight adjustable camera settings made it a strong candidate for accurate velocity calculation. It also has a slightly higher quoted stereoscopic accuracy (consistent with its larger distance between cameras) than the Real Sense and a wider FOV meaning it can take more accurate and further measurements from the river. Lastly, it includes video streaming capabilities and documentation for post-processing and analysis in MATLAB. The ZED 2 retails for $449.00. How the image data collected via this camera enables river surface velocity measurements to be computed is outlined in the software portion of our final design.

Figure 10: ZED 2 Stereo Camera

3.4 Float Design

The design of the float was driven by both functional mission requirements and their associated design requirements seen in Figure 8. Specifically, FR.3 creates a weight and size upper threshold, and FR. 4 sets the basis for stability requirement. These then were further refined by their associated design requirements; DR3.1-3.4, which further refines the weight requirements and additionally defines the nature of the relationship between the drone and float. These stipulates that the float should be imparting as little a moment as possible on the drone, whether it is stored underneath the drone, being lowered or raised, or on the water surface. DR3.1 specifically sets the payload weight at a maximum of 2kg. After accounting for non-float payload components, the float all associated components (mainly the electronics box) were required to weigh no more than 1.5kg. The float ended up coming in at 670 grams without pontoons and 910 grams with them attached. All of DR4 pertains to the float as this flows from the requirement of measuring the bathymetric profile. Most important of all is DR4.2 which states: *The SONAR instrument shall be located on a deployable float.* So interestingly enough, the group didn't actually choose the float as a solution, rather it was an outcome of choosing the SONAR instrument to meet the requirement of creating a bathymetric profile. The other design requirements that had the biggest influence on the float design were DR4.2.1: *The float shall be designed such that the bottom 2.5 cm of the SONAR instrument is submerged under*

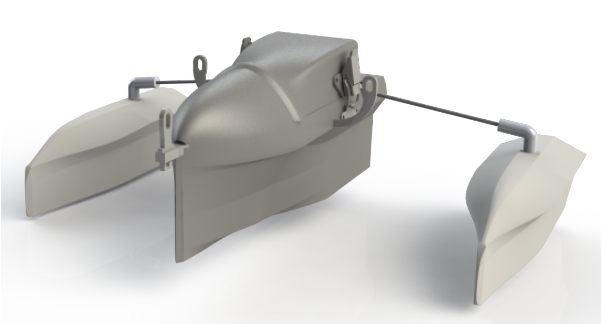


Figure 11: The SONAR Float Design

water. This is probably one of the most difficult requirement to meet since it involves a intimate understanding of hydrodynamics and hydrostatics and the interplay between the floats center of mass, center of buoyancy and attachment point. Additionally, boats in motion also experience a 'lifting' force since water is being directed/deflected under the boat as it travels through the water. Due to the complex nature of hydrodynamics the team decided to stick to high level modeling and an eyeball approach for validation (see sonar float validation section later in the report). The float does submerges the sonar by 3cm in stationary water. This was one aspect that was modeled as it is a more straightforward task since it only involves Archimedes principle which can be paraphrased as: The buoyant force acting upwards on the float is equal to the weight of the volume of the water displaced by the float, this force then acts at the centroid of the submerged volume [26]. So that meant our float needed to be designed such that when 3cm of it was submerged, the weight of that displaced water was equal to our floats weight and thus the float would be in equilibrium. For those who prefer equations over words, this equilibrium is shown in Equation 3 with Archimedes principle on the left and the force of gravity acting on the float on the right. The reason gravity does not appear in Equation 3 is left as an exercise for the reader.

$$-\rho_w * V_w = m_f \tag{3}$$

The last requirement that drove the major float design elements was DR4.2.3: *The float shall be designed such that the amount of time the float is angularly displaced by +/- 20 degrees is minimized.* The reason for this was the same as the previously mentioned requirement and are both directly tied to the ability of the SONAR to produce a bathymetric profile. Unfortunately, this requirement is just as challenging if not more complex to robustly satisfy compared to the previous one. This is because unlike DR4.2.1 which can be thought of as generally only involving motion in two directions (one from water moving along the float and one from the float moving up and down). DR4.2.3 involves coupled moments, rotations and transnational motion in every direction.

This takes the complexity of hydrodynamics to an entirely new level. For example, when deployed, the float is pulled from from one side of the river to the other from the front most attachment point (similar to Figure 12 from [27]). Thus, the group was curious how this pulling would turn the boat as an unfavorable reaction to this pull could not only impart moments on the drone and break FR4 but may also disturb the SONAR out of its 20° range. Foremost, this pull to the side creates a moment arm as it is acting in front of the boats center of mass, but the group had a feeling it wouldn't react like a plane or even like any number of the generic potato's presented in ASEN 2003. Not only did this feeling turn out to be correct but it was far worse than we could have anticipated; the pivot point of a boat *"Is only an imaginary point"*[27]. That is to say the pivot point of a boat does not exist, the interested reader is encouraged to investigate *S.G Seo*[27] for further information but essentially: because water is a yielding material *"...any active force turning a ship will cause a drifting motion at the same time (as a yawing motion)...(modeling) the unsteady process accurately is very difficult, if not impossible, particularly when various forces are involved."* The governing principles involved in solving this problem were mainly Archimedes principle again as well as a boats longitudinal centre of flotation (LCF) and metacenter (M)[1]. Major assumptions include those employed by [26] and [27] and include small disturbances, no vertical movement and the float acts as a plainer rigid body. These assumptions allowed for two models to be developed related to DR4.2.3; a roll model which investigated how much force would be required to roll the float out of the 20° requirement and a weather vain stability model which demonstrated the ability of the float to always remain pointed upstream. Both these models were validated (see Sonar Float Verification in Section 5) with the float demonstrating weather-vane and roll stability.

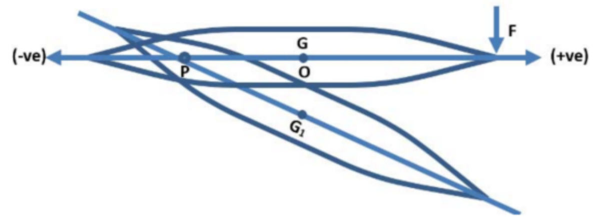


Figure 12: A Stationary Body in Water Rotating about its Imaginary Pivot Point, from [27].

[1] A visualization of the LCF and M can be found in the Appendix

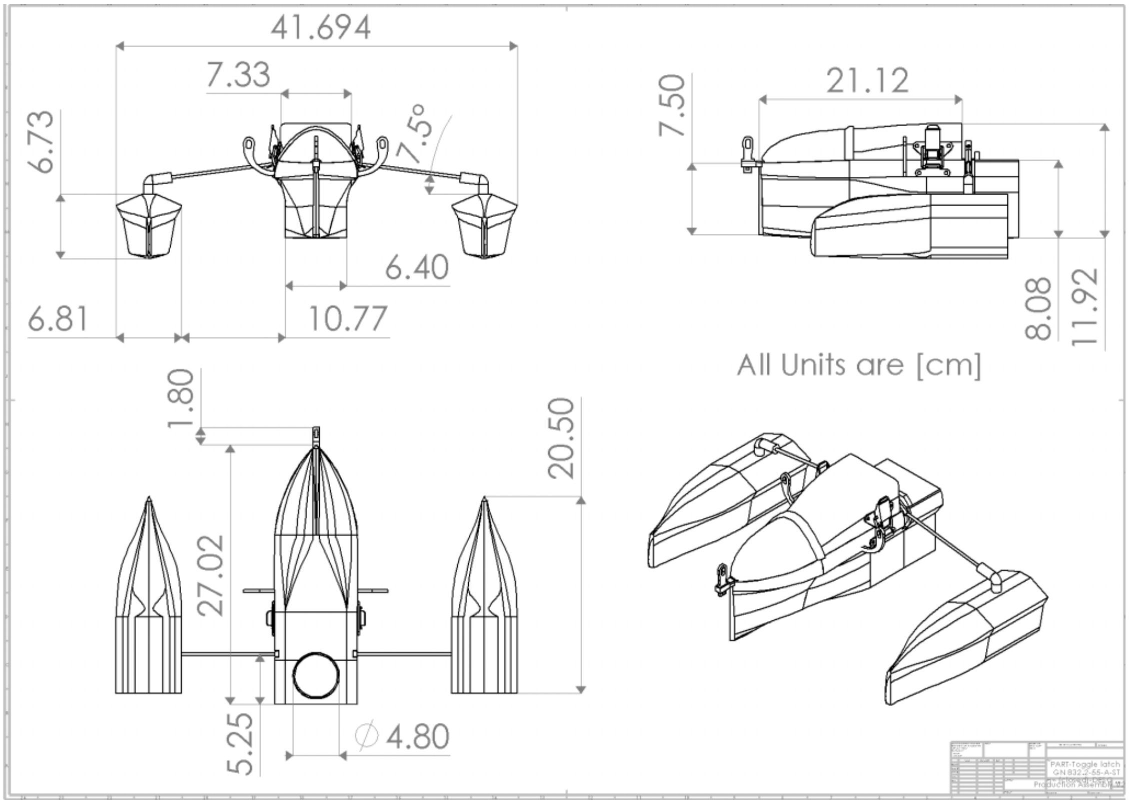


Figure 13: A Detailed Drawing of Major Float Dimensions

3.4.1 Hull Design

The hull of the float is the main structure and includes the connections to the drone as well as carries the electronics box. Additionally, because it is the biggest structure it contributes the most to the weight, buoyancy and drag of the float. It is for this reason the hull was considered for a detailed design analysis. For our analysis purposes the hull is mainly tied to DR3.3 and 3.4 which dictate that the payload imparts as little moments and external forces on the drone as possible. Specifically the bow (front part of the hull) has the most influence on this as it is the leading edge and responsible for the overwhelming majority of interaction with the water. The bow design chosen by the group can be described as a combination of an Axe bow and X-bow. An Axe bow presents a sharp edge to the water to efficiently cut through the water with minimal drag [28]. The X-bow is a relatively new design from the Ulstein Group (a marine enterprise specializing in shipbuilding and design) which has only been employed on large vessels which traverse heavy seas. This style of bow seeks to minimize disturbances from large waves [29]. The combination of these designs enabled both a low drag and smooth, even, traverse of rivers in all conditions tested.

3.4.2 Float Electronics Housing

Design Requirement 4.2.2 states: *The float shall be designed such that the electronic components are located inside of a waterproof housing.* This requirement was addressed by making the entire inner volume of the float waterproof instead of just a part of the hull. Figure 14 depicts a visualization of the electronics in the hull. The detailed breakdown of how the hull was manufactured, including challenges encountered, can be found in the following section under float manufacturing.

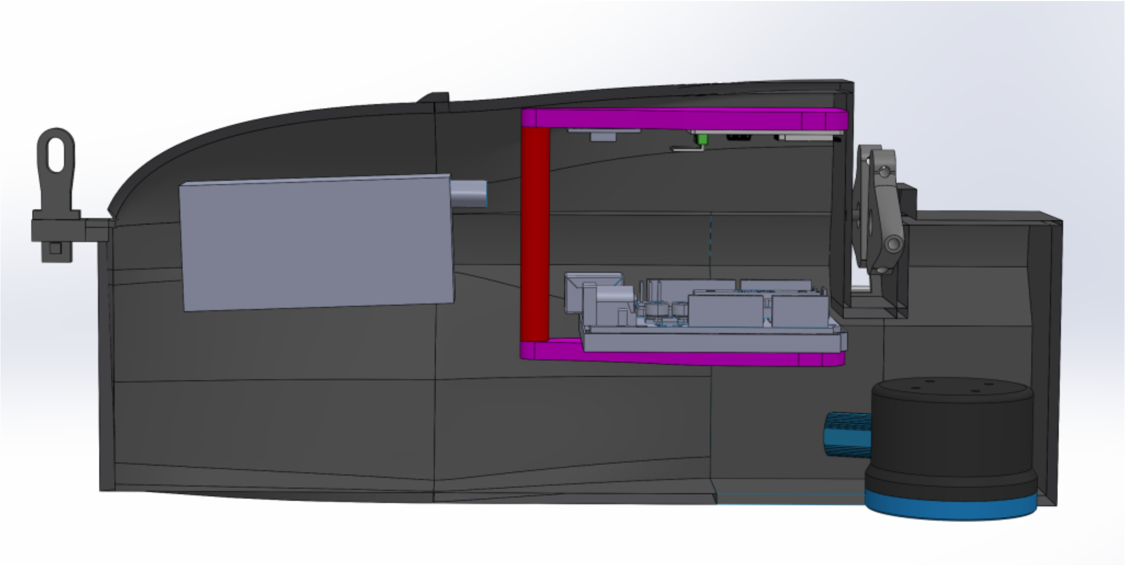


Figure 14: A section view showing how the electronics sit in the float

3.4.3 Pontoon Design

Given the aim of creating a float design that is as stable as possible while also producing the least amount of drag, the decision was made to go with the trimaran design as mentioned above. However, the float was limited by the space between the mounting points of the drone's landing legs. This presented a challenge as roll stability correlates to as wide of a base as possible. The solution was to make the arms connecting the side pontoons foldable. The arms were connected together with an elastic material such that when free of the landing legs they deploy outward. Conversely when being retrieved, the arms collided with the underside of the landing legs causing them to fold downward and become stored under the hull of the float for transport.

Since one of the objectives was to minimize the drag that is produced by the float, the stabilisers were designed to be as low profile as possible while still being able to provide sufficient righting moments in the case of a disturbance. This was accomplished by having them taper from a relatively thin section at the bottom to a wide section at the top. While under normal conditions the side floats present minimal area to the oncoming flow, but when submerged they produce an exponentially increasing buoyant force with regard to their vertical displacement. This then produces an exponentially increasing righting moment to counteract the original disturbance. When looking at the side stabilisers from above, it can be noted that the front tapers to a point. Through this design, the bow can split oncoming waves rather than going over them. This reduces unwanted disturbances translating from the stabilisers to the central hull.

3.5 Electronics and Data Handling

The selected computer to enable on-board command and data handling was the Jetson Nano. This decision was made due to the Nano's ability to interface with all on-board sensors, specifically, with the ZED 2 camera, which required specific hardware which the Jetson Nano is equipped with. On top of this, the Jetson Nano provides enough processing power to run simultaneous data collection programs while also sending out commands to other instruments incorporated on the on-board sensor suite. The chosen computer to deal with the deployable unit's command and data handling was the Arduino Uno. The Arduino Uno was chosen due to its ability to interface with all sensors on the deployable unit while still operating at a low weight and consuming minimal power. Below, the power, command, and data flow between all the on-board and deployable float units can be seen.

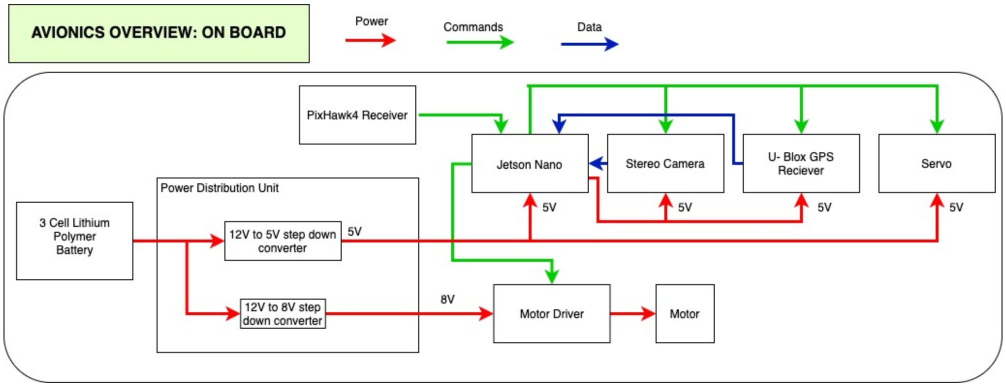


Figure 15: On Board Avionics Overview

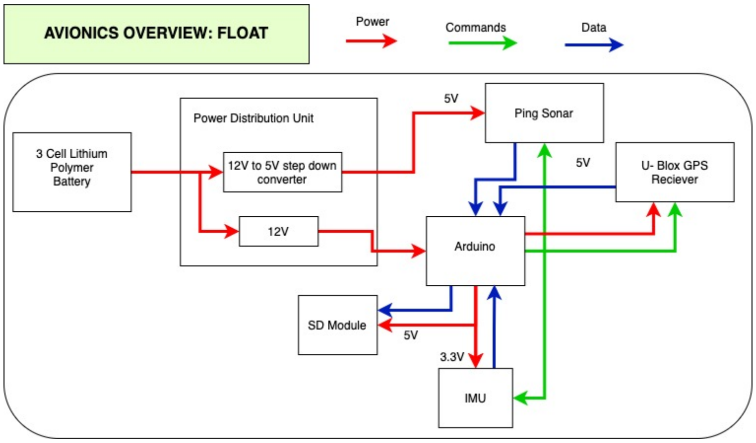


Figure 16: Deployable Unit Avionics Overview

3.5.1 Data Flow Analysis

Since RiBBIT utilized many sensors for data collection, it was very important that data requirements were met. Provided below is a table that shows all data needs for the on-board sensor suite.

Component	Maximum estimated data required
Jetson Nano	6 Gb
ZED 2	64 Gb
Ublox Receiver	1 Gb
Additional Code	5 Gb
Total	77 Gb

Table 1: Data requirements for on-board sensor unit

Because the maximum estimated data required for the on-board sensor unit is 77 Gb, RiBBIT used a 128 Gb microSD card to ensure that the unit did not run out of memory for data collection. Through testing, it was found that these models correctly predicted the amount of data needed. As a result, the 128 Gb microSD card provided the proper storage capacity which allowed for uninterrupted data collection. Below is a table that estimates data needs for the deployable sensor suite.

Component	Maximum estimated data required
Ping Sonar	2 Gb
Ublox Receiver	2 Gb
IMU	2 Gb
Total	6 Gb

Given a maximum estimated data requirement of 6 Gb for the deployable sensor suite, RiBBIT used a microSD card with a minimum storage capacity of 8 Gb. After conducting day in the life testing, it was found that the 8 Gb microSD card provided sufficient capacity for the necessary data collection.

3.5.2 Power Analysis

Battery sizing was a major portion of avionics design for RiBBIT. To find the best battery, a power analysis was conducted. Because the on-board avionics unit has a motor that operates at 12 volts, our battery needed to provide at least 12 volts. Another important part of battery selection was choosing a battery with the correct capacity. Below is a table that analyzes battery capacity needs.

Component:	Current flow over time:
Jetson Nano	(4A)(0.5h) = 2000mAh
ZED 2	(380mA)(0.5h) = 190mAh
Ublox Receiver	(67mA)(0.5h) = 33.5mAh
Servo	(250mA)(0.5h) = 125mAh
Motor	(100mA)(0.5h) = 50mAh
Total:	2398.5mAh

Table 2: Power consumption analysis for on-board sensor suite

As shown above, the selected battery needs to provide 12 volts and have at least 2398.5 mAh. The solution to these design requirements is procuring a 3 cell lithium-polymer battery with a capacity of 3000 mAh. Through power testing, we were able to conclude that the chosen battery provided the necessary power for the entire sensor suite to operate.

Power requirements for the deployable avionics unit include providing a 7-12 volt operating range to power the Arduino Uno. Provided below is a table that specifies battery capacity requirements for the deployable unit.

Component	Current flow over time:
Arduino Uno	(80mA)(0.5h) = 40mAh
Ping Sonar	(100mA)(0.5h) = 50mAh
Ublox Receiver	(67mA)(0.5h) = 33.5mAh
IMU	(12.3mA)(0.5h) = 6.15
Total:	129.65 mAh

Table 3: Power consumption analysis for deployable sensor suite

The deployable avionics unit has a battery capacity requirement of 129.65 mAh. The design solution to provide enough power is by selecting a 3 cell lithium-polymer battery with a capacity of 1000 mAh. Through power testing, it was determined that the selected battery sufficiently met the deployable sensor suite power requirements.

3.6 UAV

After consideration of all criteria, as indicated in table 19, the most optimal flight vehicle for RiBBIT's system is the Tarot 680. The primary reasons why the Tarot 680 was chosen was its cost, payload capacity, and availability. The Tarot 680 setup costs $1,611 which leaves plenty of money to be spent on additional instruments, mountings, and battery. It was determined that the Tarot X6 was too expensive to purchase even though it has better performance. Given the payload capacity of the drone is 1.5kg, the mounted and deployable payload can easily be designed to meet weight requirements. Given the instrument suite can be customized to mount to varied drones, the team would suggest buying the Tarot X6 with the camera option in the future.

Additional analysis was performed to determine if the drone was able to be stable in a fast flowing river. The results showed that a hexacopter with a simple PID control system is capable of maintaining stability and that it satisfies DR 1.1-1.3. The analysis is covered in more detail in section 11.0.4.

3.6.1 Payload Housing Mount

The payload is mounted to the drone using a 3D printed board that uses 3D printed arms that snap onto the UAV's mounting rods. Below is the distribution of the individual instruments on the mounting plate. The individual components are positioned by their required connections, and holes are made for each component to be bolted down. The plate itself is 15x25x5cm, the mounting arms are about 10cm tall, and the vibration dampener mounting arms are around 2cm tall.

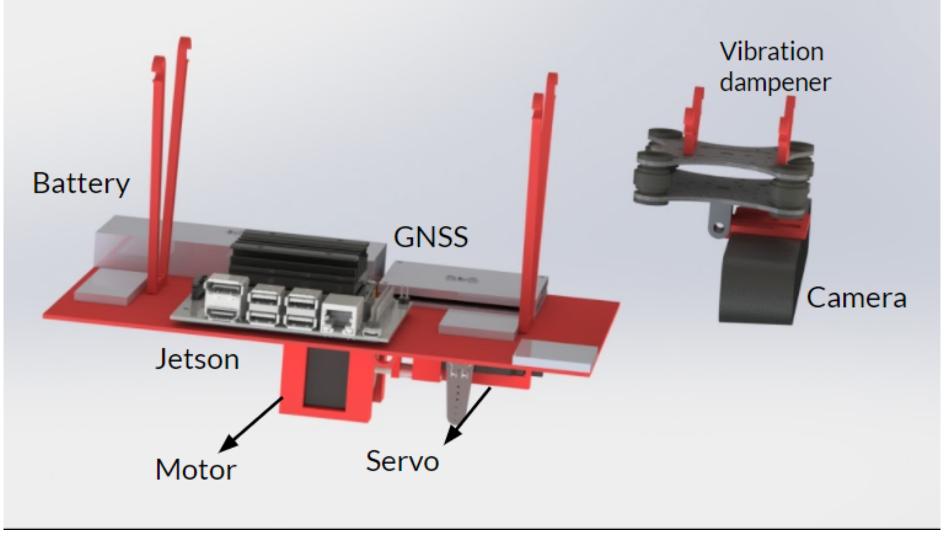
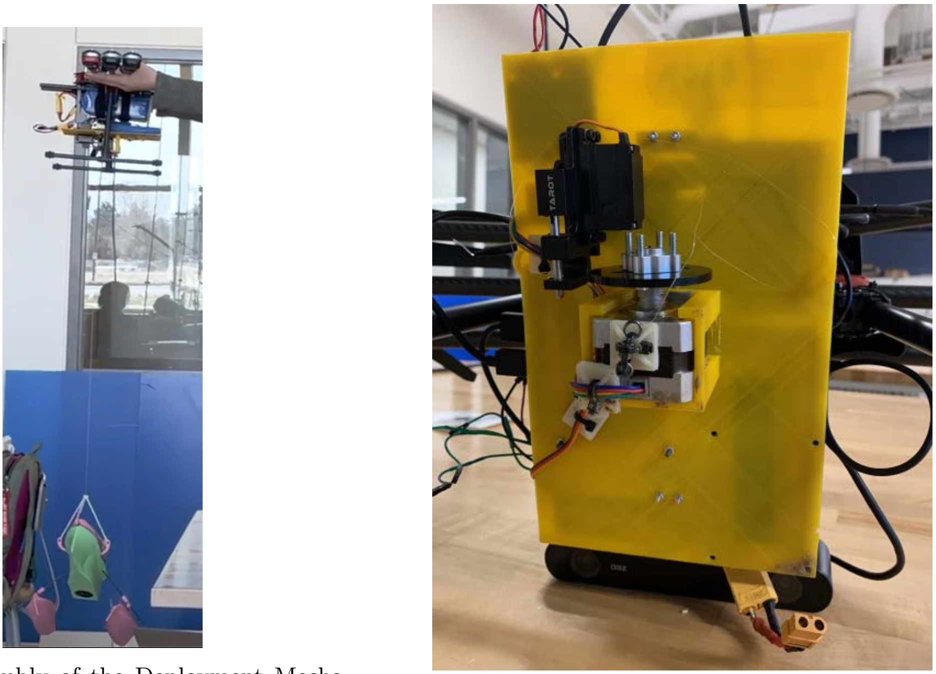


Figure 17: Payload Housing Mount

3.7 Float Deployment Mechanism

In order to be able to collect measurements on the surface of the water to ultimately calculate discharge, a safe and successful deployment mechanism of the float must be designed. In the current choice of deployment mechanism there are five critical components: (1) Motor; (2) Pulley; (3) Fishing line; (4) Servo; (5) Swivel. Fig. 18a below acts as a visual for the complete assembly of the components of the deployment mechanism.



(a) Full Assembly of the Deployment Mechanism

(b) A closer Look at the Components of the Deployment Mechanism

Figure 18: Deployment Mechanism

The motor can be seen in the center of the figure above and attached to the mounting plate using double sided tape. The motor chosen was a 12V bipolar stepper motor. The pulley, which is the second main component in the deployment mechanism, is attached to the stepper motor as shown above. The size of the pulley plays a role in the deployment and retraction time cycle. Initially in the CAD model, the pulley was designed with a 1.5 *cm* radius which yields a circumference of 9.4 *cm*. However, the quality of the 3D prints were poor potentially because how small the pulley was. Thus, a pulley was ordered online that had a 15.8mm radius. The fishing line was the third main component. The holding strength of the fishing line is 4.5 *kg* which was more than sufficient since the float only weighed close to 1 *kg*. Another critical component was the servo which is shown in black in the figure above. The purpose of the servo was to act as another attachment point for the fishing line as well as it has the capability to release the fishing line through a controllable latch that is only used in a case of an emergency. The fifth and last main component for the deployment mechanism was the swivel which helps prevent tangling in the fishing line and adjust to any twisting that may happen to the float. Furthermore, the entire deployment or retraction mechanism was controlled via the pix-hawk and the flight controller that the pilot of the drone may use when the drone is hovering above the point of the river where the team wants to collect the measurements.

3.8 Positional Determination and Geo-Referencing Technique

The first hardware change that effected localization was the decision to include an inertial measurement unit on the sonar float. Although the decision to include this instrument was primarily for sonar angle correction, its presence provides immense value to localization efforts.

As the drone and the entire sensor package are each self-contained modules designed to never interact, online control only happens within the built-in drone flight computer. This allows us to perform localization after the mission, thus providing more freedom in available methods and the ability to compare the performance of multiple methods. The initial selection of localization methodology was to select a variant of the Kalman filter. This led to the first investigations of measurement noise, a dynamics model, and its associated process noise. The measurement noises from the specification sheets were given in terms of percentage circular error probables (CEP).

After consultations with Professor Ahmed, it was decided to switch out the Kalman filter and use Simultaneous Localization and Mapping (SLAM) software for post-processing localization. The advantages of SLAM include not having to develop a high-fidelity dynamical model of the combined drone and float unit, utilizing techniques of loop-closure and landmark observation to reduce positional drift over long and short periods, and backwards smoothing. The variant of SLAM chosen is Georgia Tech Smoothing and Mapping (GTSAM), which models the system state as a factor graph. By incorporating unary measurements (direct state observations, such as GPS) and binary measurements (relating multiple states, such as accelerometer or gyroscope), GTSAM encodes the state's posterior density, represented by a mean and covariance, conditioned on all measurements. Thus, constructing a multivariate Gaussian density approximates the the real-world nonlinear dynamics. Loop closure takes advantage of the flight path's return to the same location multiple times, as the drone makes several passes across the river. The inclusion of such constraints can greatly increase accuracy of the full trajectory estimation. Observing stationary landmarks at different locations induces additional binary measurements, increasing the overall accuracy.

Out of the box, these receivers are configured to output NMEA format messages from all of their ports, which are pin headers, USB connections, and radio downlink. This was a problem initially because NMEA messages are too heavily processed to be post-processed relative to the Emlid base station data, which requires raw GNSS data. The pin headers only had the capability to output NMEA messages. The radio downlink required expansions of the power, weight, and volume budgets which exceeded capabilities. This left the u-blox's USB ports for receiving data. However, only the Jetson Nano was capable or receiving data from a USB cord. The Arduino Uno's could only receive NMEA data. Although these messages would be partially useful, the difficulty of either choosing a new microcontroller or integrating an Arduino-USB shield into the already tight setup was one of the main factors that led to the decision to not include a u-blox receiver on the float.

To account for the absence of a u-blox receiver on the float, an additional method of relative positioning was investigated at the recommendation of Professor Ahmed. AR Tags are a commonly used method of relative position and orientation estimation in robotics. Though their accuracy decreases with distance, it is quite reliable within short ranges. Given that the float only deploys from a distance of about 2 meters, this method of positional determination would not introduce significant error relative to the GNSS receiver's error. The specific variant of AR Tags chosen was the AprilTags initiative developed by the University of Michigan Department of Electrical Engineering and Computer Science. This open-source C++ software required the pixel dimensions of the incoming images, the optical focal length (in pixels) of the camera, and the physical dimensions of the printed AprilTag. Given these parameters, a folder of PNG or JPG could be fed to the software package which then returns the relative position and orientation of the tag. The code was modified to then append these values to a CSV file for further use in positional post-processing software. The AprilTag used for testing and missions is shown in Fig. 19.

Figure 19: April Tag

3.9 Software

While the manufactured components are a large element of RiBBIT, the software to both collect and post-process data is critical to ensure river discharge data products are effectively produced for scientists. Below, an overview of the scientific instruments, data outputs, and post-processing software applications are presented. Each of the instruments require data initiation and saving scripts written onto the Jetson Nano or Arduino Uno. The NVIDIA Jetson Nano is a small, powerful computer that will be used to control the on-board instruments; the stereo camera, GNSS receiver, and deployment mechanism control. The data collected by the instruments is then saved on the Jetson computer for later off-loading and post-processing. The Arduino is used on the deployed sonar float to facilitate the river depth data collection. It is responsible for the initiation and termination of data collection from the SONAR and IMU as well as saving that data to an on board SD card.

Below, an overview of all of the data collected and processing programs can be visualized. In the following sections, each component of this software system will be covered and discussed.

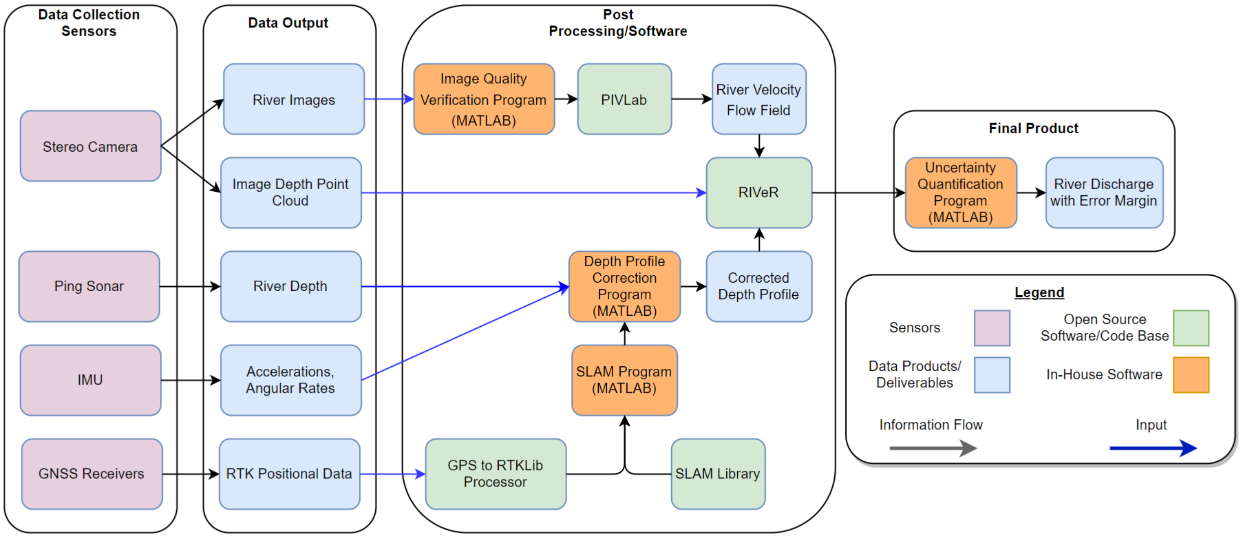


Figure 20: Post-Processing Software Overview

3.9.1 Sonar Data Post-Processing

The final depth profile of the river was created in a custom post-processing software application in Matlab. The purpose of the software is to combine the different data streams of both the SONAR and IMU. This being necessary to account for any angular displacements that the SONAR may have been subjected to during its time on the river. Furthermore, this software package also determines data points that are deemed to be erroneous or unfit. This connects to the confidence value produced by the SONAR for each data point. To begin with, the data produced by the SONAR and IMU is read into the environment and started as an array in the cell of a cell array. This being for scalability reasons as storing each data set into its own cell allows for simple scaling and simultaneous analysis of multiple data sets at once. Once the data was imported into the environment, the script called a function that checks to see which of the SONAR data points are valid. This is done by looking at the confidence value associated with every data point. The threshold here is set to 95% confidence to allow for minor uncertainties but remove all clearly false data. A major contributor for false data is the SONAR trying to measure distances less than 0.5m or attempting to measure distances in air. Having this confidence check has been extremely effective at only allowing data points that are valid and related to the test. Following this section is the angular correction section. Here the function takes in the pitch, roll, and measured depth from the aforementioned cell array and begins by computing the net angular displacement from the gravity vector given said roll and pitch measurements. From this net displacement angle (Θ) as well as our raw SONAR depth data, it then became possible to solve for the corrected depth using the sine rule. This being shown in the equation below.

$$CorrectedDepth = \frac{Uncorrected_{Depth} * \sin 90 - \Theta}{\sin 90} \tag{4}$$

This data is then stored in the corrected depth array. From here, the associated error bars are then calculated for each data point. These error bars contain both uncertainties associated with the SONAR as well as the IMU. For the SONAR, an error of 0.65% of the measured depth was applied. This was derived from the pool testing completed prior to the day in the life test. A further 1 angular displacement error was also set into the software to account for the IMUs' inherent uncertainty. From here, one then has the option of also calculating the cross-sectional area of the river at the point in question using a trapezoidal approximation. This step is completely optional since RIVeR does not require the value in question. From here it then becomes a matter of plotting both the depth data as well as the associated error bars at each point. During this step, the user also has full control over all aspects of plotting the data such as the data displayed, legend, labels, and font size.

3.9.2 Stereo Camera Error Uncertainty Quantification

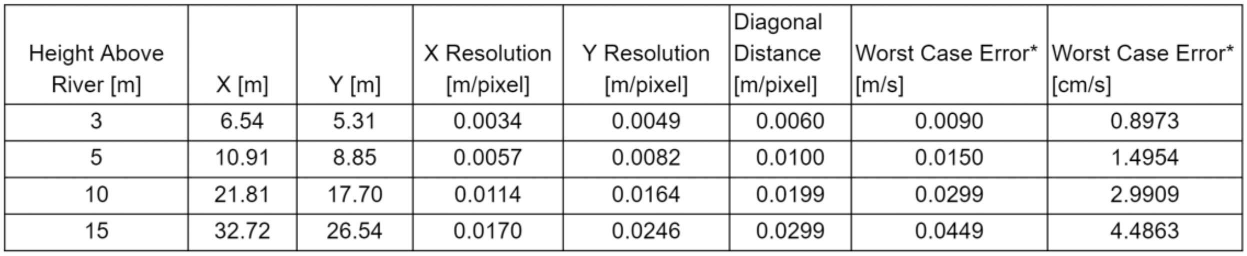
The surface velocity of the river is calculated using a Particle Image Velocimetry (PIV) tool developed in MATLAB called PIVLab. In general, there are several factors that contribute to the error associated with particle image velocimetry itself (not considering the software toolbox we will be using). These sources of error come from the flow field (such as shear and rotation in the flow), the physical system (such as camera lens distortion or environmental factors), the time response and scattering of identified tracer particles, illumination (reflections or background light), particle removal and algorithm response in the image analysis phase, vector replacement in the data validation phase, and spacial resolution in the data reduction phase. PIVlab is able to estimate the error in the computed u- and v-velocity components based on the image quality of the frames being analyzed which can be correlated to real-world velocity errors. In its development, several image processing methods were used to determine which could be used to minimize the uncertainty in the velocity calculation.

PIVlab uses cross correlation on an image pair to determine the most likely displacement of particles within an interrogation area. The statistical technique of cross correlation implements the following function:

$$C(m,n) = \sum_i \sum_j A(i,j)B(i-m, j-n) \tag{5}$$

where A and B are corresponding interrogation areas from images A and B. The most probable displacement is determined from the location of the intensity in the resulting correlation matrix. There are a few ways to go about solving equation 5. PIVlab performs peak-finding by running several passes of a Direct Fourier Transform (DFT) on the data set. In short, this technique minimizes the loss of information due to particle displacement.

In his PhD thesis, the developer of PIVlab performed several in-depth tests on various potential PIV methods to use in his software using synthetic control images. He also experimented with various image conditions to determine how the accuracy was affected by the quality of the data set. He concludes that the accuracy of PIVlab is dependent upon particle diameter, particle density, sensor noise, particle pair loss, and motion blur. Our biggest concern for minimizing our velocity error contribution is the particle diameter and particle density. The optimal conditions for these factors are defined by PIVlab to be a particle image diameter of 1-4 pixels, and a particle density of 1-8%. Rigorous tests and PIV analyses performed by the developer showed that with this software, if our images meet optimal conditions, we will be able to achieve a bias error less than 0.005 pixels, and a random error less than 0.02 pixels. Using the camera resolution of 1920 × 1080p and 30 frames per second, the bias and random error can be mapped at different camera heights to the velocity the error. This is done used trigonometry to relate the a camera pixel to real world distance, then by assuming maximum particle displacement error at the longest distance represented by each pixel (the diagonal distance), the bias and random error per pixel can be related to a distance and then correlated to time using the frame per second value of the camera. The results of the study are outlined in Figure 21 below.



Height Above River [m]	X [m]	Y [m]	X Resolution [m/pixel]	Y Resolution [m/pixel]	Diagonal Distance [m/pixel]	Worst Case Error* [m/s]	Worst Case Error* [cm/s]
3	6.54	5.31	0.0034	0.0049	0.0060	0.0090	0.8973
5	10.91	8.85	0.0057	0.0082	0.0100	0.0150	1.4954
10	21.81	17.70	0.0114	0.0164	0.0199	0.0299	2.9909
15	32.72	26.54	0.0170	0.0246	0.0299	0.0449	4.4863

Figure 21: Pixel Error to Velocity Error
(*assumes optimal image conditions)

3.9.3 Stereo Camera Data Pre-Processing - Particle Diameter

As mentioned above, the surface velocity of a river can be calculated using PIVlab if the images taken with the stereo camera meet optimal conditions. To determine if our data meets these conditions, the average particle diameter present in the images must be checked. If the particle diameter of the stereo camera-collected images is between one and four pixels, a bias error less than 0.005 pixels, a random error less than 0.02 pixels and a root-mean-square error less than 0.1 pixels can be achieved.

To calculate particle diameter, a Matlab script was developed that utilizes an image processing toolbox. The script takes an image as the input and outputs the average diameter of surface particles within the image. The tool also presents the user with an image displaying the identified particles and a histogram depicting the probability of occurrence of various particle diameters within the image. With this script, all river images can be pre-processed before uploading them to RIVeR and PIVlab to ensure that the bias and random errors lie within the desired ranges.

3.9.4 Stereo Camera Data Pre-Processing - Particle Density

In addition to particle diameter, the particle density of the river images must be checked. Similar to calculating particle diameter, a Matlab script was developed that utilizes the image processing toolbox to compute the density of particles within images. This script takes an RGB river image as an input and outputs the density of particles within a user-selected interrogation area of the image. The input RGB image is first converted to a grayscale image, enabling the particles to stand out as white pixels in the grayscale image. The user is then able to select the area of interest. This should be the region encompassing the desired river cross section where flow velocity and discharge can be computed. Next, the density of particles within that interrogation area can be found by computing the number of white pixels over the total amount of pixels in that section. Using this technique, the particle density for each frame of image data can be computed and determined if they meet optimal conditions, as defined by PIVLab. If the density of particles in our images is within 1%-8%, a root-mean-square error less than 0.1 pixels can be achieved. In the case that the video data does not meet optimal conditions for PIVlab analysis, the errors must be propagated using different methods of error quantification.

In the figures below, the particle diameter and particle density tools can be observed.

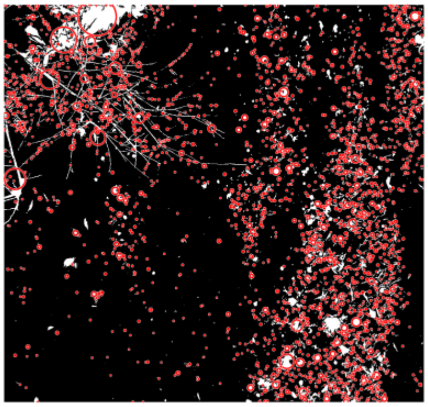


Figure 22: Particle Diameter

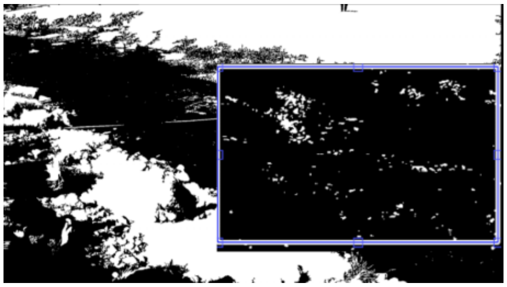


Figure 23: Particle Density Interrogation Window

3.9.5 Velocity and Discharge Post-Processing

As can be seen in the software overview visual in Figure 20, after the stereo camera has collected the river surface images from the location of interest, the images are then run through several software applications to extract velocity flow field measurements. The MATLAB software application Rectification of Image Velocity Results (RIVeR) is the main post-processing tool used. This software tool has a particle image velocimetry software, PIVLab, built in to compute surface velocity measurements. PIVLab compares sets of image frames side by side to compute velocity vectors of surface particles along the entire region of interest of the river selected.

Once both the river velocity flow field and corrected depth profile have been collected, these data are input into the Rectification of Image Velocity Results (RIVeR) software tool [8]. This MATLAB tool first rectifies the individual and mean surface velocity results with real world coordinates to get real surface velocity flow vectors in the region of interest. The software then estimates discharge at a specified cross section by either the mean section or mid-section method to compute total discharge (see [4]). This step correlates the depth and flow velocity at each specified station across the cross-section to get the corresponding discharge. The tool then adds up all of the discrete discharge measurements to provide the estimated total discharge. RIVeR then outputs the discharge summary in HTML format, providing total discharge in $\frac{m^3}{s}$, river width, area, and mean velocity. The depth, velocity, and discharge values are plotted along each station for easy data visualization.

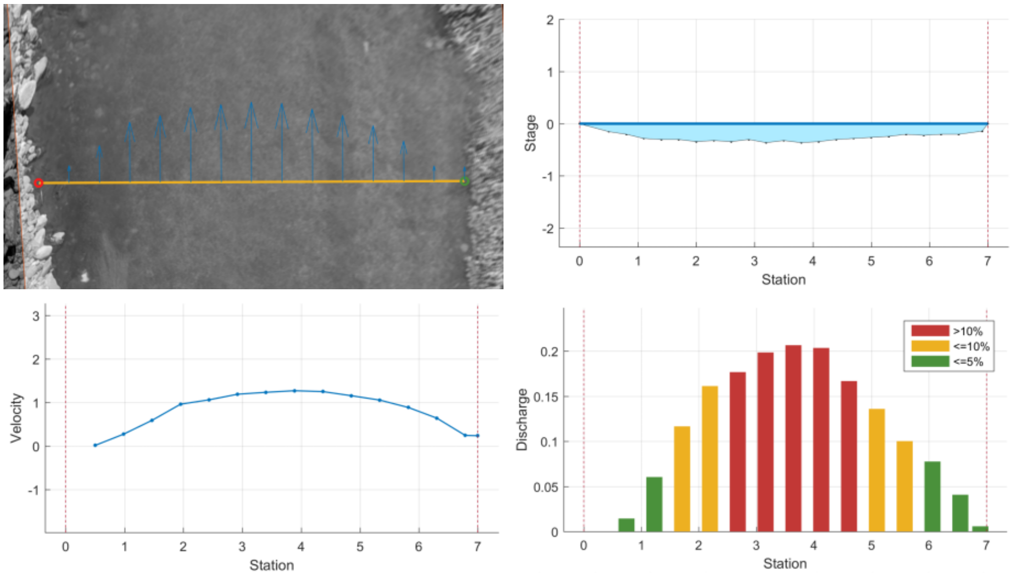


Figure 24: RIVeR Software Process

The set of images above depict the river flow vectors along a specified cross section (top left), the stage chart in [m] (top right), the velocity chart in [m/s] (bottom left), and the discharge chart in [$\frac{m^3}{s}$] (bottom right). These outputs provide an assortment of information beneficial for hydrological science.

3.9.6 Flight and Instrument Software

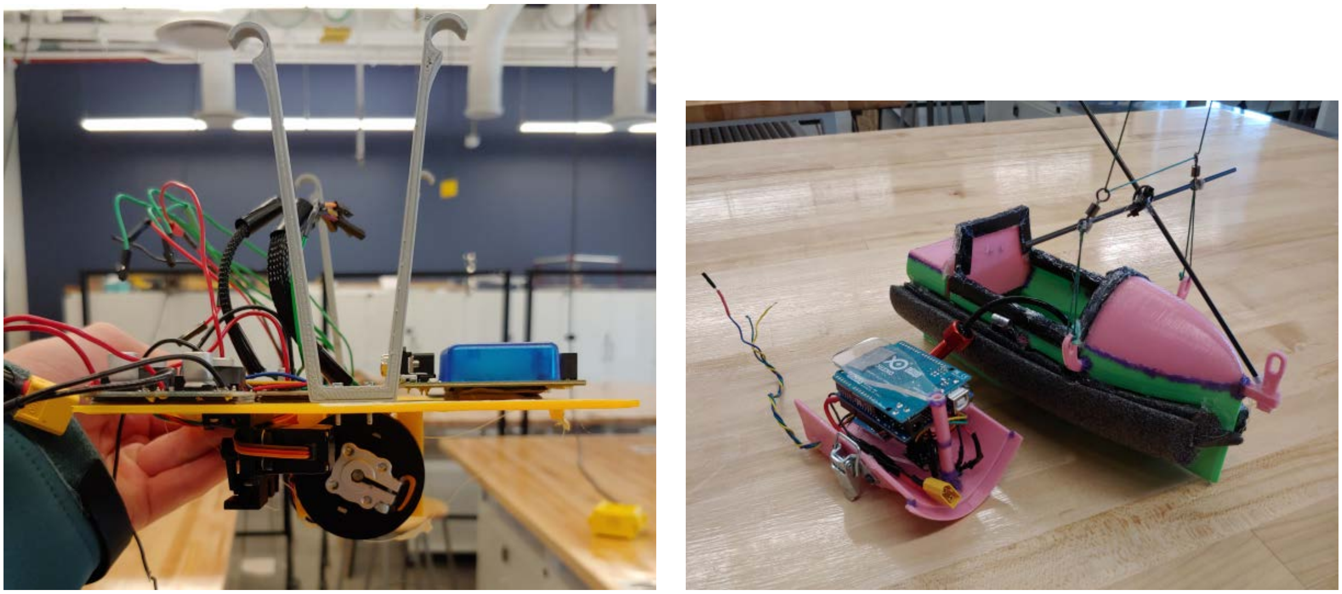
The instrument software on the deployed SONAR float was written in Arduino's native language. This being a language very similar to C/C++. The responsibilities and functions of this software are to initialize all onboard sensors and supporting electronics, start and stop data collection, store the collected data in the desired format on the local SD card as well as giving visual indications of these processes on externally mounted LEDs. The first action performed by the software in question is importing the necessary libraries to add support for the onboard electronics. These including packages for the SONAR, IMU, and SD card module. Specifically, the main purpose these packages serve is streamlining the data conversion between that native to the sensor to one easily understood by the Arduino. The other added benefit is automatically saving the data produced by each sensor into a unique structure. This allowing for easy access. It is also at this point where custom pin in and outs are defined for the SONAR and LED indicators. Following this, the "void setup()" function is entered. The purpose of this function is to initialize the IMU, SONAR, and SD card modules while also checking for a functional connection between them and the Arduino. It is also here where the file to which the SONAR data is written is created. Following its creation of the file, it is opened and the header saved to it before then closing the file. The final section of this function involves truing on one of the two LEDs. This shows to the user that all systems are powered, initialized, and have an established connection to the Arduino. The reason why the LED indicator code was placed at the end of this function was that if there were to be an issue with any of the components listed above. The function would get stuck at that point and eventually time out. Therefore not reaching the LED indicator section and subsequently alerting the user to an issue with the electronics due to their failure to initialize. This being expressed by the above-mentioned LED not turning on. Following the completion of all sections of the "void setup()" function, we then progress into the "void loop()" function. It as the word "loop" suggests, will constantly loop as long as the arduino does not lose power and no exit conditions are met. The first thing that is done at the beginning of every loop is monitoring the analog signal produced by the water sensor. The reason for this being that the only useful data we collect is data produced when the float is in contact with the water and the SONAR is submerged. Therefore, if the water sensor does not measure the presence of water we do not want to start collecting and saving data. If however water is detected, then we enter into the data collection subroutine within the loop function. It is here where we query both the SONAR and IMU for their current measurements and then save them to the SD card. This is also where we command the second LED to turn on and stay on for 0.5 seconds. This creating a blinking effect visible to the user and informing them that data collection is in progress. To save our data we first open the data file that we had previously created and saved the header to. We must open the same file as before such that our data get saved to the correct location. Once we have opened the file we will then save the pitch, roll, and yaw angles alongside the measured depth and related depth confidence interval using a comma separated value format. As soon as the current data point is saved the file is closed and the loop repeats. This being critical for the safety of the data that has been already collected. The reason is that in the situation that power is lost or the connection between the SD card shield and arduino is compromised, any data that has been written to an open file without closing said file would be lost. Therefore, it is in our best interest to open the file, write the data and close the file for every data point as any failure would mean that the most amount of data that can be lost is only one single entry. This entire process will repeat as long as the float is in contact with the water. Once it is lifted clear, the water sensor will report that it is no longer submerged and data collection will stop along with the flashing status LED. However, if the float assembly is placed into the river again, the Arduino will once again start collecting data and simply append it to that collected previously. It is only once the system is powered off that data cannot be collected again until all sensors pass the "void setup()" function.

4 Manufacturing

Table 4 shows a summary of the major components of the project and details on their manufacturing process. Figure 25 show the final manufactured and fully integrated float and drone platform.

Part	Manufacturing Technique	Additional Modifications
Drone Mount	3D Printed PLA	None
Camera Vibrational Dampeners	COTS	PLA printed attachments to drone rails
Drone Electronics Platform	3D Printed PLA	None
Float Rope Rigging	Welded Steel Bars	Applied Epoxy to Glue to Float
Float Structure	3D Printed PLA	- Coated with Polyurethane to waterproof. - Welded some parts together with PLA.
Pontoons	3D Printed PLA	Coated with Polyurethane to waterproof.
Electronics Shelving	Laser cut Polyacrlyic Shelves PLA Shelving Spacers	Used double sided tape to attach electronics to shelving

Table 4: A summary of manufacturing preformed for the project



(a) Final Manufactured Drone Platform

(b) Final Manufactured Float

Figure 25: Final Manufactured Systems

4.1 Float Manufacturing

The float itself was entirely 3D printed using PLA plastic and designed entirely on SolidWorks. There were two versions of the hull that were printed. The biggest challenge by far was the waterproofing of the float, with other obstacles including attaching the parts of the float together and incorporating the suspension system.

Waterproofing the float involved two different activities; the waterproofing of the PLA plastic itself and creating a waterproof seal where the electronics lid attached to the float hull. To waterproof the PLA plastic, a minimum of four coats of polyurethane were applied to both the inside and outside of the various members that make up the float body. Due to the welding technique described later in this section, some parts were coated before welding them to the rest of the float while other weren't coated until after being attached. The reason for this was that the polyurethane coat prevented the plastic from melting together properly so parts which required welding had to be coated after being attached. Since the electronics needed to be accessible to access the SD card module, the electronics were attached to a removable lid. To create a tight seal and prevent water from leaking in, a rubber gasket was added along the edge of the the float where the lid met the float. Then latches were added (can be seen in Figure 11) to clamp the lid down onto the float. This was somewhat successful, as will be elaborated on in the challenges section, but there were still leaks when when exposed to water. So, additional silicone chalking was added to the rubber gaskets that were much more squishy than the rubber. This worked but after a few weeks of taking the lid on and off the group noticed that some of the silicone was coming off and more needed to be added.

Because 3D printing PLA plastic does not allow for extremely complex shapes or overhangs, the fact the hull itself is hollow presented a problem. Notice in Figure 25 that the bottom part of the float is green while the top parts and lid are pink. This is because these parts needed to be printed separately and then attached together. Along the interface between pink and green parts is a purple PLA, this was added in order to weld the separate multiple float pieces into one and can be seen in Figure 26. This worked extremely well as it created very strong connections, to the point where the part itself seemed like it would break before the welded connection itself would. This allowed the team to create parts like they would have been if they were printed as one solid part.

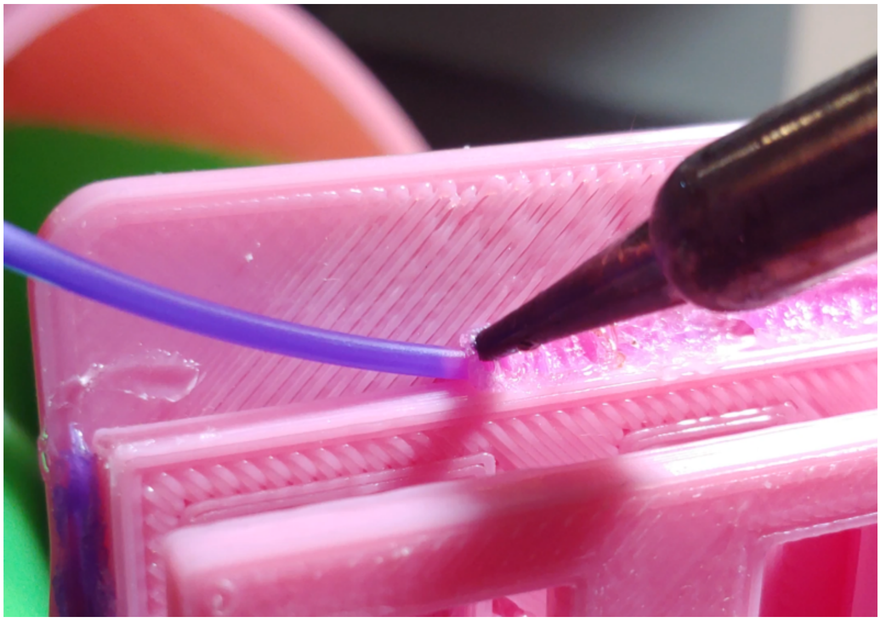


Figure 26: Welding Technique to Attach Float Structural Components

The last aspect of the float manufacturing that presented some obstacles was the suspension system which connects the float to the pontoons. The first thing the team discovered was that two pieces of PLA that rotate around one another are not a very low friction system. So, after the first iteration of the hull, small 3mm (inner diameter) bearing were purchased to put inside of the float suspension to help the pontoons rotate up and down. This drastically reduced the friction and allowed the pontoons to rotate much more smoothly.

4.1.1 Float Manufacturing Challenges

The biggest challenge with float manufacturing was waterproofing the float. There was an inherent tension in this part of the project since, for a connection to be waterproof, there must be a perfect seal and very small tolerances. The second version of the printed float was still very far from perfect with very large variances. An example of this is: the latches did not have specific locations for where they should be mounted, the group 'eyeballed' their placement to get a tight fit. Also, the SolidWorks model did not account for the added rubber gasket so when the lid was printed it immediately did not fit correctly. When adding silicone chalking, if too much was added in one spot this would create a new leak spot further along the interface. Additionally, more than only two latches should have been used. The latches used were towards the back of the lid since this was the flattest mounting location. This meant the interface at the front of the float didn't have as much force pulling down on it, making the front part more susceptible to leaks. Even though the latches were attempted to be placed on flat surfaces, the group found that sometimes the latches would come undone and other times wouldn't latch at all. If the lid was perfectly placed on to begin with, then the latches would remain closed.

4.2 Drone Mounted Plate Manufacturing

The entire drone mounted plate and its attachments were manufactured using a 3D printer. The plate itself had holes in the print for bolts to pass through, but since the Jetson and the GNSS both had pins on the bottom, additional spacers were also made to provide proper clearance. Since there was no CAD model of the servo the holes were drilled after finding a suitable location unlike the rest. The rest of the components were extremely light and therefore did not need anything stronger than double sided adhesive.

4.2.1 Drone Mounted Plate Manufacturing Challenges

The main challenge with the plate was simply getting it printed successfully. There were multiple failed prints since the plate was very large and flat. The other challenge was getting the hole sizes correct, since when printing the material expands and makes smaller holes. Fixing this was simple since all that was needed was a drill with the correct sized bit.

4.3 Power System Manufacturing

4.3.1 On Board Power System

The on board power system utilized a 3 cell lithium polymer battery (LiPo) with a capacity of 3000 mAh. Due to the dangers of using a LiPo, we installed a fast-blow fuse rated at 5A to ensure that if any short-circuits caused trouble within the power system, the on-board electronics would not be damaged. In addition to the fuse, the on board power system used two DZS Elec LM2596 DC-DC buck converters to ensure that the proper voltage was supplied to all electronics. One buck converter provided a 5V output which supplied power to the Jetson Nano and the servo driver. The second buck converter was tuned to provide the optimal output to the motor driver. Through testing, we settled on tuning the second buck converter to provide a precise 12V output. Because both the ZED 2 stereo camera and the Ublox receiver both operated using USB ports, the Jetson Nano provided the necessary 5V to power these instruments.

4.3.2 Deployable Unit Power System

The deployable unit power system utilized a 3 cell LiPo with a capacity of 3000 mAh. We installed a 3A fast-blow fuse to mitigate any damages that could be caused by unintentional short-circuits. Because our 3 cell LiPo operated under 12V, and the operating voltage of the Arduino Uno was 7-12V, it was directly powered by the battery. To handle power distribution to the sensors installed on the deployable unit, we installed a proto shield on the Arduino Uno, which allowed all electronics to operate off of the Arduino's 5V and 3.3V power supplies. The ping sonar, the microSD card shield, and the water sensor all utilized the 5V provided by the Arduino. The IMU used the 3.3V also provided by the Arduino.

4.3.3 Power System Manufacturing Challenges

Due to the compact and low-weight nature of RiBBIT, the biggest challenge for the power system was cable management. For both the on board and deployable unit power systems, there was little space to run all necessary power cables, alongside other cables for data collection. In order to mitigate this problem, we had to splice and re-solder many connections to ensure they would fit in limited space provided. On top of this, we used adhesive wire mounts and mesh casings to harness cables together which allowed the many different wires to have a clean layout.

4.4 Software Manufacturing

The philosophy used to create the final functional float package for the deployed sonar float followed an iterative process. Given that the purpose of this program was to control and connect all of the sensors supporting electronics (water sensor and SD card shield). Smaller testing scripts were produced that would only be responsible for having the Arduino interface with one of the sensors/ supporting components. This allowed for the components to be tested as well as identify what supporting packages and code that would have to be included for the desired functionality. This process was then repeated for all components. Following this, the challenge became integration all of the previously written software that was only responsible for one aspect of the sensor float into one. This again followed an iterative process starting with the SONAR. Following this, the supporting code and libraries for the SD were included and tested to ensure that the Arduino was able to correctly wire data to the SD card. Once this was established the previously developed code for the IMU was included along with the necessary libraries and testing the current build. This process was then repeated once more for the water sensor and status LEDs with the final build undergoing more extensive testing to verify its desired functionality. Some of the major challenges that were faced during this process included missing documentation on the specific pin layout of our Arduino which resulted in issues arising with certain sensor components not functioning as desired. This in conjunction with such issues not being communicated by the Arduino made certain debugging processes rather time-consuming.

The connection between the Pixhawk and the Jetson utilized the Dronekit Python library. This library makes connecting to the Pixhawk, as well as viewing the RC channel values that the Pixhawk receives, very simple. The program with the Dronekit connection code was first tested using a software in the loop system that simulates the connections and communication channels. After making sure the code works the program was moved to the Jetson and the testing continued. Once the Jetson was capable of connecting to the Pixhawk the software for the motor, servo, and camera were integrated. Each of the programs start when a specific RC channel changes goes over a given threshold. These RC channels are manipulated by three different methods: a dial, a switch, and a button. The dial was used for the motor by providing a range of values for the motor to rotate to. The switch was used to start the camera software. Finally the button was used to start the emergency release software. The biggest problem that was faced with this software was the inconsistency in a successful connection. This was solved by simply adding a try-catch loop that would reattempt the connection until successful.

One of the main programs developed in-house for post-processing was the pre-processing script to validate the image quality before running the images through RIVeR for velocity and discharge computations. These functions to check particle density and diameter were written in MATLAB and called in a main image quality verification script. Additionally, code was written for sonar data cleaning, IMU angular offset corrections and fusing IMU and depth data together to reconstruct the cross sectional profiles.

4.4.1 Software Manufacturing Challenges

One of the main challenges to software development was adequate documentation. Given these programs would ideally be passed off to the customer for personal use, it's important to ensure the inputs and outputs of the software application as well as general order of operations be documented well. Without this, the usability of our system and subsequent data processing can become challenging if members of the team are not personally performing the tasks.

Similarly, commenting in the developed code what was adapted from existing code found from external sources would improve clarity. Given the code was passed between team members for utilization and further

4.5 Component Integration

Both the float and drone mount were manufactured and integrated in parallel. Once individually integrated, the full system was integrated together, which began by attaching the drone mount to the underside of the drone, and then by attaching the float to the deployment mechanism. This section describes the process for each for each of these stages.

4.5.1 Float Integration

The float subsystem involves the float itself, Arduino Uno, Sonar, battery, water sensor, IMU, SD card reader, and indicator LEDs. To begin integration, the electrical connections between all the electronics components were established on a bread board to make sure that they were the proper connections. Once the connections were confirmed to be correct and working, the Arduino, IMU, and SD card reader were assembled on a shelving system that was designed to stack the float's electronic components to fit inside of the water-proof float housing. This shelving was welded to the removable lid of the float. The water sensor was connected to longer wires as it is located on the outside of the float. Setting up all the connections before soldering anything in place allowed the team to determine how much wiring was required to make the proper connections for easy wire management. Finally, the appropriate wires were soldered to an Arduino shield to finalize the connections. Afterward, the components were powered on using the battery and one of the LEDs was used to indicate all components had correctly initialised and the other indicated when data collection had began.

The biggest challenged faced related to the integration of the float to the drone. One can noticed that in Figure 25 the pontoons are not a part of the final build version. This was for a few different reasons relating to both manufacturing challenges and things discovered in testing. For the aspects due to manufacturing, looking at 25a, the motor which hosted the float itself can be seen in the middle of the picture. The string ran from the motor and horizontally outwards to the edge of the platform, through a loop, and then down to float. This was to help prevent the float from spinning as much from the downwash of the drone. However, running the string through these small loops on the side of the platform added significant amounts of friction and thus force required to raise the float. To the point which the motor was no longer able to rise the float. Therefore, we had to make significant reductions in weight to allow the float to be raised. Our easiest and only option that didn't require significant redesign was to take the pontoons off. This created the new problem of the the float not being stable and rolling over in the water. In technical terms, the float with no pontoons had a negative metacenter height. To compensate for this, polyethylene foam (the material pool noodles are made of) was added to the main hull of the float to give it the additional buoyancy it needed to become stable. This was accomplished by applying a combination of hot glue and epoxy to attach the foam to the float. This is unfortunately only a temporary solution, the foam pieces are not symmetric and when exposed to the river create asymmetrical forces on the float resulting in the float not facing directly into the flow.

4.5.2 Drone Mounted System Integration

Prior to integrating the physical elements of the drone-fixed suite, each of the electronic components were integrated to the main computer, the Jetson Nano to ensure proper interfacing and simultaneously aid in software development. First, the Jetson was individually integrated with the Zed 2 stereo camera and U-Bloxs GNSS receiver by plugging in via USB and running python scripts to initiate data capture. Then, the Jetson was integrated with the motor driver and stepper motor using an external power source and activated via python script. Similar to the motor, the Jetson was integrated with the servo driver and servo using an external power source and activated via python script. Once each component was individually integrated with the Jetson, the code written to active each process was organized and written to a main driver script. The integration of these scripts are outlined in the software integration section.

The next step was in the process is to integrate the power system with the drone-fixed components. RiBBIT designed a power distribution board which directs power from the battery to the on-board computer, the motor, and the servo. It should be noted that the GNSS receiver and stereo camera are powered through the Jetson. In integrating the power distribution board with the respective components, the process was first done on a bread board to verify that each component was getting the correct voltage. Once the final configuration was determined, the components and wiring on the power distribution board were soldered together and the connections were tested again for verification. Once the electronics components were correctly connected to power, the pin connections from the servo driver and motor driver were plugged into the Jetson and functional testing was done to ensure the correct pin connections.

Once the electronic connections were verified, the components were installed onto the drone mounting plate. This part is a thin plate 3D printed out of PLA to fit beneath the drone's flight battery and holds all on-board electronics and the deployment connection to the float. This plate includes holes to secure the Jetson and servo using screws. Additionally, the plate has a holder for the motor to rest in. Once this part was printed, the electronic components were laid out and secured using a combination of 3M adhesive and screws. The placing of the components was chosen after analysis to ensure even weight distribution. Once the components were secured, the wires were managed using an adhesive wire mount which is used to gather wires and secures them to a flat surface using a zip-tie. After this integration, the electronics were powered on and tested for functionality to indicate successful integration. The integration between the drone mounting plate and the drone is described in the section below.

4.5.3 Drone Integration

For commanding the deployment system and starting the data collection of the ZED 2 camera RiBBIT required a connection from the flight controller to the Jetson. The connection was three wires a Tx, Rx, and GND that come out of the telem2 port on the Pixhawk and connect to the Jetson on its Rx, Tx, and GND pins. This allows for commanding the motor by turning a dial on the controller, starting the camera data capture by flipping a switch, and starting the emergency release mechanism by pressing a button.

5 Verification and Validation

5.1 Depth Profile Testing and Validation

5.1.1 SONAR Accuracy Testing and Validation

The ping SONAR was tested at the Clair Small pool at the CU rec center for the purpose of verifying its performance. During this test, the group would collect data at 4, 6, 8, and 15-foot depths. This data was then compared to truth data which was taken using a tape measure. The pool was not deep enough to accommodate some of these depths, so instead measurements were taken across the pool (parallel to the sides) toward one of the ends. To do so, a simple test rig was constructed which held the SONAR perpendicular to the surface it was measuring against. Approximately 20 depth data points were taken at every distance with the collected data being stored to the SD card associated with the arduino. The data was imported into Matlab and the percent error between the SONAR data and truth measurements calculated. The percent error was calculated to be 0.63%. This is within the SONAR depth accuracy requirement of 1% (DR4.1.2). The difference between the truth and SONAR data is shown in the plot below.

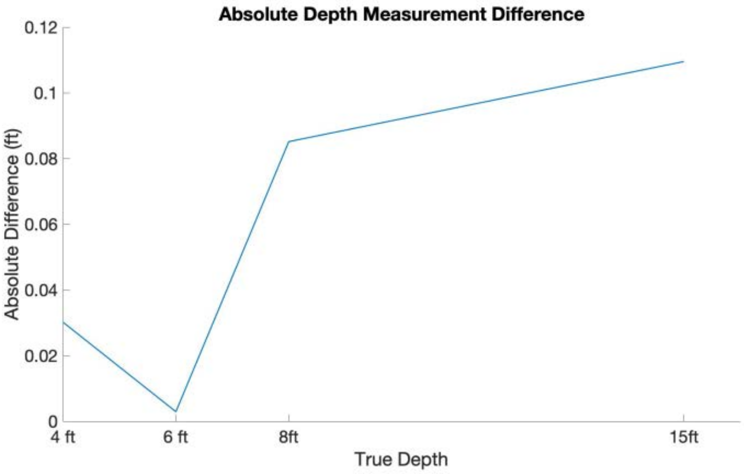


Figure 27: SONAR Depth Accuracy Test

This plot highlights the nature of the inherent error associated with the SONAR measurements. Since the error is a percentage of the total depth, as the depth increases, so does the total difference between the true depth and the measured depth as shown by the general increasing amount of deviation in the plot above.

5.1.2 IMU Accuracy Testing and Validation

To verify the performance and accuracy of the IMU a test stand was used which allowed the IMU to rotate by a known angle. Specifically, angular displacements of 5, 10, 15 and 20 were tested. For this test, the IMU would be rotated to the first angular displacement and held there for approximately 20 seconds while the arduino collects the IMU data and saves it to the SD card. Once the 20 seconds have passed the IMU would then be rotated to the next position. This process was repeated for each angle. After the test was completed the data was imported into Matlab and plotted along with the truth angles and associated error bars. Figure 28 shows the results of this test.

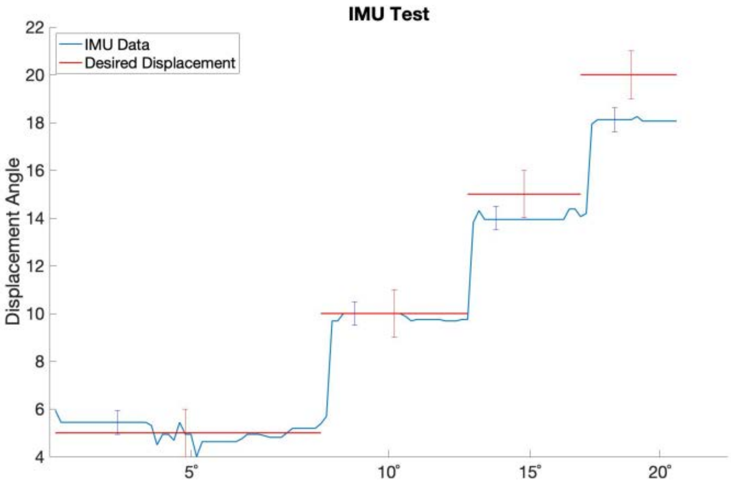
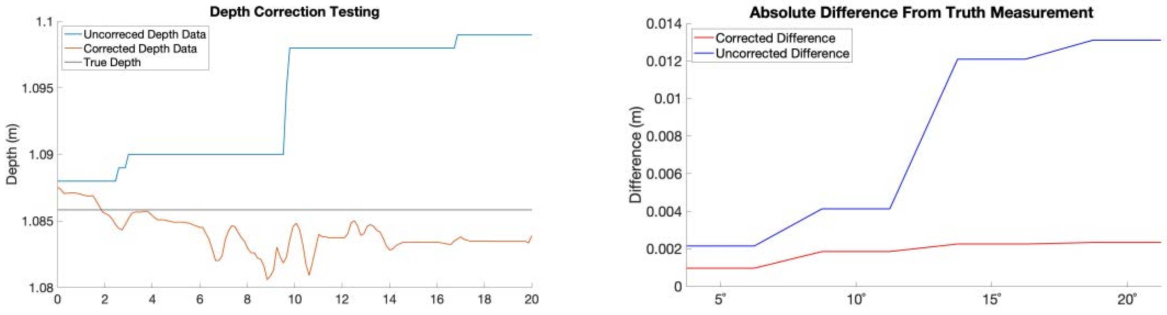


Figure 28: IMU Accuracy Test

Figure 28 shows the raw IMU data plotted in blue and the truth angles shown in red. The red error bars represent the ±1 accuracy requirement that was set for the IMU (DR 4.6.2). The IMU was able to produce accurate data sufficient for meeting requirements for three out of the four displacements tested. The only data point that was not within ±1 degrees was that of the largest displacement angle of 20 degrees. Specifically, this measurement was 0.8 degrees outside of the defined range. A ±0.5 degree error bar was applied to the raw IMU data to account for any errors introduced by either human error while testing or minor construction faults in the test stand. Overall, the test was deemed successful and the accuracy of the IMU was determined to be sufficient for project applications.

5.1.3 Depth Correction Testing

One of the most important tests that were conducted involved the angular displacement data from the IMU and the SONAR data for the purposed of obtaining a corrected depth profile. Any deviation from the gravity vector would create an erroneous depth measurement given that the data would no longer correspond to the depth measurement straight down but rather along the hypotenuse of a right triangle. Similar to the SONAR test, the Clair Small pool and the angular displacement test stand were utilized to conduct this test. The SONAR and IMU were rigidly mounted together such that any angular displacement that the SONAR experienced would then be measured by the IMU and saved through the arduino onto the SD card. Similar to the IMU test, angular displacements of 5,10,15 and 20 degrees were used and data was collected for approximately 20 seconds at each station. This data was imported into Matlab and the previously explained correction technique was used to calculate the corrected depth profile. The comparison between the raw and corrected data as well as the comparison between the absolute difference from the true depth is shown in the Figures 29a and 29.



(a) Depth Correction Test

(b) Absolute difference

Figure 29: Depth Correction Testing Results

The left plot confirms that adding the angular correction reduces the amount of error present in the measurement. Additionally integrating the angular correction provided, on average, a far flatter depth profile. A flatter profile was expected given the shape of the pool. The benefit of accounting for angular displacement is shown clearly in Figure 29. The difference between the true measurement and raw depth data as well as the difference between the truth and corrected data is shown in this plot. In terms of error reduction, applying the correction reduced the absolute percentage error at 5 degrees by 50%. The error reduction increased further to 84% at a 20 degree displacement. These results indicate that the depth correction software functions correctly.

5.1.4 Depth Profile Validation

For the purpose of validating depth data collected by the SONAR instrument, truth data was collected by physically measuring the depth of the St. Vrain river in Longmont, Colorado. Measurements were collected using a yardstick, recording the river depth at 10-inch intervals. Once this process was completed the SONAR float was prepared by initializing the sensors. Then, the float was placed into the flow at one side of the river cross section. From there it was directed across the river cross-section while collecting both depth and angular displacement data. Data collection continued until the SONAR float reached the other side of the river. The data was imported into Matlab and run through the SONAR data parsing and cleaning software procedures. The angular correction function was used to develop the corrected depth profile. A minor speed of sound correction was applied to the data to account for the temperature difference between the fixed speed of sound that the SONAR is initialized to, compared to the one present in the river. The final corrected profile along with the uncorrected data and truth data are shown in Figure 30.

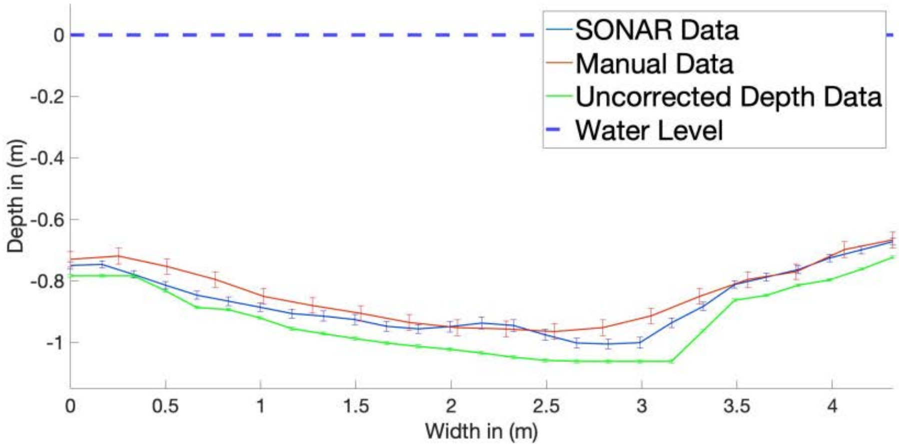


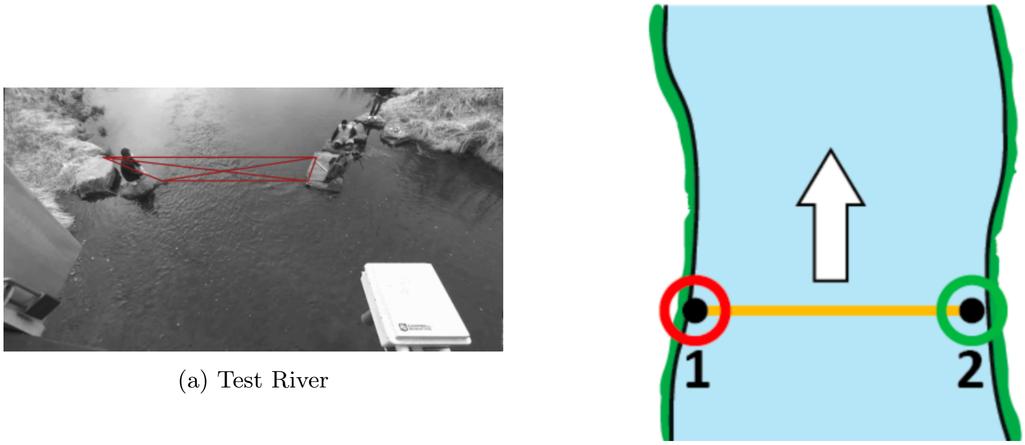
Figure 30: DITL River Depth Data

Figure 30 verifies that the system was able to produce useful and accurate depth data. In comparison to the uncorrected data the corrected data displayed significantly less error.The average difference between the truth and corrected data was 3.5% (outside of the 1% requirement (DR4.1.2)). The uncorrected data showed a mean depth error of 10.4% while as previously stated, the corrected profile had an error of 3.5%. This is approximately a 3 times reduction in error. When it comes to the cross-sectional area of both the truth and SONAR data, the SONAR data yields an area of $3.77m^2$ while the truth data provides a cross-sectional area of $3.7m^2$, a difference of 3.1%. This test in conjunction with the previous pool testing verifies requirement (DR4.1.1) pertaining to the SONAR's measurement range (0.5m - 3m) as well as (DR 7.2) pertaining to the development of a corrected bathymetric profile of the river's cross section.

5.2 Stereo Camera Velocity Post-Processing Testing and Validation

Field testing at St. Vrain Creek in Longmont, Colorado was performed to test the post-processing capabilities of PIVLab and RIVeR. The critical components for velocity post-processing are a top-down recording of the river and measured ground control points. Control points are used in the post-processing software to relate distances in pixels to distances in the real world. Two ways of collecting control points include measuring the physical distance between four points at the testing site, or extracting 3D point cloud data from the stereo camera. Both methods were employed for testing.

Below, a photo of the set up is presented. The left shows an ideal test set-up over the St. Vrain Creek. On the right a visual of the cross section selection step in RIVeR is shown.



(a) Test River

(b) Cross Section Selection

Figure 31: River Test Set Up

Once all video and ground measurement data has been collected, the post-processing procedure is as follows:

1. Load video data from the Zed 2 stereo camera to a Windows machine running the RIVeR software application.
2. Extract images from the video file. Step number can be chosen.
3. If images are distorted or shaken (as they likely will be from a UAV), run the images through the unshake algorithm in RiVER. A region of interest in one image is selected and the following images are matched up. This will create a folder of undistorted images ready for velocity image processing.
4. Load corrected images into PIVLab.
5. Set a region of interest and a mask of area around the river.
6. If desired, the images can be preprocessed though enhancing the contrast.

7. Select the desired method of image cross-correlation. There are two algorithms to choose from; DCC (single pass direct cross correlation) or FFT window deformation (direct Fourier transform correlation with multiple passes and deforming windows). The size of the interrogation windows will also be specified.

8. Begin analysis. PIVLab will then run through the selected PIV analysis method, computing the velocity vectors between every set of image frames. The median flow vectors will now be computed.

9. Data validation. In this optional step, erroneous vectors that show up due to poorly illuminated regions or lack of seeding pattern at the water surface can be removed. Velocity limits can be set to reject vector outliers.

10. Load PIVLab image processing session into RIVeR.

11. Load a background image as well as a control points (CPs) image. This image must have all CPs visible.

12. Define the control points and region of interest. If camera is nadir pointed then 2 or 4 points and the measured distances can be input. Four points are necessary if the camera lens has an oblique view. Alternatively, a 3D point cloud can be input, with at least 10 CPs surveyed in the region horizontally and vertically. The stereo camera has the capability to collect depth point clouds. A script was developed for the purpose of obtaining these 3D control points.

13. Background image rectification. Once CPs and ROI are defined, a map of the real world pixel diagonal size will be displayed.

14. Define cross-section. This will be selected based upon where the sonar data was collected. At this stage, the perpendicular velocity vectors along the selected cross section can be visualized.

15. Discharge estimation. Flow discharge can now be computed for the selected cross section. A CSV file of station distances along the cross section and the associated depths will be input. The summary of the discharge calculation can then be saved. This presents all station velocity, depth, and discharge information as well as mean velocity and total discharge.

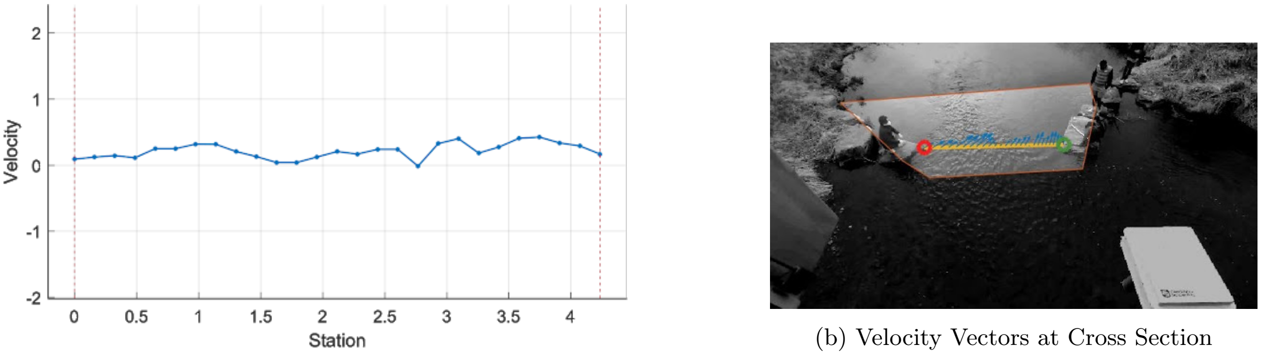
The data collected from this velocity testing procedure includes a median velocity flow field, velocity vectors perpendicular to the desired cross section, and mean cross-sectional velocity. A cross-section at the river site was identified in the software to further estimate the river discharge.

Below, the results from the St. Vrain day in the life test are tabulated. The gauge velocity measured by Colorado Department of Natural Resources at the same river site is also provided. The Colorado Department of Resources measures river discharge at 15 minute intervals throughout the day. These measurements are determined via extrapolation over time. Due to the fact that RiBBIT's measurement is a direct one, it is reasonable to consider that the new developed system may be more accurate than that of the Colorado Department of Natural Resources' measurement.

Mean Velocity Measured at Cross-Section	0.21 ± 0.03 m/s
Colorado Department of Natural Resources	0.34 m/s

Table 5: Boulder Creek Velocity Data

The following images were generated in PIVlab during post-processing of the stereo camera data. The plot shows the surface velocity measured at each station used. The second image shows the velocity vectors calculated at the selected cross section.

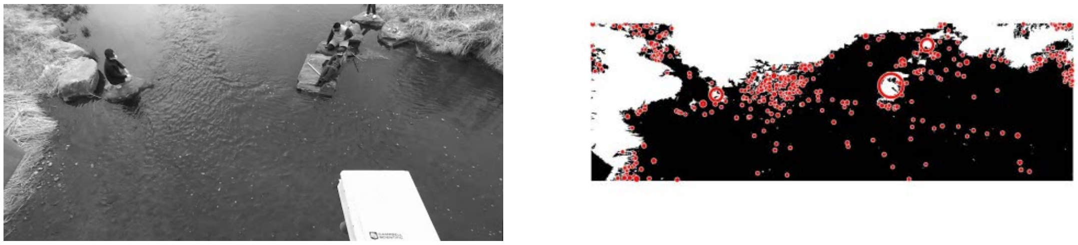


(a) Velocity vs. Station

(b) Velocity Vectors at Cross Section

Figure 32: Velocity Measurements from PIVlab

5.2.1 Stereo Camera Image Quality Verification



(a) St. Vrain River

(b) Cross Section Particles

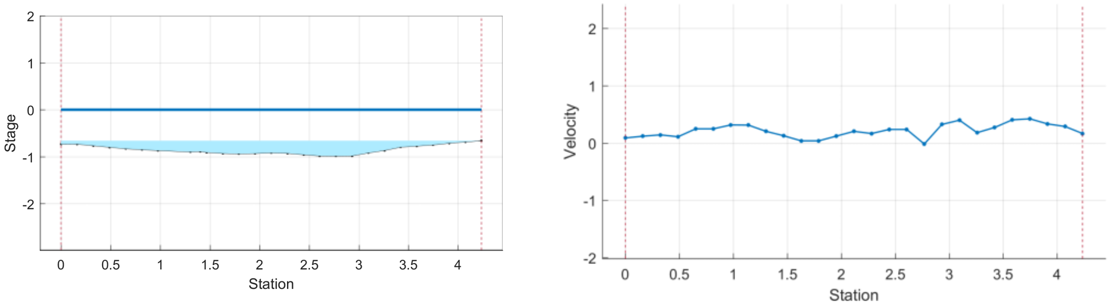
Figure 33: Image Quality verification Process

Prior to running the images through PIVlab, they are passed through a MATLAB developed program to check that the image quality meets optimal conditions as defined by the particle image velocimetry (PIV) software. PIVLab velocity accuracy is dependent on adequate particle image diameter, particle density, sensor noise, motion blur, and particle pair loss. If the density of particles is between $1\% - 8\%$ and the particle diameter is between 1-4 pixels, the RMS error will be less than 0.1 pixels. The developed MATLAB program *ImageAnalysisMain.m* enables the user to select the river image, select the desired cross section, and runs it through two functions, *findDensity.m* and *findParticleSize.m.*

For the test case at the St. Vrain river in Longmont (Figure 34), the particle density was found to be 2.89% while the average particle diameter was 1.11 pixels. Given that the camera was situated about 10 meters above the river, this correlates to a worst case error of 0.0299 [m/s]. This process should be repeated for all new site locations to ensure the particle diameter and density requirements are met to compute reliable surface velocity measurements. If the river conditions do not meet the optimal conditions as defined above, it is suggested to seed the river with leaves to ensure enough objects are present for tracking. Therefore, it is fair to state that FR5: *The instrument suite shall be capable of measuring the surface velocity of a river cross section* and DR7.1: *The stereo camera data shall be post-processed to calculate river surface velocity.* were validated. This also contributes to FR6 which relates to powering and commanding the instruments and sensors.

5.3 River Discharge Validation

Once the river images had been run through the image quality verification program, they were then run through the RIVeR software to compute the mean surface velocity and total cross-sectional discharge. PIVlab is built into RIVeR and first compares the image sequences, identifying the particles and computing their velocities. Next, RIVeR enables the user to select the desired region of interest and cross section. The depth data from the stereo camera or manual measurements are also input to adjust the images from pixels to real-world distances. Then, the user inputs the corrected sonar depth data in csv format. Next the program computes the discharge at each station as well as the total cross-sectional discharge. Below, the station depths and velocities from the St. Vrain river test are shown. These data sets are utilized to compute the river discharge.



(a) Cross-Sectional Station Depths [m]

(b) Cross-Sectional Station Velocities [m/s]

Figure 34: St. Vrain Discharge Calculation Process

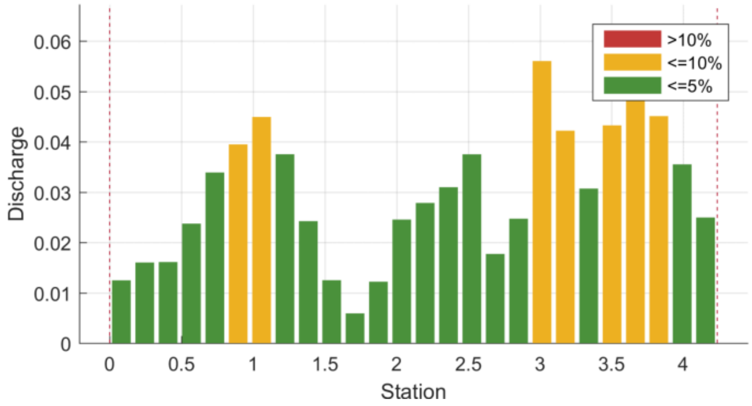


Figure 35: St. Vrain Discharge Calculations $[m^3/s]$

The system-collected data run through RIVeR was computed to be $0.8 \pm 0.1[\frac{m^3}{s}]$. The error in this measurement comes from the average surface velocity multiplied by the cross sectional area. The actual uncertainty is likely much smaller due to the models used in the RIVeR software application. The system-collected data was compared to that computed by the Colorado Department of Natural Resources (CDNR) at the same river site. At the same location and time, CDNR measured a discharge of $0.85 \pm 0.07[\frac{m^3}{s}]$. This validates FR7: *The collected data shall be post-processed to calculate river discharge*, and at a larger scale, also validated the entire mission and overall system itself.

5.4 Sonar Float Model Validation

The float needed to demonstrate weather-vane and roll stability per DR4.2.3. However, due to the complex nature of hydrodynamics (see final hull design in section 3 for more details on this) the development of boat models and their validations was a less analytically rigorous process. Additionally, dynamic modeling of the system was difficult as there was a learning curve associated with using SolidWorks to produce analysis worthy models to perform validation tests against. Instead, a high level validation of these qualities took place using an observational approach. For example, the SolidWorks model produced an estimate of the location of the float's center of gravity when all the individual components in the float were added to the model. A rope was attached around this location to see if, when suspended from the rope, the float would remain level. The float was found to be fairly level, only tipping backwards slightly.

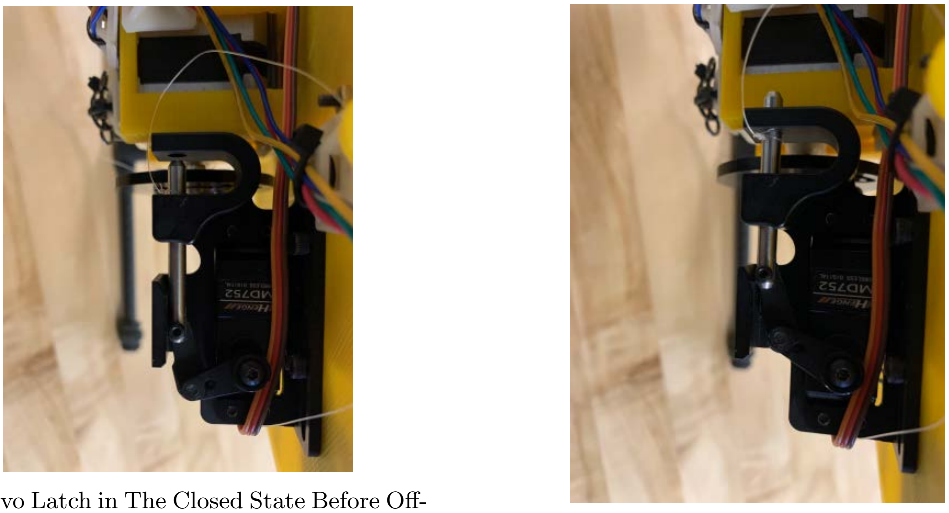
A model that determines the center of pressure of the float could be used to calculate where the rope attachment points would need to be to achieve weather-vane stability. However, such a model would involve significant simplifying assumptions, ones that would become less accurate when the float was in a moving flow. Alternatively, ropes were attached to the float at different points and tested in a body of water to determine which connections gave the float weather-vane stability. Weather-vane stability was achieved when the rope was mounted towards the front of the float; however, the rope was moved closer to the center so that it would remain mostly level during flight. To adjust for this the pontoons were moved backwards consequently also moving the center of pressure backward. This allowed the float to regain the floats weather-vane stability. However, moving the pontoons backward changed the mounting locations to in front of the pontoon's CG. This caused the pontoons to be pushed underwater when exposed to a moving flow. Because of this and the problems mentioned in the Manufacturing Float Integration Section, the pontoons were removed all together. Removing the pontoons pushed the center of pressure forward, which caused the float to loose weather-vane stability. Thus, the final corrective measure, which brought the float to its configuration seen in Figure 25, was to move the attachment points further foreword again. To prevent the float from tipping too far backwards, a T shaped rigging was added that kept the float in a relatively level arrangement while in flight.

Roll stability introduced similar issues as weather-vane stability in terms of the complexity of developing models and was similarly easy to validate by testing the system in the water. To validate whether the float had roll stability, it was placed in water and disturbed. The float's stability was analyzed by observing its reaction to the disturbance. The float displayed roll stability when the pontoons were attached, but once the pontoons were removed the float was unstable. Polyethylene foam was attached to the sides of the float, shown in Figure 25, and the float was placed back in a large sink and disturbed once again to analyze its roll stability. This was significantly less stable than with the pontoons but it could survive relatively significant disturbances while still returning to an upright position. The stability of the boat directly impacts the quality of the sonar data collected, thus these designs aided in meeting Functional Requirement 4, enabling an accurate bathymetric profile to be collected.

5.5 Deployment Mechanism Testing

Throughout the assembly and testing of the deployment mechanism the mechanisms sub-team was able to identify several issues and make multiple changes in the overall design accordingly. In the early stages of testing, the stepper motor was tested lifting a 1kg payload at the range of voltages from 6 to 12V, which is the maximum operating voltage for the bipolar stepper motor. The stepper motor was able to lift, hold and deploy the payload whenever the voltage running through it was at minimum 8V. Any voltage below 8V made the stepper motor either stutter while lifting the payload or completely fail resulting in the load being dropped. While the stepper motor was hooked to the power supply, it was noted that the stepper motor gets extremely hot, especially when it was running at its maximum operating voltage. Because of that, the mechanisms sub-team had then thought about adding a heat-sink to the back of the stepper motor. However, it later came to the attention that this was a common behavior for stepper motors and that their safe operating temperatures can reach up to 100°C. The tests were done using a Jetson Nano and a motor driver in order to deliver the high and required operating current for the stepper motor

On the other hand, the servo was also tested separately with a Jetson Nano and a PWM servo driver to deliver the correct PWM signal specifications to the servo. One issue discovered in the servo, even before testing, was that it had a manufacturing disadvantage where the latch does not fully close and a small gap can is left which the fishing line can easily fall through. This issue was simply resolved by unscrewing the pivoting point of the servo and off-setting it so that it would then fully close.



(a) Servo Latch in The Closed State Before Off-Setting

(b) Servo Latch in The Closed State After Off-Setting

Figure 36: Servo for The Emergency Release

At the stage of assembly, the mounting plate was 3D printed to house the components of the deployment mechanism. A 15.8mm radius spool was purchased and attached to the shaft of the stepper motor to be able to reel the fishing line around it. In addition, to mitigate the rise in the temperature of the stepper motor, a power distribution unit (PDU) was used to supply the stepper motor with the minimum voltage that is just enough to enable it to lift and deploy the float, which was 8V based on the testing conducted previously. The mechanisms sub-team tested the deployment mechanism in this set up and made even further discoveries. When testing, the stepper motor failed to lift the float at all and it was then decided to redo the test at a higher voltage. Surprisingly, the stepper motor still failed to lift the float at any voltage, including 12V, this rose a concern because those test results did not match the results of the tests made in the early stages. After further investigation and assessment of the test results, the issue was understood to be coming from the fact that the very first deployment mechanism tests were done without a spool and that the fishing line was directly attached to the shaft of the stepper motor which increased its torque due to a smaller radius. This initially not being accounted for as well as the friction in the system not being adequately modeled. Thus, the float design was modified and its weight was decreased, along with removing the spool and attaching the fishing line directly to the 2.5mm radius shaft of the stepper motor. This new set up is illustrated in Fig 18b. After testing the deployment mechanism again with the new adjustments, the stepper motor showed success and was able to generate enough torque to lift and deploy the float on the command of the pilot. This test satisfied functional requirements 4.3.1, 4.3.2 and 4.3.4. The functional requirements are explained in more details in the Appendix.

5.6 Drone Verification and Validation

5.6.1 Drone Test Flights

In order to ensure a safe flight with the payload attached the team performed multiple test flights with different configurations. The test flights that were performed were done as follows:

- Flight test 1 - Updated the firmware but in doing so may have changed motor configuration in the software causing a crash
- Flight test 2 - Fixed motor configuration, can fly by hand but need to constantly hold down on the stick
- Flight test 3 - Flew autonomous mission, tested auto landing and return to launch modes
- Flight test 4 - Attached the base plate with the drone battery placed on top. Had issues being stable in position hold mode, tipped over on landing and broke a motor
- Flight test 5 - Flew in the ASPEN Lab with the drone battery below, didn't have position hold mode and flew in altitude hold mode, noticed a tendency to drift, flew with float attached and with some success

managed to winch the float

- Flight test 6 - Flew with the mounting plate and found that the drone still had stability issues in position hold mode and drifts in altitude hold mode, an attempt was made to use autotune but was cancelled due to battery limits
- Flight test 7 - Same test as 6 but initially recalibrated the accelerometer, which fixed the instability issue, tried to perform another autotune but resulted in a crash when the battery suddenly died. Only the landing feet were damaged.
- Flight test 8 - Flew with the deployment mechanism attached, stable flight with position hold mode with the float being deployed as well as undeployed.
- Flight test 9 - Went to St.Vrain for full systems test, drone flew stable but had trouble holding position from the river pulling it. The float really swung around when making any movements with the drone. But relatively easy to control.

The biggest problem faced with integrating the payload was the strange drift when in altitude hold mode, seen in flight test 4, or instability in position hold mode seen in flight test 6. The initial reasoning was that having the battery on top of the drone with the flight controller sitting on top of the battery was causing the drone to be unstable. The effect of this change was not noticed during flight test 5 since the flight was done in altitude hold mode rather than position hold mode. The cause for the drift/instability was noticed during flight test 6 and was corrected in flight test 7 with the recalibration of the accelerometer. The reason the instability was only observed when the payload was attached was likely due to the increase of moment of inertia. The drone might have some sort of limiting factor on the motor power used for level flight so it wouldn't correct for its angle offset. Using autotune would have likely fixed this issue as well since these motor limits would be adjusted.

These flight tests showed that our system satisfied Functional Requirements 1 and 8 along with their Design Requirements, with some additional success accomplished on Functional Requirement 2. If there were more time to work on the project, further testing of the UAV system would be carried out to improve control, commanding, and data collection strategies.

5.6.2 Full System Stability Simulation

In order to make sure that the design of the drone/float system will remain stable in a flowing river, the team designed dynamic simulation of the drone and float system. The design simulates a simple PID control system that uses the current position and attitude of the hexacopter in order to maintain a given position. The simulation was made using simulink to aid in visualization of the data as well as simplify some of the design work. Below is the main loop of the simulation as well as the defined coordinate system that the simulation uses. Starting with the input path, it feeds into the position controller which has PID controllers for the X and Y axes.

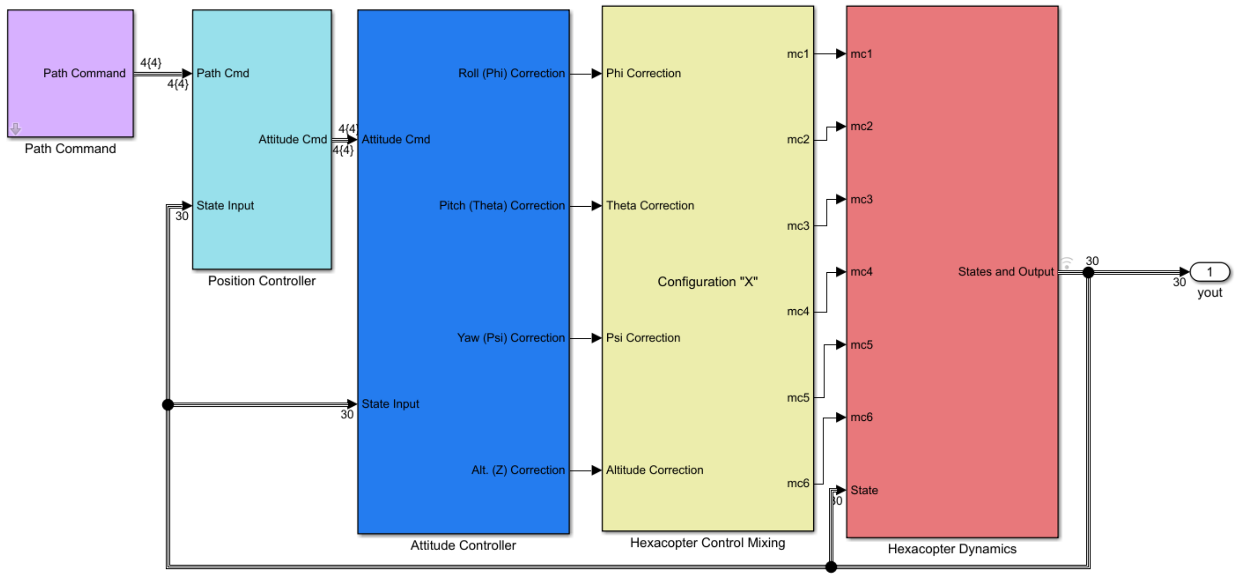


Figure 37: Main Loop

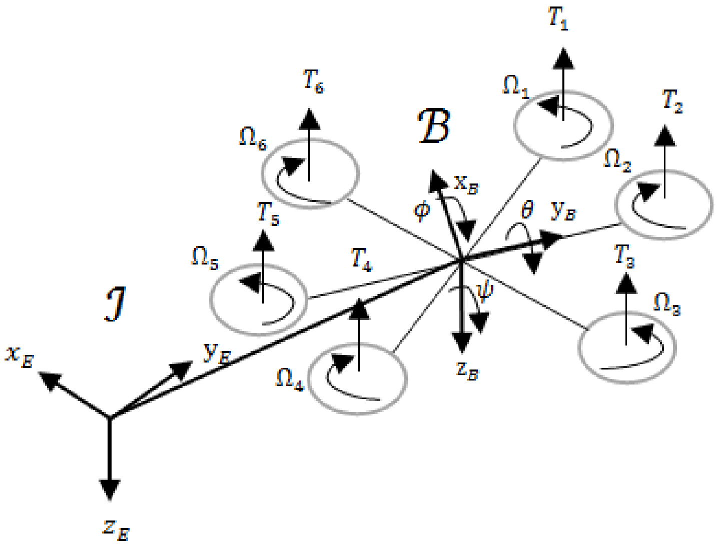
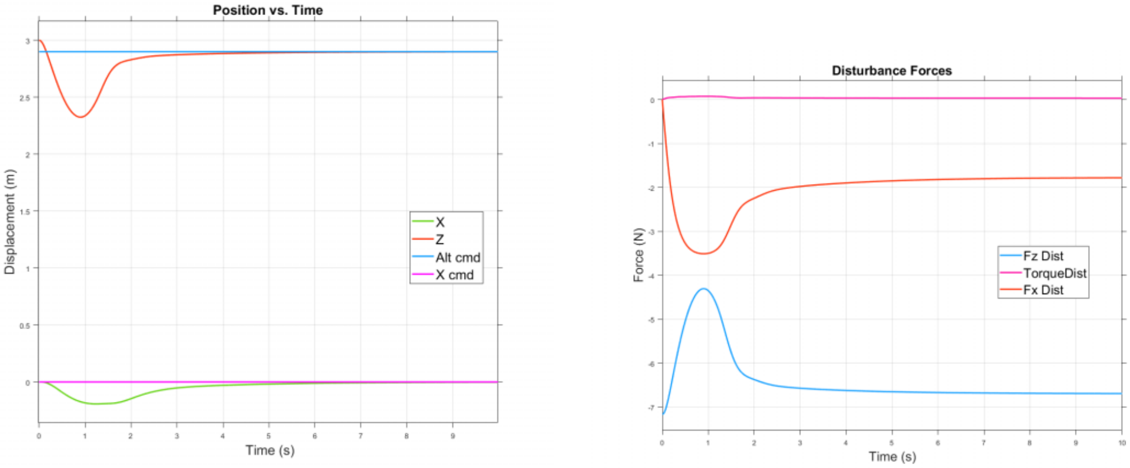


Figure 38: Dynamic Model

This then feeds into the attitude controller which has a PID controller for ψ, θ, ϕ, and the altitude, which is in the Z axis. The Attitude controller gives the required change in attitude and altitude, these commands feed into the hexacopter control mixer which translate them into motor commands. This is done with the equation below. Where the proportions are scaled relative to the motor's effect on the change in attitude rate and altitude.

$$\begin{pmatrix} \mathrm{Mc}_1 \\ \mathrm{Mc}_2 \\ \mathrm{Mc}_3 \\ \mathrm{Mc}_4 \\ \mathrm{Mc}_5 \\ \mathrm{Mc}_6 \end{pmatrix} = \begin{pmatrix} 1 & \text{-}1/2 & \sqrt{3}/2 & 1 \\ 1 & \text{-}1 & 0 & -1 \\ 1 & \text{-}1/2 & \sqrt{3}/2 & 1 \\ 1 & 1/2 & \sqrt{3}/2 & -1 \\ 1 & 1 & 0 & 1 \\ 1 & 1/2 & \sqrt{3}/2 & -1 \end{pmatrix} \times \begin{pmatrix} \mathrm{Alt}_{cmd} \\ \phi_{cmd} \\ \theta_{cmd} \\ \psi_{cmd} \end{pmatrix}$$

Finally these motor commands are passed into the hexacopter dynamics block. In this block the forces and moments from the motors are calculated and combined with any external forces and moments, like the forces from the float. These forces and moments are passed into a builtin 6 degree of freedom dynamics block which simulates the dynamics of the drone. The forces from the float are simulated assuming the float stays at a 3m distance and never has slack, also, the float is assumed to have the maximum drag that it can physically experience without being pulled out of the water. With these worst case scenarios the drone still maintains stability as shown in the plots below.



(a) Displacement

(b) Disturbance Forces

Figure 39: UAV Displacement and Disturbance Forces

The plots show a slight dip in the Z direction as well as a displacement in the x direction, but it recovers fairly quickly. Again, this model uses very simple PID controllers, the actual drone will have a more robust control system that will deal with disturbances even better.

The tests mentioned in the previous section demonstrate the successful implementation of these systems. Because this represents the full integrated system, those previously mentioned tests enable functional requirements 1, 3 and 8 to be met.

5.7 Battery Life Testing

5.7.1 On Board Battery Life Testing

Once all on board electronic components were integrated and tested, the battery life of the on board battery was conducted. This test was conducted by running all on board electronics in a configuration that allowed them to achieve their maximum power consumption. Using the fully charged 3 cell 3000 mAh LiPo, the entire on board avionics unit was operating in its maximum power consumption state for a time of 45 minutes. The success criteria to pass this test were that all electronic components remain powered on for the full 45 minutes and operate at their full functionality. We were able to verify this visually and by inspecting the collected data. For the instruments that don't collect data, the motor and servo, we visually observed that they were on and operating for the entirety of the 45 minute test. To verify that the success criteria was met for the data collecting sensors, we verified that continuous data was collected throughout the whole test. After conducting the test, the success criteria were verified. For redundancy in this test, we checked the final voltage of the battery and concluded that the on board battery passed the test having only used a small fraction of its capacity (20%). As a result of the described criteria, the battery was deemed suitable for the on board power system. This test satisfies design requirement DR6.3, "Both the on-board and deployed sensor units shall include batteries to provide enough power for 30 minutes of operation at maximum power consumption".

5.7.2 Deployable Unit Battery Life Testing

After all deployable unit electronics were integrated and separately tested, the deployable unit battery life testing began. The acceptance criterion of this test was that all deployable unit sensors collected continuous data for 45 minutes. The 3 cell 1000 mAh LiPo dedicated to powering the deployable unit avionics system was charged to 12V. After powering on the entire deployable unit sensor suite, the system was left untouched for 45 minutes. At the end of the 45 minutes, we found that the data collected was continuous and the battery only used a small fraction of its capacity. From this test, we concluded that the deployable unit battery was sufficient and that the test was successful. This satisfies design requirement DR6.3, "Both the on-board and deployed sensor units shall include batteries to provide enough power for 30 minutes of operation at maximum power consumption".

5.8 Positional Determination Validation

5.8.1 Positional Determination and Geo-Referencing Test Plan

The following test plan was developed to validate GNSS hardware and SLAM software. Along with a power supply and miscellaneous cables, these tests require the u-blox GNSS receivers, the IMU, at least one microcontroller, and either the second microcontroller or the Emlid RS2 base station receiver. One u-blox receiver and an IMU will be set up to record their data outputs on and microcontroller, and these components will be bundled together such that they are fixed relative to one another. This hardware setup will act as the GNSS rover. The GNSS base station can either be the other u-blox receiver connected to a microcontroller, or the Emlid RS2 base station. This option will depend upon the ease of borrowing the Emlid unit. Each test will involve setting up the base station in an unobstructed area such as a park or yard and collect a minimum of thirty minutes of data before the rover becomes active, allowing the base station to sufficiently estimate its position. The rover unit will begin collecting data and then travel a specified route. The route should be verifiable in its properties such as length of distinct segments. Suggestions for such test routes include city blocks (which can be measured with applications including Google Maps or Earth), athletic fields or tracks which have distance markings, and any surveyed location with known geometric properties. Ideally, the path should contain several points which are visited more than once, which will allow the testing of GTSAM's loop closure capabilities. After collecting the data and analytically constructing the test trajectory, the raw GNSS data of the rover and base station will be processed to recover the rover's position with respect to time. At this point, the GNSS and IMU data may be fed into GTSAM to test this component of software. The resultant path from GTSAM will be compared to the analytically determined path to verify the accuracy of the entire process.

Several subsystem tests verify the functionality and accuracy of the positional determination requirements, the most significant being the u-blox C94-M8P GNSS receivers and AprilTags.

The u-blox GNSS receivers were first tested to be capable of recording data as standalone components. While they were quickly shown to work in providing generic NMEA positional messages, they were then tested with the raw UBX format. These messages are processed in the u-blox software suite to confirm their validity. Similar to the AprilTags tests described below, the GNSS receivers were held at constant intervals. The Emlid base station was also collecting data during this test to allow relative positioning (PPK) and account for ionosphere corrections. The results, shown in Fig. 40, validate the receiver's accuracy.

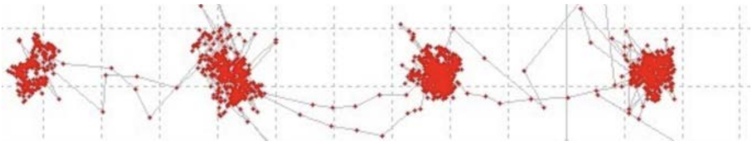


Figure 40: GNSS Receiver Interval Accuracy Test

On a larger scale, the GNSS antenna was attached to the roof of a car to test its performance at greater distances and speeds. Starting from the NCAR parking lot, this test consisted of driving down to the surrounding neighborhood and returning to the same location. The Emlid base station was also collecting data during this test to allow relative positioning (PPK) and account for ionosphere corrections. The result of this post-processed trajectory is shown in Fig. 41. The results of this test, After the success of this larger scale test, we had enough confidence to use the GNSS receiver on the drone payload.

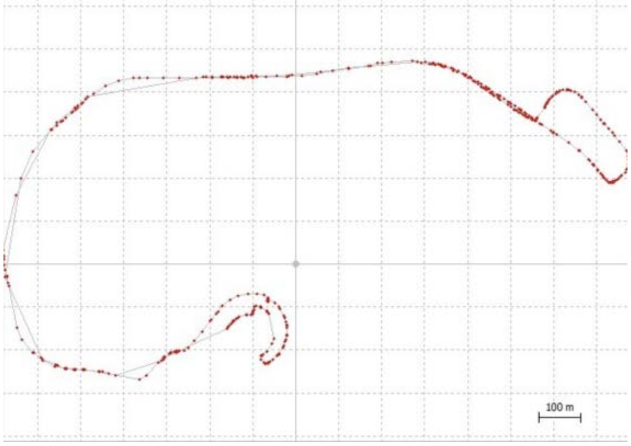


Figure 41: GNSS Receiver Long Range Test

To test the AprilTag software discussed in the Final Design section, a physical tag was first created. After the optical specifications of the ZED2 camera were also obtained, these parameters were inputted to the software package. 10cm intervals, up to 2m, were measured from the camera lens and the AprilTag was held at each of these intervals. After running this video data through the University of Michigan software, the distance and orientation of the AprilTag, relative to the camera, were successfully recovered. At distances smaller than 50cm, the error was less than or equal to 1.2cm; between 50cm and 150cm, the error was less than 2.7cm; and between 150cm and 200cm, the error was less than 3.3cm. An example image that was fed into the software is shown in Fig. 42. Note the image blur, which, to a mild degree, did not effect the post-processing software's functionality.

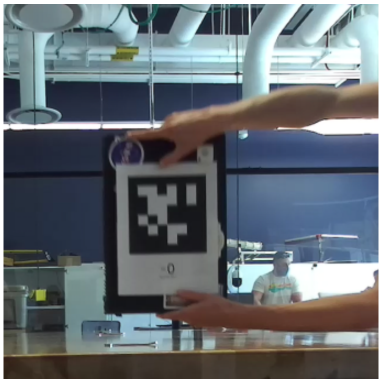


Figure 42: AprilTag Test

All these tests help meet FR5, specifically DR5.2 *"The drone-fixed instrument suite shall include a GNSS receiver."* It also indirectly contributes to FR6 which relates to powering and commanding all instruments and sensors as well as FR7 which relates to post processing the data. However, this testing did not satisfy DR 7.4: *"The GNSS data shall enable post-processed positioning with horizontal accuracy of +/- 4 cm and vertical accuracy of +/- 5 cm in ideal conditions."*

6 Risk Assessment and Mitigation

Prior to system manufacturing, testing, and integration, the main sources of risk to team RiBBIT were identified as follows:

1. Stereo Camera Data Error
2. Sonar Data Collection Errors
3. Environmental Hazards
4. Float Deployment Mechanism Failure
5. Test Schedule Slips

Below, the pre-mitigation matrix is presented depicting where the team has characterized the five main project risks on a scale of likelihood and severity.

		Severity				
		Negligible	Minor	Moderate	Major	Severe
Likelihood	Ext. High	🟨	🟧	🟥	🟥	🟥
	High	🟨	🟨	🟧 5	🟥 1,2	🟥
	Medium	🟩	🟨	🟨	🟧	🟥 3,4
	Low	🟩	🟩	🟨	🟨	🟧
	Extra Low	🟩	🟩	🟩	🟨	🟨

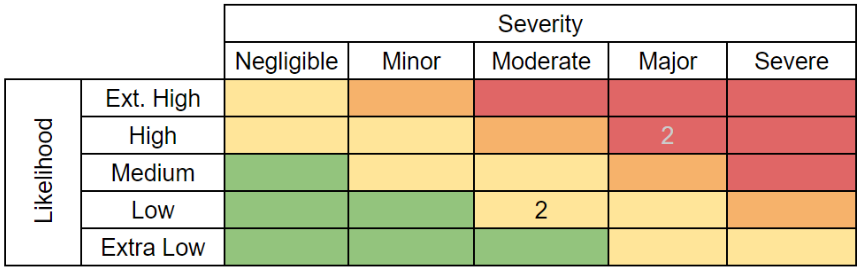
Figure 43: Pre-Mitigation Risk Matrix

The first risk pertains to the concern that the stereo camera would not collect adequate data to enable velocity calculation post-processing. The consequences of this risk occurring would result in river discharge calculations not being able to be computed, due to the dependency on the river velocity. To mitigate this risk, the team tested image post-processing capabilities with varied environmental conditions to ensure that the software could extract velocity information. From initial testing, presented in the verification and validation section of the document, velocity information was able to be extracted, and the main factors that contribute to velocity accuracy were identified. If particle diameter and density in the images meet optimal conditions, as defined in the velocity final design section, the velocity will be able to be calculated. In case the optimal conditions are not met, a back up option of seeding the river for easier object tracking can be carried out. In the off chance that even this method doesn't enable proper velocity calculations, single object tracking can be performed by measuring a specified distance and timing how long the object takes to travel the distance.

		Severity				
		Negligible	Minor	Moderate	Major	Severe
Likelihood	Ext. High	🟨	🟧	🟥	🟥	🟥
	High	🟨	🟨	🟧	🟥 1	🟥
	Medium	🟩	🟨	🟨	🟧	🟥
	Low	🟩	🟩	🟨	🟨 1	🟧
	Extra Low	🟩	🟩	🟩	🟨	🟨

Figure 44: Stereo Camera Data Error Risk Matrix

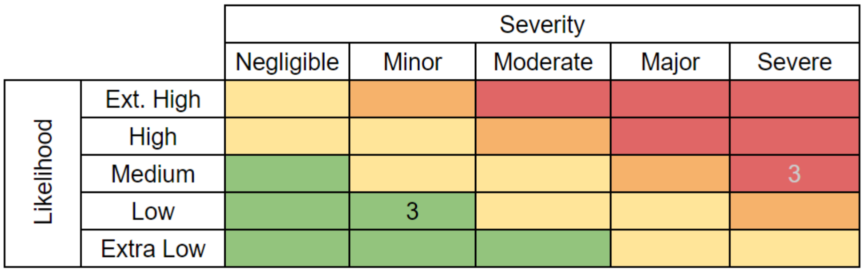
The second risk is if the sonar collection results in erroneous data. With this risk occurring, the river depth profile would not be accurately generated and subsequently an accurate river discharge would not be able to be calculated. To mitigate this risk, a stable boat design was designed around ensuring proper data be collected. Data post-processing angular corrections can be carried out to account for tilts caused by the river conditions.



		Severity				
		Negligible	Minor	Moderate	Major	Severe
Likelihood	Ext. High	🟨	🟧	🟥	🟥	🟥
	High	🟨	🟨	🟧	🟥 2	🟥
	Medium	🟩	🟨	🟨	🟧	🟥
	Low	🟩	🟩	🟨 2	🟨	🟧
	Extra Low	🟩	🟩	🟩	🟨	🟨

Figure 45: Sonar Data Error Risk Matrix

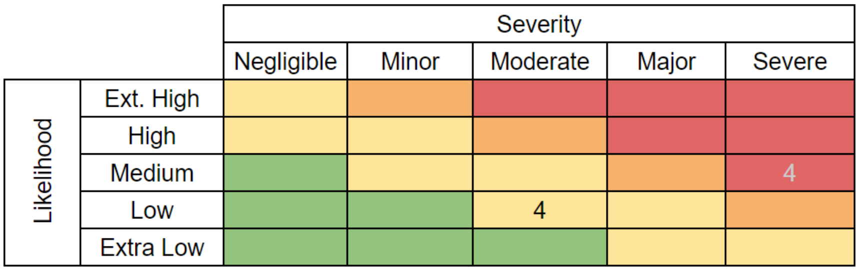
As this product is for field-site surveying, the third risk is environmental hazards. If the environment is not adequate, equipment can be damaged or lost. Additionally, inaccurate data may be collected if a suitable test site is not chosen. To mitigate this risk, a team observer was tasked with finding a safe testing site that doesn't have hazards that may harm anyone or any equipment. A professional drone pilot must fly the drone in line of sight to mitigate flight risks. Additionally, a release mechanism to drop the payload in case of the float getting caught was added to the design.



		Severity				
		Negligible	Minor	Moderate	Major	Severe
Likelihood	Ext. High	🟨	🟧	🟥	🟥	🟥
	High	🟨	🟨	🟧	🟥	🟥
	Medium	🟩	🟨	🟨	🟧	🟥 3
	Low	🟩	🟩 3	🟨	🟨	🟧
	Extra Low	🟩	🟩	🟩	🟨	🟨

Figure 46: Environmental Hazards Risk Matrix

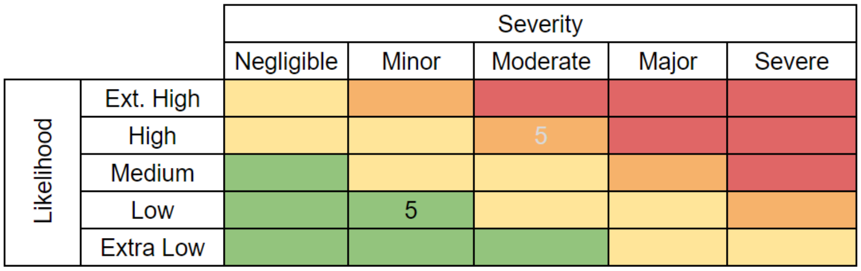
A similar risk is if the float deployment mechanism fails. The consequences to this risk would be no sonar depth data being collected, as the float holding the sonar would not be in contact with the water's surface. To mitigate the risk, the motors and servos were tested in varied conditions. Part of this testing entailed ensuring the power needs of the mechanism were met as well as testing RC commanding. In case of faulty deployment, a back up option of having the float be manually deployed before flight can be performed.



		Severity				
		Negligible	Minor	Moderate	Major	Severe
Likelihood	Ext. High	🟨	🟧	🟥	🟥	🟥
	High	🟨	🟨	🟧	🟥	🟥
	Medium	🟩	🟨	🟨	🟧	🟥 4
	Low	🟩	🟩	🟨 4	🟨	🟧
	Extra Low	🟩	🟩	🟩	🟨	🟨

Figure 47: Floating Deployment Mechanism Risk Matrix

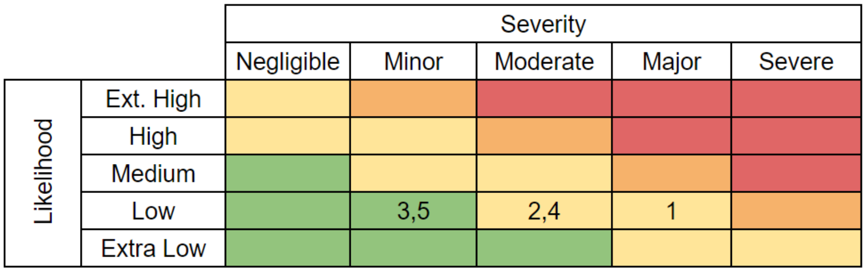
The fifth risk is test schedule slipping. Given the ongoing COVID-19 pandemic, it was expected that facilities and team testing would be impacted. To mitigate this risk that would result in full system integration delays, a robust test plan was put together with lots of margin. Backup days were planned for in case of poor conditions.



		Severity				
		Negligible	Minor	Moderate	Major	Severe
Likelihood	Ext. High	🟨	🟧	🟥	🟥	🟥
	High	🟨	🟨	🟧 5	🟥	🟥
	Medium	🟩	🟨	🟨	🟧	🟥
	Low	🟩	🟩 5	🟨	🟨	🟧
	Extra Low	🟩	🟩	🟩	🟨	🟨

Figure 48: Test Schedule Slip Risk Matrix

After reviewing mitigation strategies to the aforementioned risks, it was determined that the risks fell into the below likelihood-severity rankings.



		Severity				
		Negligible	Minor	Moderate	Major	Severe
Likelihood	Ext. High	🟨	🟧	🟥	🟥	🟥
	High	🟨	🟨	🟧	🟥	🟥
	Medium	🟩	🟨	🟨	🟧	🟥
	Low	🟩	🟩 3,5	🟨 2,4	🟨 1	🟧
	Extra Low	🟩	🟩	🟩	🟨	🟨

Figure 49: Post-Mitigation Risk Matrix

6.1 Risks Identified in Manufacturing, Testing, and System Integration

After completing the second semester of the project, it was found that the most prominent risks that came up were the float deployment mechanism as well as test schedule slips. Given the nature of the pandemic, in-person meetings for system integration were pushed back to starting in late February. With a lot of integration and testing work to occur over a relatively short period of time, the schedule got delayed and margins were used up. Due to a UAV crash, a motor burned out and needed to be replaced, pushing the schedule back further. While there were plenty of unforeseen delays that occurred, all of the subsystems and components were successfully tested. The full day in the life test, however, was delayed and sonar data collection did not immediately start on contact with the river surface.

The second most prominent risk, the float deployment mechanism, led to last-minute design changes as well as schedule slips. Due to the configuration of the motor, spool, and fishing line holding the float, the motor strength did not meet what was desired as it had a hard time pulling up and deploying the full-weighted float successfully. Due to this problem, the stability pontoons on the deployable float were removed to cut down on weight. In addition, the spool diameter was decreased to increase the torque force that the motor could supply. These design changes, while not perfect, enabled to float to be deployed successfully to meet requirements. The back up option of having the float pre-deployed before UAV takeoff was carried out several times during testing for stability analysis and to enable testing of the sonar float data collection.

While many of the risks identified in the first semester of planning the project aligned with what actually occurred in project manufacturing, integration, and testing, some risks that were not initially identified came to light. These included water leaking into the sonar float, faulty water sensor functionality, and electronics/component housing. After repeated float data collection testing, some tests resulted in small amounts of water entering the float and pooling at the bottom. While the electronics are elevated within the float and did not get damaged, this design flaw must be improved upon for safe and reliable sonar depth data collection. Similarly, during our last day of testing, the main flaw in our system test was the float water sensor (used to initiate sonar data collection) not triggering data collection. This resulted in no sonar data being collected with our system, which ranks as severe on a risk matrix given calculating river discharge requires a reliable depth profile to be constructed. This risk could have been mitigated with more testing time and design iterations to validate the float system's reliability. The last unplanned risk that came up was the electronics and component housing. Due to the amount of connections on the on-board unit, the wiring got quite messy and difficult to manage. This was mitigated by putting heat shrink around connections as well as using cable management sleeving to keep certain groupings of wires together. Another component housing risk that came up was with the float not being able to snugly fit beneath the on-board unit. Due to this design flaw, the float had to be pre-deployed a set length away from the drone before UAV takeoff. While this wasn't a mission-critical design flaw, it would've been preferable to have a snugly fit float beneath the system. This risk could have been mitigated through more design iteration and lengthening the UAV's carbon fiber legs.

All of the aforementioned risks were tracked by communicating and noting any problems or blockers from component as well as integration testing. These problems were recorded and tracked throughout our remote and in-person meetings. Priority would be put on solving the most mission-critical problems and risks that came up.

7 Project Planning

To ensure that the project design and critical project elements could be prototyped, manufactured, integrated, and tested, a concise organization structure, work breakdown structure, and test plan were developed to enable project success leading up to the final product demonstration.

7.1 Organizational Chart

To ensure all team members had meaningful roles and responsibilities, an organizational chart with predetermined team roles was created. All members were given roles to ensure a product for project customers, Steve Nerem and Toby Minear, could be effectively put together.

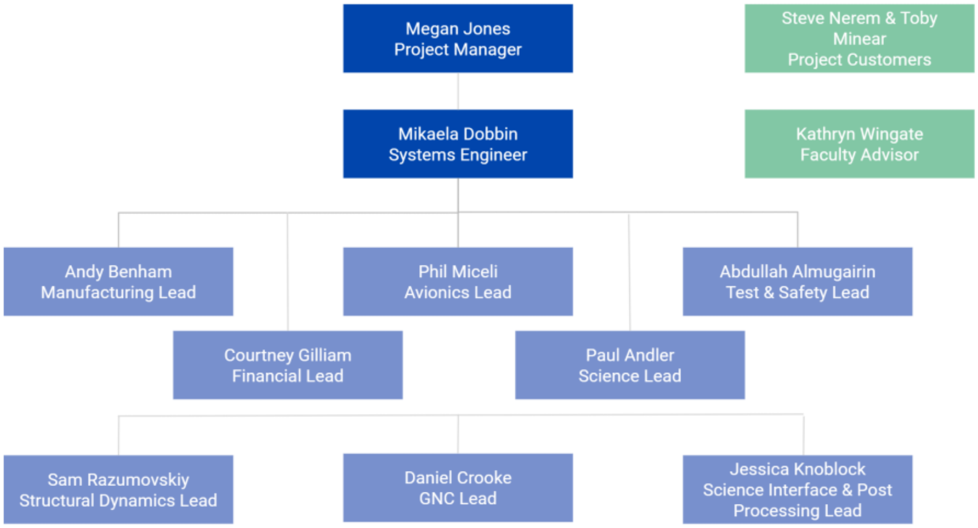


Figure 50: Organizational Chart

7.2 Work Breakdown Structure

Below, Team RiBBIT's work breakdown structure is depicted, showing the primary work products. The green boxes show work that has been completed. The work breakdown structure is broken up into deliverables as well as subteam tasks. The work products were determined by laying out the optimal final deliverable and working backwards to identify critical tasks that must be accomplished by each subsystem to enable critical project elements to be met. Subteam researching, trades, product selection, and design analyses have were completed in the Fall semester. Prototyping, design development and iteration, manufacturing, testing, and component integration were completed in the Spring semester.

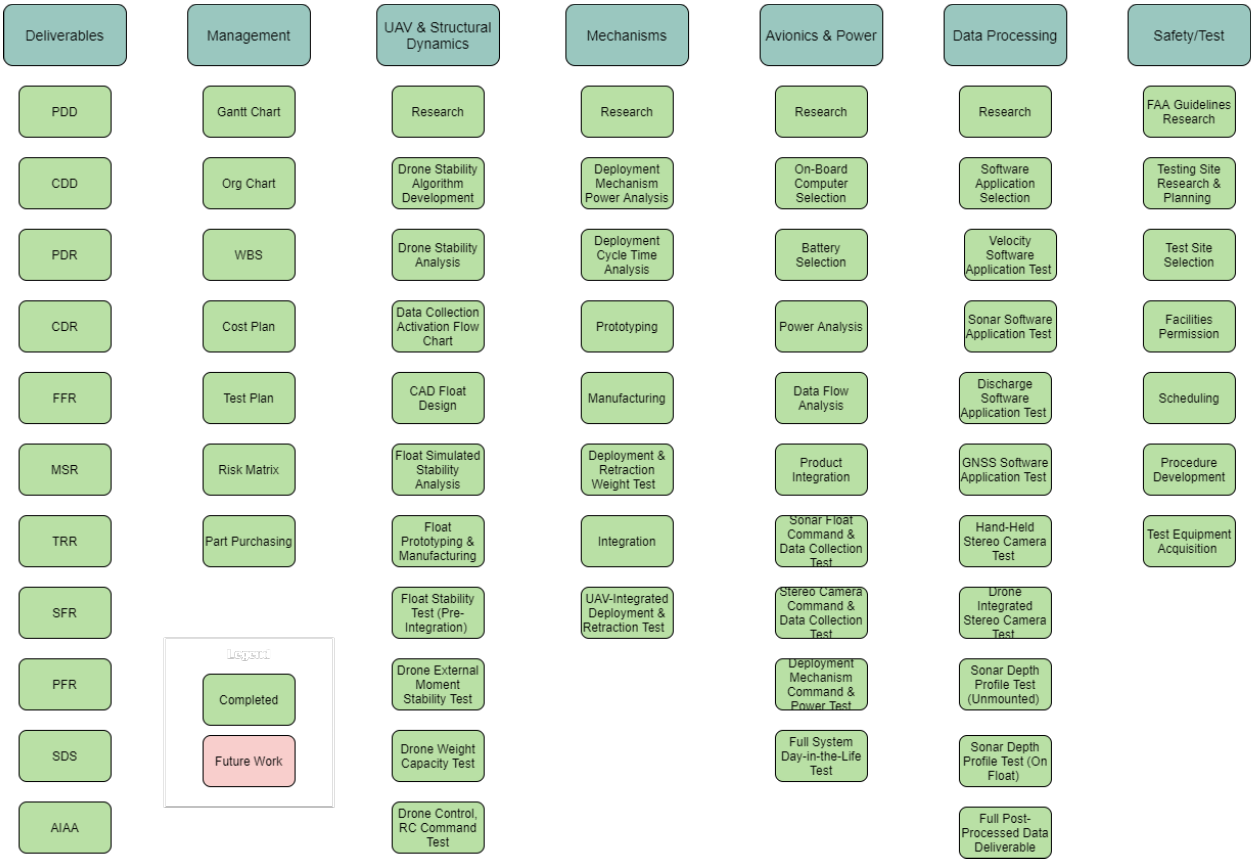

Deliverables
PDD
CDD
PDR
CDR
FFR
MSR
TRR
SFR
PFR
SDS
AIAA
Management
Gantt Chart
Org Chart
WBS
Cost Plan
Test Plan
Risk Matrix
Part Purchasing
Legend
Completed
Future Work
UAV & Structural Dynamics
Research
Drone Stability Algorithm Development
Drone Stability Analysis
Data Collection Activation Flow Chart
CAD Float Design
Float Simulated Stability Analysis
Float Prototyping & Manufacturing
Float Stability Test (Pre-Integration)
Drone External Moment Stability Test
Drone Weight Capacity Test
Drone Control, RC Command Test
Mechanisms
Research
Deployment Mechanism Power Analysis
Deployment Cycle Time Analysis
Prototyping
Manufacturing
Deployment & Retraction Weight Test
Integration
UAV-Integrated Deployment & Retraction Test
Avionics & Power
Research
On-Board Computer Selection
Battery Selection
Power Analysis
Data Flow Analysis
Product Integration
Sonar Float Command & Data Collection Test
Stereo Camera Command & Data Collection Test
Deployment Mechanism Command & Power Test
Full System Day-in-the-Life Test
Data Processing
Research
Software Application Selection
Velocity Software Application Test
Sonar Software Application Test
Discharge Software Application Test
GNSS Software Application Test
Hand-Held Stereo Camera Test
Drone Integrated Stereo Camera Test
Sonar Depth Profile Test (Unmounted)
Sonar Depth Profile Test (On Float)
Full Post-Processed Data Deliverable
Safety/Test
FAA Guidelines Research
Testing Site Research & Planning
Test Site Selection
Facilities Permission
Scheduling
Procedure Development
Test Equipment Acquisition

Figure 51: Organizational Chart

7.3 Work Plan

Below, the work plan is shown in the form of a Gantt chart. The critical path tasks include manufacturing, system assembly, and testing. A more detailed test plan can be seen earlier in the document. The test plan presents the component and integration testing that were carried out as subsystems were manufactured. Margin was built into the testing plan to ensure that all testing would be able to be completed even with COVID-19 related closures and facility scheduling. Even though margin was added in the schedule, the team faced testing and COVID-related set-backs that pushed the day in the life testing past the initially set deadline.

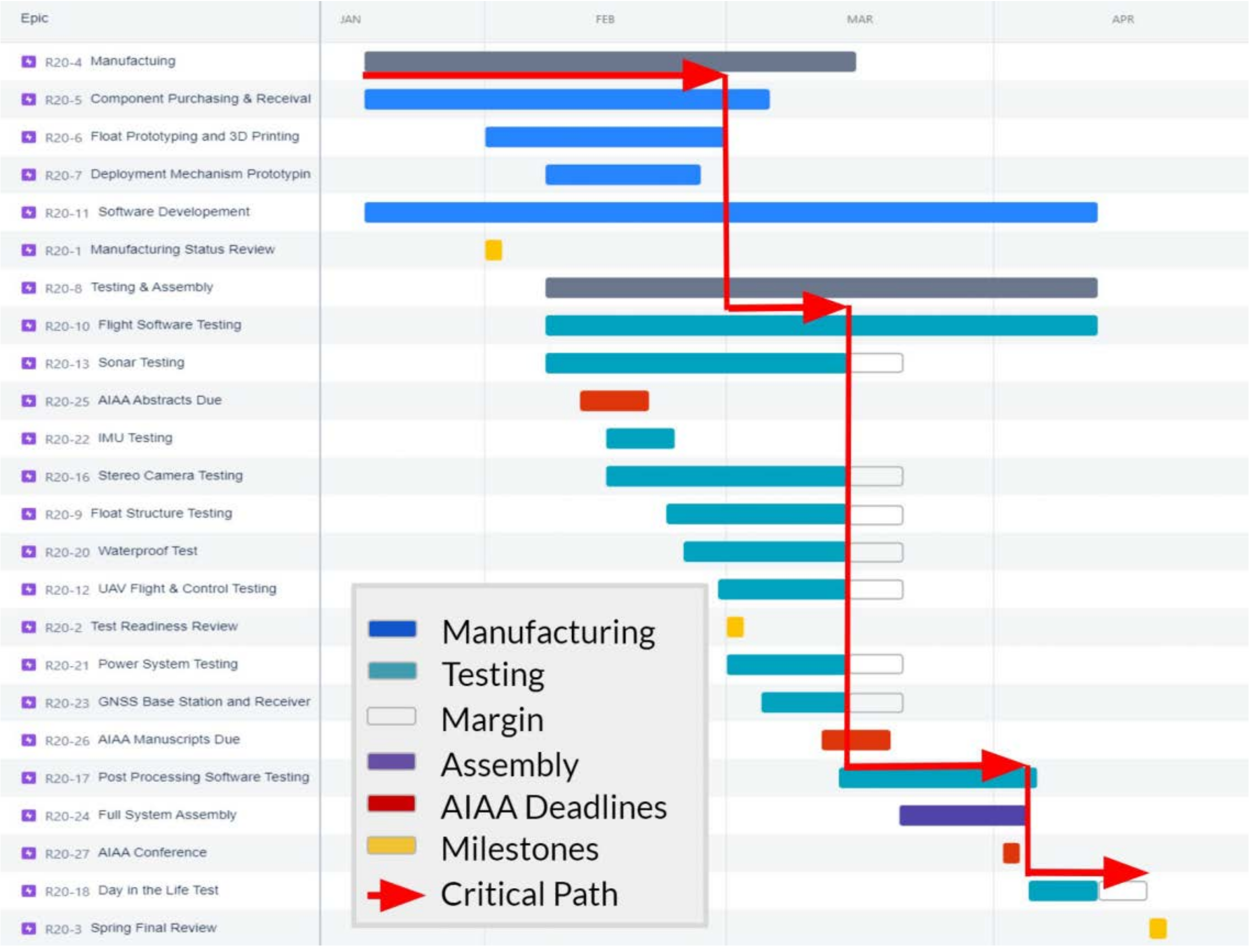
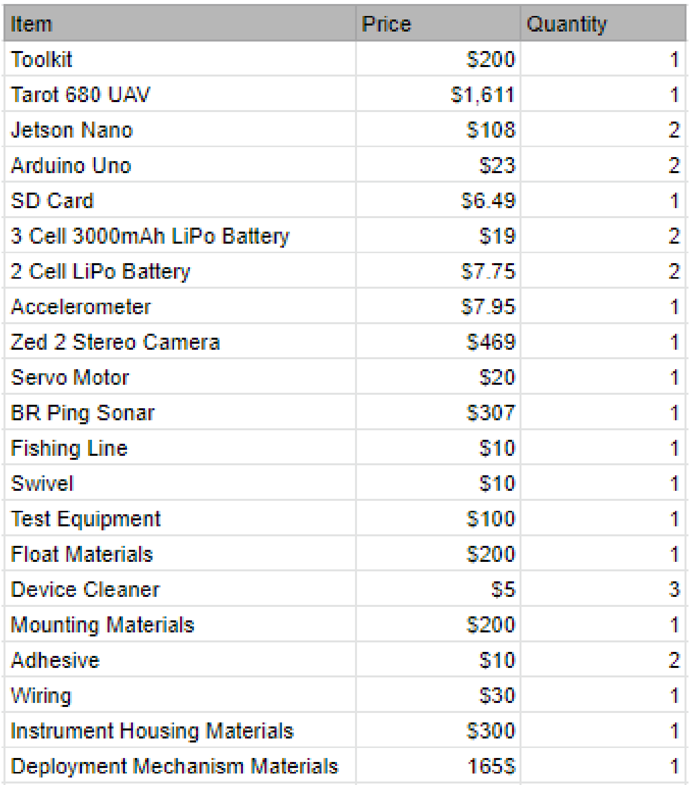


Figure 52: Work Plan

7.4 Cost Plan

The starting budget for this project was $5000. The The budget is broken down to four categories: scientific instruments, manufacturing materials, electronics, and miscellaneous fees. The scientific instruments were determined in the fall with no cost uncertainty. As the design process progressed, more and more materials were needed for troubleshooting and additional iterations. The planned budget was $3358.09, and the actual amount spent by the team was $4770.63. The contingency budget of $1641.91 was sufficient for our needs. This was more than the suggested contingency budget amount, which is 10 percent. The most uncertainty in the budget surrounded the manufacturing materials for the deployment mechanism and parts needed to set-up the electronic systems. In January and February, $3112.61 and $705.45 were spent on supplies. In March and April, $596.38 and $427.23 were spent, mostly on unplanned expenses. This supports the initial The decision to have a substantial contingency budget. The figure below displays our predicted cost plan from the beginning of manufacturing. Every section of this budget for materials is an estimate. The scientific instruments were the set costs here. The instrument housing materials ended up being the lowest expense category we had, and the highest expense category was barely accounted for here at this point. The test equipment category refers to the replacement parts for those that broke during the testing process. There were very few places we could locate these parts, and they are very specific to the brand. There were no options to go with any other brand. Doing so would have created an instability with the adjusted weight distribution. The propellers were fragile, so chipping one through the design process was inevitable. All purchases and itemized receipts are located in the monthly Concur procurement card reports.

Costs for this project very well could have been reduced to under $4000, less than 80 percent of what was spent. Nearly every purchase in the month of April had to be rush shipped. This was caused by unexpected schedule delays and testing accidents that required replacements that we can only source from another country. While the Tarot brand UAV system proved itself to be a great choice, it was more difficult to source the parts with few options available. Also, the shipping for the lipo batteries was more expensive and could not be rush shipped for safety reasons. The additional, unused materials purchased for this project will be donated to the Aerospace Engineering department at CU Boulder.

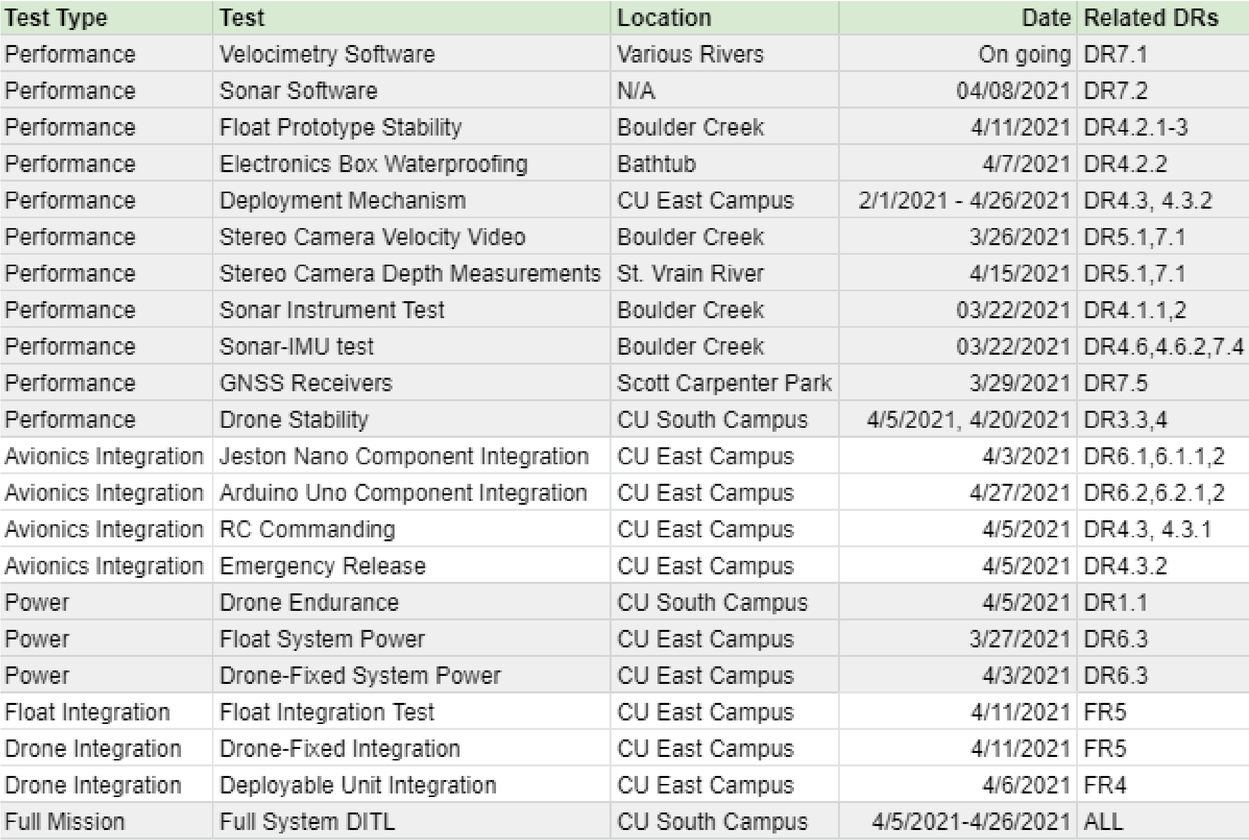


Item	Price	Quantity
Toolkit	$200	1
Tarot 680 UAV	$1,611	1
Jetson Nano	$108	2
Arduino Uno	$23	2
SD Card	$6.49	1
3 Cell 3000mAh LiPo Battery	$19	2
2 Cell LiPo Battery	$7.75	2
Accelerometer	$7.95	1
Zed 2 Stereo Camera	$469	1
Servo Motor	$20	1
BR Ping Sonar	$307	1
Fishing Line	$10	1
Swivel	$10	1
Test Equipment	$100	1
Float Materials	$200	1
Device Cleaner	$5	3
Mounting Materials	$200	1
Adhesive	$10	2
Wiring	$30	1
Instrument Housing Materials	$300	1
Deployment Mechanism Materials	165$	1

Figure 53: Cost Plan

7.5 Test Plan

Given the complexity of our system, it was critical to ensure that there was a concise test plan to enable component and integration tasks to be completed and tracked. The facilities used were the Clair Small Pool, CU South Campus, the senior projects work space, the aerospace electronics shop, and the ASPEN flight space. All facilities were utilized following permission and scheduling with lab space coordinators.



Test Type	Test	Location	Date	Related DRs
Performance	Velocimetry Software	Various Rivers	On going	DR7.1
Performance	Sonar Software	N/A	04/08/2021	DR7.2
Performance	Float Prototype Stability	Boulder Creek	4/11/2021	DR4.2.1-3
Performance	Electronics Box Waterproofing	Bathtub	4/7/2021	DR4.2.2
Performance	Deployment Mechanism	CU East Campus	2/1/2021 - 4/26/2021	DR4.3, 4.3.2
Performance	Stereo Camera Velocity Video	Boulder Creek	3/26/2021	DR5.1,7.1
Performance	Stereo Camera Depth Measurements	St. Vrain River	4/15/2021	DR5.1,7.1
Performance	Sonar Instrument Test	Boulder Creek	03/22/2021	DR4.1.1,2
Performance	Sonar-IMU test	Boulder Creek	03/22/2021	DR4.6,4.6.2,7.4
Performance	GNSS Receivers	Scott Carpenter Park	3/29/2021	DR7.5
Performance	Drone Stability	CU South Campus	4/5/2021, 4/20/2021	DR3.3,4
Avionics Integration	Jeston Nano Component Integration	CU East Campus	4/3/2021	DR6.1,6.1.1,2
Avionics Integration	Arduino Uno Component Integration	CU East Campus	4/27/2021	DR6.2,6.2.1,2
Avionics Integration	RC Commanding	CU East Campus	4/5/2021	DR4.3, 4.3.1
Avionics Integration	Emergency Release	CU East Campus	4/5/2021	DR4.3.2
Power	Drone Endurance	CU South Campus	4/5/2021	DR1.1
Power	Float System Power	CU East Campus	3/27/2021	DR6.3
Power	Drone-Fixed System Power	CU East Campus	4/3/2021	DR6.3
Float Integration	Float Integration Test	CU East Campus	4/11/2021	FR5
Drone Integration	Drone-Fixed Integration	CU East Campus	4/11/2021	FR5
Drone Integration	Deployable Unit Integration	CU East Campus	4/6/2021	FR4
Full Mission	Full System DITL	CU South Campus	4/5/2021-4/26/2021	ALL

Figure 54: Test Plan

8 Lessons Learned

When it comes to manufacturing and design in a senior project, above all else, remember to not limit yourself by what you think may or may not be possible. If someone had told us this project would involve designing a boat that we welded together using PLA that was hung by fishing wire, we wouldn't have believed you. It can be difficult because one must balance thinking creatively and out of the box but also still try to tackle things that are realistic. Talking to your team members and asking for creative solutions to your projects problems can be a great way to generate new ideas. Or presenting them with what a designer has come up with and asking 'how can we improve this design' also yielded a lot of great results. There is no such thing as starting too early. It can also be difficult to know when to start prototyping vs spending more time on designing. In our case it was almost always better to start prototyping since there are always going to be things you don't anticipate and thus need to add to the design anyways.

The largest lesson learned from the software development on this project was to always take an iterative approach. The reason for this being that using this process allows for a more contained style of debugging as well as an iterative approach to validating the functionality of each software addition. It was found to be far simpler and time-efficient to develop all the subroutines separately and then later combine them into one cohesive program than trying to build the entire package in one go.

There are a number of lessons learned in the development of this project from a systems engineering lens. During the project design phase, it's important to work with the team and customers to identify the important parts of the project in order to clearly identify the scope. It is very easy to create a scope that is too large (this may have happened with this project), so it is important to check with the team and advisors to make sure that the scope is within reason to be accomplished in a matter of nine months. For new seniors entering this course, if something sounds too simple, it likely isn't so do not underestimate "easy" sounding tasks/challenges. During the design implementation phase, it was difficult to create a realistic manufacturing and test schedule. Along the same lines, manufacturing and integration should be worked on a soon as possible to identify what problems that were unplanned for come up. Given the complexity of this project and the amount of integration work, it would have been beneficial to start sooner to work through the problems that came up and ultimately pushed our schedule back.

This project took place during the COVID 19 pandemic, therefore it should be kept in mind that these are lessons learned for a senior project in these circumstances. With the pandemic, no in person meetings occurred in the first semester which made it difficult to have productive design meetings. In the second semester, system integration only began when all appropriate members were comfortable with coming in with larger groups to the project space. Due to this, the system integration and testing got condensed to the very last couple of months to complete the project. Due to these logistics, it was difficult to adequately integrate, test, and troubleshoot all necessary subsystems before the semester ended. This all said, if all of this was accomplished over zoom, other projects under normal circumstances have the potential to accomplish much more.

9 Individual Report Contributions

Team Member	**Contributions**
Abdullah Almugairin	Boat Stability, Deployment Mechanism feasibility, Model, Testing and Prototyping
Paul Andler	Bathymetric Technique, depth uncertainty quantification, angular displacement correction, Depth data post-processing, Float instrument software and software manufacturing, (SONAR,IMU, depth profile correction) validation, contributions to lessons learned
Andy Benham	Conceptual Velocimetry, Float Stability, Dynamics Research, Float Design, Assembly, Manufacturing
Daniel Crook	All Positional Determination and Georeferencing (PDG) related work
Mikaela Dobbin	FBD, Requirements, Velocity Error Quantification, Test Plan
Courtney Gilliam	Cost Plan, Experimentation, Product Inquiry
Megan Jones	Project Purpose, CONOPS, Risks, Project Planning, Software Post-Processing
Jessica Knoblock	Velocity Data Collection, Velocity Error Propagation, Image Pre-processing and Post-processing for Velocity Calculations
Phil Miceli	All avionics related conceptual design, detailed design, and verification and validation
Sam Razumovskiy	UAV-Detailed Design, Payload Housing, UAV-Basline Design, Drone Test Flights, Full System Stability Simulation

Table 6: Team Contributions

References

[1] Durand, M., Gleason, C. J., Garambois, P. A., Bjerklie, D., Smith, L. C., Roux, H., Rodriguez, E., Bates, P. D., Pavelsky, T. M., Monnier, J., Chen, X., Baldassarre, G. D., Fiset, J.-M., Flipo, N., Frasson, R. P. D. M., Fulton, J., Goutal, N., Hossain, F., Humphries, E., Minear, J. T., Mukolwe, M. M., Neal, J. C., Ricci, S., Sanders, B. F., Schumann, G., Schubert, J. E., and Vilmin, L., "An intercomparison of remote sensing river discharge estimation algorithms from measurements of river height, width, and slope," *Water Resources Research*, vol. 52, 2016, pp. 4527–4549.

[2] Engel, F. Guidelines for the Collection of Video for Large Scale Particle Velocimetry (LSPIV) - OSW Surface Velocity Workgroup - MyUSGS Confluence. USGS Surface Velocity Workgroup. https://my.usgs.gov/confluence/pages/viewpage.action?pageId=546865360.

[3] Engel, F.E., Patalano, A., Garcia, C.M., 2017, Use of Large-Scale Particle Image Velocimetry (LSPIV) for continuous streamflow gaging during flood events, in Hydraulic Measurements and Experimental Measurements (HMEM) 2017: Durham, NH.

[4] Fulton, J. How to Translate Surface-Water Velocities into a Mean-Vertical or Mean-Channel Velocity. USGS Surface Velocity Workgroup. https://my.usgs.gov/confluence/display/SurfBoard/How+to+translate+Surface-water+Velocities+into+a+Mean-vertical+or+mean-channel+Velocity.

[5] Hock, R., G. Rasul, C. Adler, B. Cáceres, S. Gruber, Y. Hirabayashi, M. Jackson, A. Kääb, S. Kang, S. Kutuzov, Al. Milner, U. Molau, S. Morin, B. Orlove, and H. Steltzer, 2019: *High Mountain Areas. In: IPCC Special Report on the Ocean and Cryosphere in a Changing Climate* [H.-O. Portner, D.C. Roberts, V. Masson-Delmotte, P. Zhai, M. Tignor, E. Poloczanska, K. Mintenbeck, A. Alegria, M. Nicolai, A. Okem, J. Petzold, B. Rama, N.M. Weyer (eds.)]. In press.

[6] J. Gelling, "The Axe Bow: The Shape of Ships to Come," in 19th International HISWA Symposium on Yacht Design and Yacht Construction, Amsterdam, The Netherlands, 13 and 14 November 2006.

[7] Molland, A. F., Ed. Chapter 3 - Flotation and Stability. In The Maritime Engineering Reference Book, Butterworth-Heinemann, Oxford, 2008, pp. 75–115.

[8] Patalano, A., García, C.M., Rodríguez, A., 2017. Rectification of Image Velocity Results (RIVeR): A simple and user-friendly toolbox for large scale water surface Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV) 109, 323–330. doi:10.1016/j.cageo.2017.07.009

[9] Patalano, A., García, C. M., and Rodríguez, A. RIVeR Tutorials. RIVeR. https://riverdischarge.blogspot.com/p/tutorials.html

[10] Tauro, F., Petroselli, A., and Grimaldi, S. "Optical Sensing for Stream Flow Observations: A Review." Journal of Agricultural Engineering, Vol. 49, No. 4, 2018, pp. 199–206. https://doi.org/10.4081/jae.2018.836.

[11] Thielicke, W. and Stamhuis, E.J. (2014): PIVlab – Towards User-friendly, Affordable and Accurate Digital Particle Image Velocimetry in MATLAB. Journal of Open Research Software 2(1):e30, DOI: http://dx.doi.org/10.5334/jors.bl

[12] Turnipseed, D. Phil, and Vernon B. Sauer. "Discharge Measurements at Gaging Stations." Report. Techniques and Methods. Reston, VA, 2010. *USGS Publications Warehouse*. https://doi.org/10.3133/tm3A8.

[13] Ulstein, "X-bow," Ulstein, 2005. [Online]. Available: https://ulstein.com/innovations/x-bow. [Accessed Nov 20 2020].

10 Appendix

10.1 Conceptual Design Alternatives Considered

10.2 Bathymetry Technique

As previously discussed, calculating river discharge involves two quantities: area and velocity. Bathymetry is concerned with the cross-sectional area of the river. The National Oceanic and Atmospheric Administration (NOAA) defines Bathymetry as "...the study of the "beds" or "floors" of water bodies, including oceans, rivers, streams, and lakes." [11].

When it comes to the technology used to measure the depth of a body of water one has to factor in not only the cost, weight, and size of the sensor unit but also the accuracy and usability. For the application of having a UAS mounted system, three depth sensing technique options were considered. These include LiDAR (light detection and ranging), SONAR (sound navigation and ranging) and ADCP (Acoustic Doppler Current Profiling). ADCP is similar to SONAR in the way it would be implemented, but it has the additional ability to measure flow velocity. The major difference between the three systems is the fact that LiDAR does not require contact with the water while Sonar and ADCP instruments do. The Sonar and ADCP instruments must be immersed under the surface, adding another level of complexity to the whole system. This approach would require developing a method of sensor package deployment, but a LiDAR system could easily be flown over the water while mounted on the drone.

10.2.1 LiDAR

Light detection and ranging, more commonly known as LiDAR, works by emitting pulses of light waves that bounce off of surrounding objects and return to the sensor. The sensor uses the time taken for the pulses to return to the sensor to calculate the distance from the sensor to the surrounding objects. Most commercial LiDAR systems use lasers that operate in the infrared spectrum mainly due to their availability. However, for bathymetric applications it is critical to have a LiDAR system that operates in the correct spectrum of 532 nm (green). This is because water absorbs different colors of light at very different rates. Infrared wavelengths can barely make it past the water's surface because wavelengths in this spectrum are absorbed the fastest. In contrast, green is absorbed the least by water. Systems using this type of laser are able to take accurate measurements up to a depth of 20m.

10.2.1.1 RIEGL BDF-1 Laser Range Finder

The RIEGL BDF-1 is a laser range finder specifically designed for bathymetric surveying. The device is intended to collect river profile data when operated from a UAV at low altitudes. The scanner sends out laser pulses of 532nm at a rate of 4 kHz. Optional features for this device include lens tilt compensation, an embedded GNSS-inertial system, and external digital cameras. The scanner produces highly accurate and reliable data. The downside of this system is the weight and the cost. The sensor weighs approximately 5.3kg and the cost of over 10,000 dollars exceeds the entire project budget. This would be an ideal sensor system to achieve the project's goals, but it is not a realistic candidate.

Figure 55: RIEGL BDF-1

Pros	Cons
Does not require in-situ deployment	Expensive
Integrated IMU, GNSS, and data storage unit	Heavy for a drone to carry
Can measure up to 50m in depth	Can only measure a minimum depth of 1m
Approximate accuracy of 20mm	Measurement accuracy heavily dependant on water clarity
High data collection rate	
Comes with up to two external cameras	
532nm Laser wavelength	
Designed for UAV/drone mounting	

10.2.1.2 RIEGL VQ-840-G Laser Scanner

The RIEGL VQ-840-G is a laser scanner designed for bathymetric and topographic UAV-based surveying. The device sends out 532nm laser pulses at a rate of 50 kHz to 200 kHz. The scanner is highly accurate and has a number of optional features including an inertial navigation system, an infrared rangefinder, and an integrated digital camera. The downside of this product is that it was made for larger UAVs, weighing approximately 12kg. In addition, this sensor is far out of budget by about eight thousand dollars, and therefore not a sound candidate for the purposes of this project.

Figure 56: RIEGL VQ-840-G

Pros	Cons
532nm Laser Wavelength	Expensive
Integrated inertial system navigation	Heavy
Accuracy of approximately 20mm	Minimum depth measurement range of 20m
Optional integrated digital camera	

10.2.2 SONAR

SONAR, also known as sound navigation and ranging is an acoustic technique primarily used to measure distances underwater. There are two main types of sonar. One being passive and the other active. The difference between them is that the passive sonar only listens while the active sonar sends out bursts of sound known as 'pings' and then listens for the echo. Distance is calculated by starting a timer when the ping is sent and stopped when the echo is received. Given the time and the speed of sound in water, the distance is calculated inside of the unit. Active sonar is most useful for bathymetric applications as a ping is required for distance measurements.

10.2.2.1 BlueRobotics Ping Sonar

The Ping sonar by BlueRobotics is a great option for applications in bathymetric surveys for a multitude of reasons, the first of which being its price. Coming in at under 300 dollars, it is a sensor package comfortably within budget. Furthermore, with its measurement range being from 0.5m-30m, this device is able to cover the set levels of success for depth as well as accuracy according to the manufacturers value of 0.5%, as a function of depth. In addition, it does not employ any optical techniques. Therefore, the system is nearly unaffected by turbid water. Due to its small size and weight, this sensor only requires a smaller drone to lift. Finally the unit can also interface with Arduino, C++ and Python libraries, allowing for simple data handling and analysis.

Figure 57: Blue Robotics Sonar

Pros	**Cons**
Highly accurate, depth measurements up to 30m Minimally dependant on water clarity Cost efficient Extremely light weight Easy data handling with provided libraries Fully Water proof up to 300m Included bottom tracking	Requires surface level in-situ deployment Cannot measure flow velocity

10.2.2.2 Allied SICK UM-30

Similar to the ping echo sounder, the Allied SICK UM-30 also uses SONAR to take distance measurements. The one advantage that it has over the Ping sonar is a minimum measurement threshold of 0.1m. It also has a low purchase price of $370 as well as a light weight and form factor. That being said, after consulting with the manufacturer it became apparent that the maximum depth it can measure is approximatly 1m.

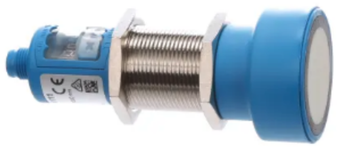


Figure 58: Allied ultrasonic sensor

Pros	**Cons**
Highly accurate	Requires surface level in-situ deployment
Minimally dependant on water clarity	Cannot measure flow velocity
Cost efficient	Very limited underwater range
Extremely light weight	Only IP67 rated (water proof for 30 min at 1m)
	Unknown difficulty for data handling

10.2.3 ADCP

An Acoustic Doppler Current Profiler (ADCP) uses active sonar to measure the depth of the river. An ADCP contains additional transducers to emit sound pings and measures the Doppler shift of the returning pings to calculate the instantaneous velocity of the entire column beneath the instrument [12],[14]. An ADCP is the preferred method of the US Geological Society for preforming Bathymetric surveys as it is essentially an 'all in one' instrument [6]. ADCPs come in a wide variety of sizes, some of which can measure depths of 6000+ meters for surveying of the ocean floor from boats. Some are designed to withstand immense pressures and are attached to buoy's that are sunk to various depths in the ocean. Project RiBBIT is interested in ADCPs that are normally attached to Unmanned Surface Vehicles that are traditionally used in bathymetric river surveys.

10.2.3.1 Sontek RiverSurveyor M9

The RiverSurveyor M9 is an extremely powerful, compact ADCP. It comes with a built in temperature sensor, compass/tilt sensor, and internal storage. It weighs approximately 2.3kg in air with a velocity profiling range of 40m. This is under its own power supply which consists of standard AA batteries that give it up to 8 hours of continuous operation. The cost of this device ccosts over $15000.

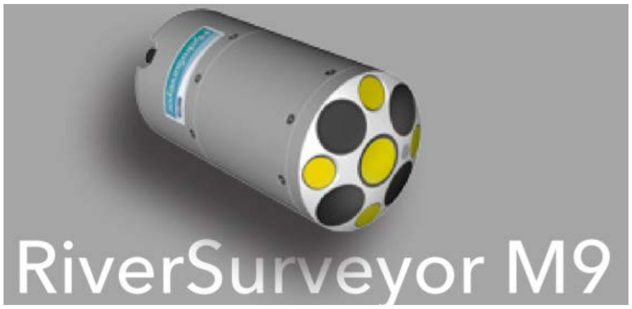


Figure 59: The Sontek RiverSurveyor M9

Pros	Cons
Large depth range [80m]	Requires in-situ deployment
No post-processing analysis required	Expensive [$15k+]
Can measure column flow velocity below device	No measurement of surface velocity
Low minimum depth measurement threshold [0.2m]	2.3kg

10.2.3.2 Rowe Technologies OEM

The OEM is a customizable ADCP package from Rowe Technologies and was likely the best candidate for this project were it in budget. Many ADCPs have capabilities that far exceed what is realistically needed for our project. Since the OEM is customizable it could have been tailored much more closely to our requirements. This also makes it the cheapest option as it wouldn't come with anything that wasn't specifically for our mission. However, this package is out of our budget and costs a minimum of $12000.

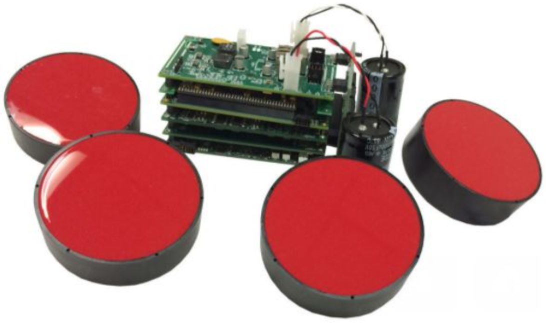


Figure 60: The Rowe OEM package

Pros	Cons
Customizable Low minimum depth measurement threshold [0.2m] Can measure column flow velocity below device No post-processing analysis required	Expensive [$12k after student discount] No measurement of surface velocity

10.3 Deployed SONAR Float Angular Displacement Correction Technique

Following the trade study of various bathymetry techniques, it was clear that sonar is the most affordable and realistic option to pursue with regards to the project requirements. As this technique requires contact on the river surface, a deployable sensor unit shall be developed. This floating sensor unit needs to enable effective data collection of the sonar instrument, which requires the instrument to be partially submerged in the water, pointing downwards. The floating mechanism can either be weighted to ensure direct downward facing of the instrument, a gyroscope can be installed to enable post-processing to get the proper depth measurements, or the sensor could be installed inside of a gimbal that is attached to the float in order to keep it pointing downward.

10.3.1 Weighted Float Technique

This design option considers weighing the float in a way similar to how a sailing boat is, increasing its stability in the water. This specifically being a weighted keel device underneath the float which lowers the system's center of gravity, therefore increasing its stability and resistance to waves and other disturbances that may change the float's relative angle to the gravity vector. While this system is simple, affordable and passive in its function, there are certain drawbacks to using it. The advantages and disadvantages are shown in the table below.

Pros	Cons
Cheap Passive	Very Heavy (sailing boat's keels are 40% their overall weight) Only dampens angular displacement Creates more drag due to higher surface area

10.3.2 Post-Processing Technique

In this approach, there is no extra weight added to the float with an exception of a 2-axis gyroscope that can interface with the same data logging computer that the sonar is communicating with. If the angle of the sonar unit is known at all times, any gravity vector angle deviation that it may experience during data collection could be corrected for in post-processing, using simple trigonometric relations.

Pros	Cons
Cheap and light (<5g) Provides full solution Low complexity Produces no extra drag	Adds extra step to post-processing

10.3.3 Gimbal Technique

This solution considers the sonar sensor mounted inside of a gimbal that is attached to the float. The gimbal would be able to correct, in real time, any disturbances that the sensor may be subjected to by the flow. This would require no extra work during post-processing the data. The down side, however, is that the gimbal would have to be water proofed and would be rather heavy, weighing approximately 250g.

Pros	**Cons**
Is a complete solution	Heavy Complex Expensive Produces extra drag Has to be actively controlled

10.4 Sonar Deployment Mechanism Technique

Following the science instrument trade study, it was identified that the sonar instrument is most optimal for collecting river depth measurements. To enable data collection, the sonar needs to be deployed from the instrument suite mounted on the drone to contact the river's surface. The mechanical and software interfacing necessary to command this mechanism is also an important consideration. The options for the deployment mechanism are outlined below. All of the options, except for the last, would require a line connecting the sonar suite to the mounted instrument suite. The material options for this feature are presented in the table below, acknowledging there are many other options and thicknesses that could be pursued.

Figure 61: Deployment Line Material Options

Material	**Weight**	**Cost**	**Diameter**	**Max Weight Capacity**
Fishing Line	1.14 g/m	\$0.21/m	0.26mm	6.8kg
Cable	23.3 g/m	\$0.98/m	2.38mm	45kg
Rope	153 g/m	\$0.78/m	9.52mm	167.98kg

10.4.1 Hanging Cable System

This option outlines if RiBBIT decides to have the sonar hanging from a cable with no controlled deployment mechanism on the primary instrument-suite. The cable would be connected to the main instrument payload with a carabiner attached to a custom hook. The hook would have the ability to be commanded by the drone pilot (through connection to the on-board flight controller) to release the attached payload in case of emergency. The arrow in the figure below depicts the opening of the hook once the flight pilot has deemed the situation dangerous, thus sending a command to the on-board flight controller to drop the payload. This option would remove the additional complexity required if a mechanical rappelling system were pursued.

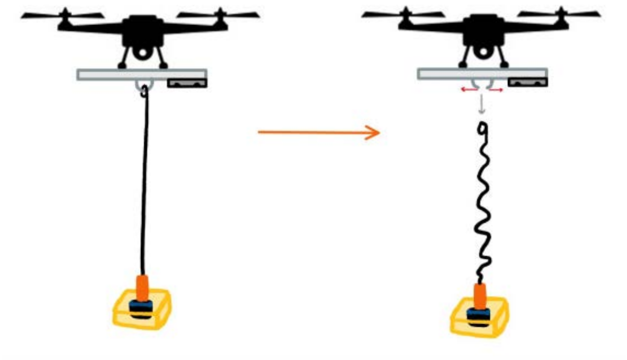


Figure 62: Hanging Cable Deployment Mechanism

Pros	Cons
Cheap Mechanically simple, not much design work necessary	Not compact for flight Potential of getting snagged May get in way of stereo camera data collection Uncertain cost

10.4.2 Cable Rappelling System

Another option to be pursued would be a mechanical system to coil the cable connected to the sonar float so that it is initially compact with the instrument unit. After the drone has flown over the river, the drone pilot would flip a switch to initiate a command to smoothly deploy the cable to drop the sonar suite to the water's surface. The drone would fly bank to bank to enable data collection and then the switch would be flipped to coil the payload back up to the main instrument suite. This sonar deployment mechanism would be a more mechanically challenging system to pursue.

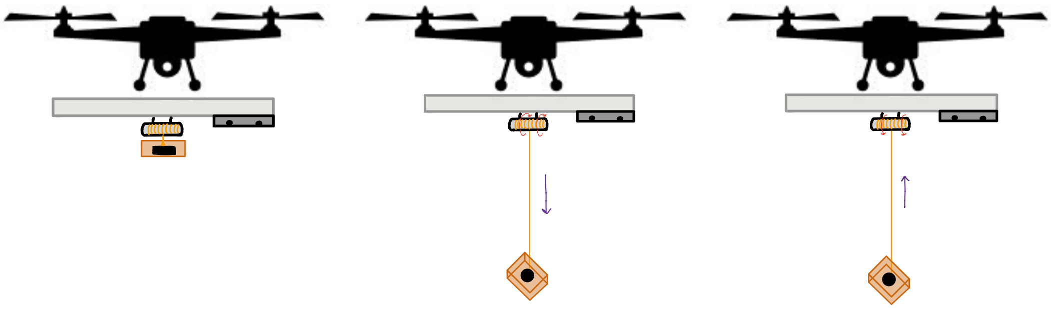


Figure 63: Cable Rappelling Deployment Mechanism

Pros	Cons
Enables compact payload during flight to field site Would enable sonar float to not be in FOV of stereo camera	Mechanically complicated Uncertain cost

10.4.3 Tarot X6 Payload Release Drop Mechanism

The Tarot X6 Payload Release Drop Mechanism is designed for dropping payloads carried on the Tarot X6 drone. The payload drop is initiated with the click of a button on the Tarot X6 remote controller. If this option were pursued, the drop would likely not be a gradual descent into the water but a free fall. Once the unit was dropped, bringing it back up would not be easily pursued, so the flight vehicle would need to return to the pilot with the sonar unit deployed below. This option would only be utilized if the Tarot X6 drone is chosen as the flight vehicle. The product includes the payload release drop mechanism, mounting onto the drone, the setup, and the configuration with the controller. The cost of this mechanism is $249.99 USD.

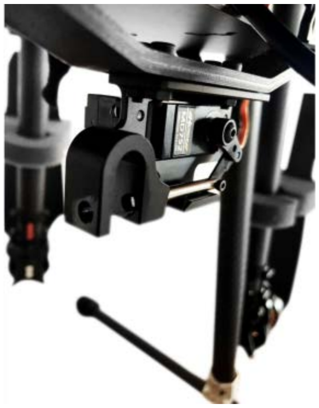


Figure 64: Tarot X6 Payload Release Drop Mechanism

Pros	Cons
Easy to integrate with Tarot X6 Drone Controlled via Tarot X6 controller High weight carrying/drop capacity (5kg)	Fully dependant on drone choice Mechanism is only foundation, more components would need to be added for smooth sonar release/retrieval

10.4.4 Rigid Sonar Deployment System

A rigid deployment option, unlike the others previously mentioned, would not include a line being dropped from the main payload to the river surface with the attached float. Instead, it would consist of a retractable pole with the sonar instrument at the end of the mechanism. The drone pilot would flip a switch and a command would result in the pole to retract down to the river surface. The drone would fly from bank to bank and once adequate data has been collected the mechanism would retract back up for the return flight.

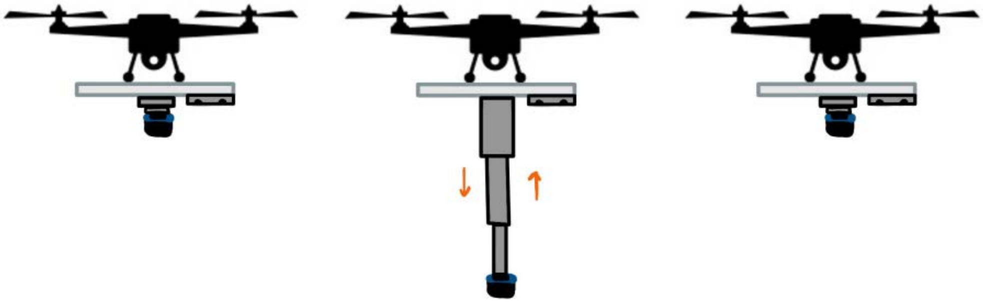


Figure 65: Rigid Sonar Deployment Mechanism

Pros	Cons
Would keep sonar payload steady Removes need for float	More complications in moment generation Likely heavier than line Complicated mechanical system

10.5 Velocimetry Technique

With ADCPs being very expensive, the velocity calculation of river discharge must still be addressed. However, this left the team with a predicament; if the velocity can't be known as a function of depth then how can discharge be accurately calculated? Luckily, non-contact methods for river discharge have been an active area of research in recent years. The methods primarily revolve around relating the river surface properties to an assumed depth averaged flow velocity to yield the velocity component of discharge [10],[8].

There are three techniques that were considered; the simplest was a optical camera to record the height of the river at a given cross section. The other two fall within what seems to be the direction of industry and use Large Scale Particle Image Velocimetry (LSPIV) to calculate the surface flow velocity. Essentially these work by taking a series of images at a known distance and angle above the water's surface. These images are then post processed where specific points on the waters surface are tracked to find the difference from each frame. The displacement of tracked particles coupled with the camera's viewing angle and distance above the water allows for a velocity calculation to be made across the width of the river [10],[8]. Three LSPIV techniques were analysed; a thermal technique, a stereoscopic technique, and a stereoscopic/structured light RGB camera combo.

Technique	Pros	Cons
Optical	Simple data collection Low power Inexpensive	Large systematic error Low resolution Manual Post-processing
Thermal	Creates time average surface velocity field Software, customer support for post-processing	Expensive Heavy No surface depth information
Stereo	Inexpensive Light weight Creates time averaged surface velocity field Measures surface depth characteristics More accurate for mountain rivers	No native LSPIV software Relatively new technique

10.5.1 Optical Camera

In researching non-contact discharge calculations the team came across a study by the University of Southampton published in Computers Geosciences (see [25]). The purpose of the study was to test the use of cameras for monitoring discharge in a glacial river. The scientists used an older line of the Brinno Time Lapse cameras, the TLC100, to take images of the river. Their choice in cameras was justified by the fact that it was inexpensive and designed to be used for unattended battery operation. The study utilized edge detection software to find image coordinates of the water margins, and combine those coordinates into a flow estimate. The image processing and camera calibration introduced large amounts of systematic error; however, the findings of the study determined that cameras could be used to make meaningful discharge estimates. The measurements taken for one river, however, could not be accurately compared to measurements collected for another river. But collected data could be compared over time for the same river location in order to detect relative changes in discharge.

10.5.1.1 Brinno Time Lapse Camera

The Brinno TLC200 is a lightweight camera that can capture time lapse videos and still photos. The camera has a rotatable lens and can be set to perform unattended. Both of these output formats have a resolution of 1280x720. The entire camera weighs approximately 120g, requires 4 AA batteries, and comes with weatherproof housing. The images would be stored on an SD card during operation. This camera does not provide any depth sensing or image analysis software, so the images collected would require a heavy amount of image processing in order to extrapolate a water velocity estimate. The total cost of this camera is $120.

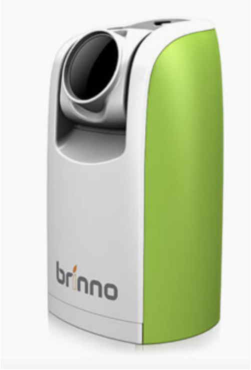


Figure 66: Brinno TLC 200

Pros	Cons
Small	Cannot sense depth
Light weight	Would require edge finding software and heavy image processing
Inexpensive	Resolution is >1Mp
Rotatable lens	Large systematic error
Weather proof housing	
Long battery life	

10.5.2 Thermal Camera

Infrared thermal cameras provide a non-contact temperature measurement using an invisible wavelength that is far right on the electromagnetic spectrum. The wavelengths corresponding to infrared are about 750nm and greater. Infrared energy (IR) is the part of the spectrum that we perceive as heat. Everything with a temperature above absolute zero emits infrared and will be picked up by the camera. The heat energy data taken is converted into an illustration showing heat patterns and is used to create a thermograph. This thermograph is then fed into LSPIV software and different patches of temperature are tracked to yield the velocity of the flow [24].

10.5.2.1 FLIR Vue Pro R

The FLIR Vue Pro R radiometric drone thermal camera makes non-contact temperature measurements with Long Wavelength Infrared (LWIR) techniques using a 6.8mm lens with a 45°(H) x 35°(V) field of view. The sensor resolution of 336 x 256 pixels, and the entire device only weighs 4 ounces. The dimensions of the camera are 2.26" x 1.75", including the lens. It is highly compatible with other devices and can be controlled from a provided app. It operates within the spectral band 7.5μm - 13μm and has an export frame rate of 9Hz or less. This device is physically versatile, having a wide range of operating temperatures and locations and a necessary input voltage of only 4.8 V. There is an abundance of support and additional products for the FLIR brand, as it is the standard for lightweight, mountable thermal imaging devices. There is a zoom capability provided on the user interface. This device costs $3149.

Figure 67: Radiometric Drone Thermal Camera

Pros	Cons
Low mass Ease of integration Highly accurate [24]	Outside of budget Cannot measure surface depth characteristics

10.5.2.2 VuIR Lepton

The VuIR Lepton is one of the cheapest systems available, and it is only recommended for beginners and those doing "simple works". For example, this camera is often used to detect moisture in obscure places. It is not suggested that to use Lepton thermal camera for any professional use where the data must be extremely reliable. The VuIR products are meant to be sold as cheaper versions of the FLIR products, which typically cost thousands of dollars. The software is compatible with android and Black Pearl monitors. The resolution is 160 with a 9Hz rate. The VuIR Lepton model is extremely versatile and easy to mount and dismount from a UAV. The product is prepared for use on many of the typical drones used for thermal imaging and comes with a complete system ready to mount, such as the DJI Mavic 1, Mavic 2, Phantom 4 Pro, Inspire 1, and other drones. The cost of this device is $699.[22]

Figure 68: Drone Thermal Imagine System

Pros	Cons
Inside budget Accessory inclusive package Self Powered Self contained Quick attachment/detachment from UAV High battery life	High mass Medium size Lower resolution Medium range (49m)

10.5.2.3 Seek Micro Core

The Seek Micro Core thermal imaging system is compact yet contains 30,000 temperature pixels. The resolution is 200 x 150, and the size is less than that of a dime. In combination with the Dual-Gain and 12 micron pixel pitch, this makes an accurate, lightweight device that is suitable for UAV applications. The imaging is uninterrupted because it is completely shutter-less. The frame rate is similar to the previous thermal cameras at 9Hz with a low power input. There is excellent image clarity and sensitivity due to the dual-gain smart pixels, which automatically adjusts the gain states of each pixel to maximise contrast on the thermograph output by the camera. The cost of this device is $499.

Figure 69: Shutter-less OEM Thermal Camera

Pros	**Cons**
Low mass Low power Low size Uninterrupted imaging USB interface [23]	High cost No housing included

10.5.3 Stereo Camera

Another technique to collect river flow data would be through use of a stereo camera. Stereo image sensing works both indoors and outdoors and utilizes two cameras to calculate depth. This is done by extracting the differences in image location of an object in a process called triangulation. This is similar to how humans perceive depth through binocular vision.

10.5.3.1 Intel RealSense D455 Camera

The Intel RealSense D455 boasts a very complete and capable sensor package that allows the camera to not just capture HD RGB video at 30fps but also HD depth data at 90fps. This being a crucial and neglected ability compared to the thermal and optical cameras as this depth data coupled with the camera's field of view and resolution is absolutely crucial for obtaining an precise pixel to distance calibration. This coupled with its depth sensing range of 0.4 to 20m, weight of 72g and standard 0.25 inch mounting screw makes the RealSense a strong contender for Velocimetry data acquisition operations. This camera costs $239.

Figure 70: Intel RealSense

[H]

Pros	Cons
Cheap Precise with long range Light weight Good technical documentation Pre-installed mounting hardware	No internal storage Needs to be connected to data logger

10.5.3.2 Stereo Lab ZED 2

The ZED 2 from Stereo Labs is a high resolution state of the art stereosonic camera. It is the first camera to employ neural networks to attempt to reproduce human vision to bring a new level to stereo perception. It has an industry leading 110° FOV as well as thermal management to reduce lens distortion and allow for more light capture. Lastly, it also utilizes AI to detect objects and improve the users spatial awareness. This camera costs $449.

Figure 71: Stereo Lab ZED 2

[H]

Pros	Cons
Wide FOV Live camera control UHD Resolution Thermal control	No internal storage Needs to be connected to data logger High post processing Requirements

10.6 On-Board Computer

In order for all sensors included in the sensor suite to produce meaningful data, they all require some sort of mechanism to store their data. The best way to store all collected data is to send data from all different sensors to one main computer. Having a computer will allow for the storage of large amounts of data as well as keeping a uniform timestamp across all incoming data.

10.6.1 Arduino Mega with SD Module

The Arduino Mega is offered at a low cost and weight. Compared to the single serial port in the Arduino Uno, the Arduino Mega has four serial ports in addition to the larger flash memory capacity that reaches $256kB$. Another important feature is the Integrated Development Environment (IDE) which allows users to interface with the Arduino and upload code through a text editor that is easy to edit and understand. The cost of this device is about $40.

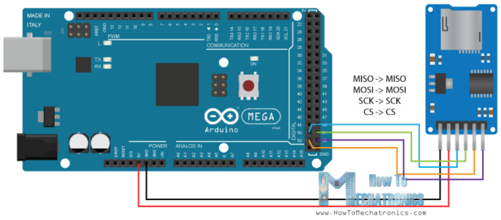


Figure 72: Arduino Uno with SD card attachment

Pros	Cons
Four serial ports	Requires integration with SD card shield
Arduino IDE is easy to understand	Limited by number of data collection ports
Robust	Only one USB port
Simple and easy to use design	
Low average power consumption	

10.6.2 Raspberry Pi 4

The Raspberry Pi 4 is a micro-computer that has four USB ports and a mounted Micro-SD card slot. It is offered at a low price and low weight making it a viable contender for RiBBIT's on-board computer. The Raspberry Pi 4 operates on a Linux system meaning it requires a reasonable understanding of the Linux operating system and how to boot the operating system on the micro-computer. This micro-computer costs about $50.

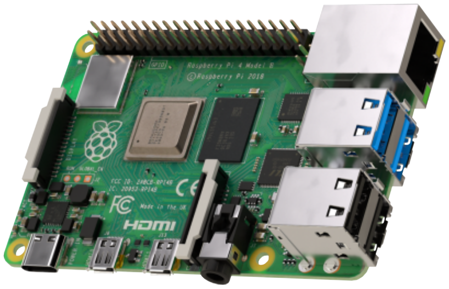


Figure 73: Raspberry Pi 4

Pros	Cons
On-board Micro-SD card slot	Learning curve to understand software
Four USB ports	Easily damaged
	High average power consumption

10.6.3 Odroid XU4

The Odroid XU4 is a micro-computer that has three USB ports and also has a Micro-SD card slot mounted on it. It is slightly more expensive than other listed products which also makes it a viable contender for the on-board computer. Like the Raspberry Pi 4, it operates on Linux. The cost of this micro-computer is $80.

Figure 74: Odroid XU4

Pros	Cons
On-board Micro-SD card slot Three USB ports	Learning curve to understand software Easily damaged Relatively expensive High average power consumption Heavy relative to other options

10.6.4 Arduino Fio with XBee

The Arduino Fio is a micro-controller with one serial port and an XBee socket, allowing it to communicate using an XBee device. It is light and inexpensive; however, it is discontinued which could lead to difficulties so there is no support for this product and the documentation is not up-to-date. The cost of this device is roughly $30.

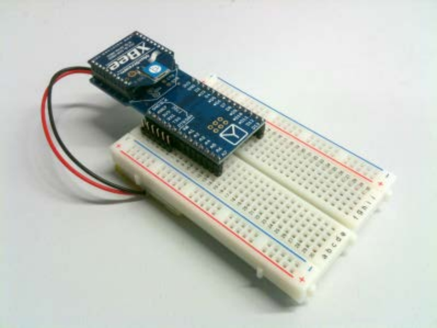


Figure 75: Arduino Fio with XBee attachment

Pros	Cons
Good for communicating with another computer Has XBees socket for communication Simple Arduino interface	Discontinued so documentation not up-to-date Easily damaged

10.7 UAV

One of the most essential elements of this project is selecting the appropriate drone that will be responsible for lifting and towing the instrument suite across the water. Leading drone manufacturers such as DJI and Yuneec provide a variety of drones with unique specifications that meet the consumers' desires. The drones are evaluated by the team based on multiple factors including cost, whether or not it has a built in GPS or camera, its carrying capacity, and its flight software. However, the budget of the project highly limits the drone options. Therefore, the team has the option to either buy their own drone with the specifications that they desire or borrow one from research labs such as The Research and Engineering Center for Unmanned Vehicles (RECUV) or The Integrated Remote and In Situ Sensing (IRISS).

10.7.1 Buy

10.7.1.1 Swellpro

The first option to buy is a drone called SplashDrone 3 manufactured by SwellPro[2]. The drone is shown below in Figure 76. In terms of cost, SplashDrone 3 is a low cost drone priced at $1,200. It includes a night camera for operations in the dark and reports a flight time up to 24 minutes. Additionally, the drone has a maximum wind resistance of 28 km/h. However, the SplashDrone 3 only has a payload capacity of 1 kg which is a lower payload capacity than desired.

Figure 76: SplashDrone 3 by SwellPro

Pros	**Cons**
Low cost	Low carrying capacity
Waterproof	Short flight time relative to carrying capacity
Night camera	No built in GPS available
Wind resistance	No autopilot

10.7.1.2 Tarot 680

The Tarot 680 is a US based drone specifically designed to accommodate payloads[?]. It has a carrying capacity of around 1.5kg which is a bit of a low margin to fit the instrument suite in. Also, the Tarot 680 costs around $2,300 which would take up nearly half of the budget for the team. One thing to keep in mind is that the company does provide discounts for students but that is something that shouldn't be counted on. The Tarot 680 uses the Pix4 flight controller with ArduPilot, which is an open source software and is approved for government use.

Figure 77: Tarot 650 by UAV Systems

This drone comes with the Taranis CX7 Transmitter.

Figure 78: Taranis Q X7 Transmitter

For an additional $1000 USD, the Hex Herelink Ground Station can be added on. The Herelink includes the controller and one HD camera installed on the Tarot 650 or Tarot X6. The camera feed can stream live to the Herelink Transmission System. This upgrade replaces the Taranis QX7 Transmitter with the Herelink transmitter. This addition is desirable if the budget allows as it enables easier drone navigation and obstacle avoidance.

Figure 79: Hex Herelink Ground Station Add-On

Pros	**Cons**
Low cost RTK option Ardupilot/Open source flight software	Low carrying capacity FPV camera is $1000 Lower flight time Wind resistance unknown

10.7.1.3 Tarot X6

The third option that remains in the team's budget is the Tarot X6[4]. The X6 has the same Pixhawk flight controller as the 650 but has a lot more power available. It can lift 5kg for 25 minutes and 35 minutes without a payload. This makes it an ideal candidate for the mission, but it comes with a high price tag of $3300. The Hex Herelink Ground Station can also be added-on on, increasing the price to $4,300. This high cost would require the instrument suite and remaining purchases to be very inexpensive.

Figure 80: Tarot X6 by UAV Systems

Pros	Cons
High carrying capacity	Mid-High Cost
Decent flight time	FPV camera is $1000
Designed for payload	Wind resistance unknown
RTK option	
Ardupilot/Open source flight software	

10.7.2 Borrow

To reduce costs, RiBBIT considers the option to borrow the drone from on-campus laboratories such as RECUV and IRISS. Borrowing a drone would allow more funds to go towards the instrument suite. However, there is only one drone available to borrow which has only half the payload capability of the Tarot X6.

10.7.2.1 DJI S900

Figure 81: S900 by DJI

Pros	Cons
Mid carrying capacity	Can't use whenever desired
Decent flight time	No RTK
Designed for payload	Complicates logistics
Vibration Dampeners	Wind resistance unknown
Ardupilot/Open source flight software	
FPV camera system included	
Low cost	

10.8 Positional Determination and Geo-referencing Technique

While UAV-based remote sensing techniques enable low-cost and quick high resolution data, the data must be validated with geo-referencing techniques to ensure accuracy and reliability. This can be done through direct geo-referencing using on-board position and orientation data or indirect geo-referencing using Ground Control Points (GCP)[10]. The different options for capturing accurate data are laying Ground Control Points, Real-Time Kinematic and Post-Processing Kinematic methods. In Figure 82, the three methods can be visualized. To address the requirement necessitating the system's ability to access difficult locations, the ground control point option was not prioritized. However, GCPs could be collected in outdoor testing to further validate the data.

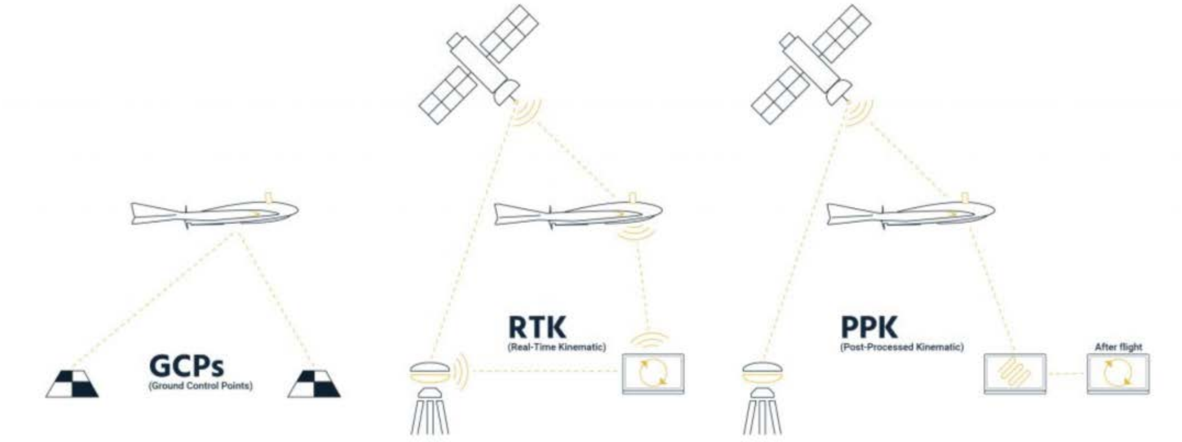


Figure 82: Different Geo-Referencing Methods

10.8.1 U-blox C94-M8P RTK Package

The $399 U-blox C94-M8P RTK Application Board Package contains two antennas and receiver modules with NEO-M8P high precision GNSS modules, each of which may act as either a base station or rover. This package is able to offer up to 2.5cm accuracy using Real-Time Kinematic positioning.

Figure 83: U-blox NEO-M8P

Pros	Cons
Included antennas Integrated chips	$399 cost

10.8.2 Custom Sensor Fusion

One option is to purchase separate IMU and GPS modules and fuse their data together ourselves. This option allows us to obtain accurate sensors without the overhead cost of pre-programmed components. The standalone $220 Sparkfun GPS board is capable of providing positional accuracy up to 2.5m. The $25 Arduino IMU is capable of providing orientation corrections, but the fusion of GPS and IMU data presents an unknown time cost of software development.

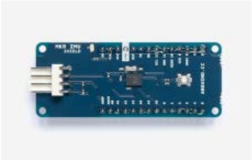


Figure 84: Micro-controller compatible IMU

Figure 85: Micro-controller compatible GNSS

Pros	Cons
Parts cost	Unknown time cost of software implementation Inaccuracy of standard positioning service

10.8.3 Emlid Base Station RTK/PPK Configuration

The Emlid RS2 RTK GNSS Receiver is a ground based system used for surveying, mapping and navigation. The cost of this product is $1899, however, it can be borrowed for no cost from the University NAVSTAR Consortium (UNAVCO) located in Boulder. This base station's 1cm accuracy, allows a larger budget for a comparably accurate on-board UAV receiver.

There are multiple options for the on-board GNSS receiver. Two leading contenders being the Aceinna OpenRTK330 ($150 chip and up to 3cm accuracy) and the U-blox C94-M8P. Their key difference is the U-blox package's contains a pre-integrated chip and antennas.

Both pieces of hardware have built-in sensor fusion algorithms that leave room for customization. These devices allow the use of either Real-Time Kinematic (RTK) or Post-Processed Kinematic (PPK) localization methods.

RTK and PPK are different methods of correcting the location of a drone. For our purposes, they each require the same hardware, but differ in their respective implementations. They each require a GNSS capable receiver ground station and drone receiver, and both require those devices to receive GNSS communications throughout the mission. There difference lies in additional required connections and when the data is processed. RTK requires the base station and drone to maintain constant communication with a drone base station, and the data is processed real–time. PPK does not require these extra signals, and thus the data must be post–processed after the flight. As these implementation do not require distinct hardware configurations, they shall be grouped together as a single key design option, which, if selected, can be used in either arrangement.

Figure 86: Emlid RS2 GNSS Receiver

Pros	**Cons**
Can rent from UNAVCO Proven accuracy RTK and PPK options	Need to coordinate borrowing logistics

Criteria	1	2	3	4	5
Cost	>$3000	3000-2000	2000-1000	1000-500	<500
Mass	>5 kg	5kg-3kg	3kg-1kg	1kg-0.5kg	<0.5kg
Size	>20*$10^{6}mm^{3}$	20-10*$10^{6}mm^{3}$	10-1*$10^{6}mm^{3}$	10-1*$10^{5}mm^{3}$	<1*$10^{5}mm^{3}$
Power	>40 W	40W-30W	30W-15W	15W-5W	<5W
Integration	1	2	3	4	5
Data Handling	1	2	3	4	5
Minimum Depth	>2m	2m-1.5m	1.5m-1m	1m-0.5m	<0.5m
Maximum Depth	<1m	1m-3m	3m-10m	10m-20m	<20m
Accuracy	>5%	5%-3%	3%-1%	1%-0.5%	<0.5%
Water Performance		considerable		negligible factor	
Water Contact	Contact required				non-contact

Table 7: Bathymetry Criteria Definitions

10.9 Trade Studies

10.9.1 Bathymetry Technique

Criteria	Weight	Rationale
Cost	20%	Instrument cost is a limiting factor. Sensors can be expensive, and there needs to be enough budget to cover drone and manufacturing.
Mass	10%	The instrument needs to be drone/UAV mountable. The weight of the instruments will affect drone selection.
Size	5%	The instrument's footprint must be small enough to be mounted on a drone with minimal issue.
Power	5%	The power source required to function may affect the weight of the sensor suite as well as how easily it can be flown on a drone.
Integration	5%	The sensor suite will need to be mounted onto the drone and should not take an excessive amount of additional hardware to do so.
Data Handling	10%	The ease of which data can be of loaded in a use full format to local storage and then later analysed is central to the success of this project since getting meaning full scientific conclusions is the objective. The ease of this is evaluated based on the instrument connector types, the instrument data format, storage compatibility and software support.
Minimum Depth	7.5%	The sensor must be able to take measurements in shallow waters.
Maximum Depth	7.5%	The sensor must be able to take measurements at least 3m deep.
Accuracy	20%	The sensors should be able to collect accurate data in order to meet the science requirements and provide meaningful results.
Water Performance	5%	The sensor should be able to perform well in clear to mildly dirty water since the water quality of rivers can vary for different locations.
Water Contact	5%	A non-contact sensor would be ideal as it does not require a sensor package hanging below the UAS

Criteria	**Weights**	**LiDAR**	**Sonar**	**ADCP**
Cost	0.2	1	5	1
Mass	0.1	1	5	2
Size	0.05	1	5	3
Power	0.05	2	5	4
Integration	0.05	4	4	4
Data Handling	0.1	3	5	5
Minimum Depth	0.075	3	4	5
Maximum Depth	0.075	5	5	5
Accuracy	0.2	5	4	3
Water Performance	0.05	2	5	5
Water Contact	0.05	5	1	1
Total	1	**2.9**	**4.475**	**3.1**

Table 8: Bathymetry Criteria Levels

From this it became abundantly clear that SONAR is the best depth measuring option available to the group given the constraints and requirements that the team is operating on. For this instrument the group needed one that is small, light, accurate, easy to work with both in regards to power and data, can perform in sub optimal water conditions and finally would ideally be non-contact. All but one of these requirements were met by sonar and because of this it became the clear winner and first choice.
That being said, it now becomes important to down select the final unit to be implements by the group from the two choices outlined above. To do this another trade study was conducted with the two units using the same criteria and weights. This being shown below.

Criteria	**Weights**	**Ping SONAR**	**SICK UM30**
Cost	0.2	5	5
Mass	0.1	5	5
Size	0.05	5	5
Power	0.05	5	5
Integration	0.05	4	3
Data Handling	0.1	5	3
Minimum Depth	0.075	4	5
Maximum Depth	0.075	5	1
Accuracy	0.2	4	5
Water Performance	0.05	5	4
Water Contact	0.05	1	1
Total	1	**4.475**	**4.15**

Table 9: Bathymetry Trade Study

Given this it becomes clear that while both sensors may be good options the Ping echo sounder by BlueRobotics is the better choice and will be the instrument the group will use going forward.

10.9.2 SONAR Float Angular Displacement Correction Technique

Criteria	**Weight**	**Rational**
Cost	15%	Cost is always a driving factor in a project but given the relatively low cost of these specific systems it is values less in this case.
Mass	20%	Given that this system has to be carried by the UAS it is of the up most importance the its weight be minimised so it has a minimal impact on UAS selection and flight time.
Integration complexity	30%	The solution should be as simple to integrate as possible as creating specialty mounting solutions or having it negatively influence the function of the overall system is to be avoided.
Effectiveness	35%	Given that the collecting valid and useful data is the driving motivation in this project it was decided to place a high amount of importance on this criteria.

Table 10: SFADCT Criteria Definitions

Criteria	**1**	**2**	**3**	**4**	**5**
Cost	>$80	80-60	60-40	40-20	<20
Mass	>100 g	100g-70g	70g-30g	30g-10g	<10g
Integration	1	2	3	4	5
Effectiveness	1	2	3	4	5

Table 11: SFADCT Criteria Levels

Criteria	**Weights**	**Gyroscope**	**Weighted Float**	**Gimbal**
Cost	0.15	5	5	1
Mass	0.2	5	2	1
Integration Complexity	0.3	5	3	3
Effectiveness	0.05	5	3	5
Total	1	**4.7**	**3.1**	**2**

Table 12: SFADCT Trade Study

Thus we can see that based on the sub-systems mass, cost, integration complexity and effectiveness the gyroscope is the winner and will be the instrument the group will use to correct SONAR data moving forward.

10.9.3 Velocity Technique

Criteria	**Weight**	**Rationale**
Cost	20%	Instrument cost is a limiting factor. Sensors can be expensive, and there needs to be enough budget to cover other essential aspects of the project.
Weight	5%	The instrument needs to be drone/UAV mountable. The weight of the instruments will affect drone selection due to payload contrains and required flight time. The weight will also effect the center of mass and other flight characteristics and thus a light instrument is preferred.
Size	5%	The size of the instrument will effect how difficult it is to mount to the drone as well as effect the drag of the drone.
Power Requirements	5%	An instrument that has high power requirements may require additional batteries adding weight or limit the data collection period if extra batteries cannot be afforded.
Integration complexity	5%	An instrument that requires communication with other on-board systems will not only create additional points of failure, but also present the team with additional challenges and constrains (such as making sure the other onboard systems have common programming languages to the instrument).
Velocity Calculation Complexity	15%	The instruments investigated came with varying levels of COTS post-processing software. The easier the post-processing is the more time and effort this will save later in the project
Altitude Measurement	5%	It is advantageous to know the drones height above the water while collecting depth measurements and enables the water height of the river to be determined with lower uncertainty than would be enabled with GPS.
Max Field of View	15%	This has implications for how high the drone will need to be to above the river for data collection but also the range of rivers possible to survey. A larger FOV also means the camera can be closer to the river, and since accuracy is a function of distance also means the camera will likely be able to obtain more accurate results.
Frame Rate	10%	Frame rate will be set the upper limit on how fast of flow can be measured as well as how accurate the velocity calculation can be.
Data collection Resolution	15%	The ability of post-processing to discretize the flow and yield velocities that are more representative of true values will be dictated by the resolution of the collected data.

Table 13: Velocity Criteria Definitions

To compare techniques a 'middle of the road' product from the instruments presented in the key-design section were used. The raw value selection for the qualitative criteria was guided by the score guides shown below.

Criteria	**1**	**2**	**3**	**4**	**5**
Cost	>$3000	3000-2000	2000-1000	1000-500	<500
Mass	>5 kg	5kg-3kg	3kg-1kg	1kg-0.5kg	<0.5kg
Size [mm^3]	>20*10^6	20-10*10^6	10-1*10^6	10-1*10^5	<1*10^5
Power	>40 W	40W-30W	30W-15W	15W-5W	<5W
Integration	Many interfaces		Few interfaces		No interfaces
Velocity Calculation	No COTS		3rd party COTS		Native COTS
Alt Measurement	No Alt				Alt
Max FOV [deg]	<55	55-65	66-75	76-85	85<
Frame rate [FPS]	<25	25-45	46-65	66-85	85<
Data Resolution	<480p	480-720p	720-1080p	1080p-2k	<2k

Criteria	**Weights**	**Optical**	**Thermal**	**Stereo**
Cost	0.2	5	1	5
Mass	0.1	5	5	5
Size	0.05	4	4	5
Power Requirements	0.05	3	5	5
Frame Rate	0.05	1	2	5
Ease of Integration	0.05	5	4	4
Velocity Calculation Complexity	0.15	2	4	3
Altitude Measurement	0.05	1	1	5
Max Field of View	0.15	2	1	5
Data Collection Resolution	0.15	2	1	2
Total		2.9	2.25	**4.2**

Table 14: Velocity Technique Trade Study

Thus it can be seen that the stereo camera is the winning technique and given the affordability of the stereo cameras, this will be the technique to be used to measure flow velocity. But, that still leaves the specific stereo instrument to be determined. Below is the trade study to determine which type of stereo camera project RiBBIT will employ.

Criteria	**Weights**	**Intel RealSense D455**	**Stereo Lab ZED 2**
Cost	0.2	5	5
Mass	0.05	5	5
Size	0.05	5	4
Power Requirements	0.05	5	5
Frame Rate	0.1	5	5
Ease of Integration	0.05	4	4
Velocity Calculation Complexity	0.15	3	3
Altitude Measurement	0.05	5	5
Max Field of View	0.15	5	5
Data Collection Resolution	0.15	2	5
Total		4.2	**4.6**

Table 15: Stereo Camera Trade Study

10.9.4 On-Board Computer

The on-board computer will be used to store data collected from the sensor suite and to send commands that start and stop data collections on sensor suite instruments. As all candidates for the on-board computer are relatively inexpensive and lightweight, these criteria in down-selecting a computer are given little weight, as they do effect the project as a whole, but not greatly. The computer power consumption is a main criteria because the mission operating time is great enough such that computers with different power requirements will greatly effect the sizing of the battery, which could lead to overall increased payload weight. It is also very important that the computer's hardware can interface with all necessary sensors in the suite. If the computer's hardware is not compatible with all sensor hardware, then the computer cannot achieve it's main objective of storing data collected from the sensor suite.

Criteria	**Weight**	**Rationale**
Cost	10%	We need to minimize the amount of money spent on a computer so we can better afford more expensive instruments. However, a computer won't break the bank so we shouldn't sacrifice cost for quality.
Weight	10%	The drone will have a limited payload capacity and most of that is reserved for larger sensors. Computers are generally very light but ideally weight will be minimized.
Power	30%	Throughout the duration of the mission the computer will be drawing power from the battery. We want to minimize the power consumed so that we can minimize the size of the battery.
Compatibility	30%	The chosen computer needs to have the ability to interface with the hardware for all chosen sensors.
Programmability	20%	The computer must be able to efficiently program the mission and store necessary data. A simplistic IDE is preferred as it will save a lot of time.

Criteria	**1**	**2**	**3**	**4**	**5**
Cost	> $100	$80 - $100	$60-$80	$40-$60	< $40
Weight	> 100g	75g-100g	60g-75g	50g-60g	< 50g
Power	> 4W	2-4W	1W-2W	0.5W-1W	< 0.5W
Compatibility	Cannot interface with any sensor	Can interface with one sensor			Can interface with all sensors
Programmability	Not programmable		Steep learning curve	Slight learning curve	No learning curve

Table 16: On-board Computer Criteria Levels

Criteria	**Weights**	**Arduino Mega**	**Raspberry Pi 4**	**Odroid XU4**	**Arduino Fio**
Cost	0.1	4	5	4	4
Weight	0.1	5	5	4	4
Power	0.3	5	2	1	5
Compatibility	0.3	5	5	5	5
Programmability	0.2	4	3	3	4
Weighted Total	1	4.7	3.7	3.2	4.6

Table 17: On-board Computer Trade study

From the results of the trade study, the Arduino Mega will be the on-board computer. Arduino's are robust and simple to use compared to other computers in the trade study. The Arduino Mega is a great choice because it is very versatile in its capabilities. While the Arduino Fio is a close second, it may be considered as a secondary computer that can be used to communicate data from the deployed sonar unit to the main (on-board) computer, the Arduino Mega.

10.9.5 UAV

Criteria	**Weight**	**Rational**
Cost	30%	The cost is the most limiting factor, if the team were to buy an expensive drone there would be less resources for the instruments.
Payload	20%	The drone must be capable of carrying the instrument suite, also having extra payload capabilities can increase the flight time.
Logistics	20%	Having to workout when and where the drone can be flown complicates the testing process. This can lead to delays but it won't be mission ending.
Flight Controller	10%	The flight controller is a limiting factor when dealing with government agencies. So if the drone is unusable with its default controller that would complicate things for the user.
Flight Time	10%	The minimum flight time is relatively easy to achieve, but additional flight time is very beneficial to users of the instruments suite.
FPV Camera	10%	An FPV camera is useful for piloting a drone from a distance and being able to keep an eye out for debris in the river. It also significantly improves safety of the drone. But isn't required

Table 18: Drone Criteria Definitions

Criteria	**1**	**2**	**3**	**4**	**5**
Cost	>$4000	$4000-$3000	$3000-$2000	$2000-$1000	$1000-$0
Payload	1kg		1kg-2kg	2kg-4kg	>4kg
Logistics	Need to schedule pilot and drone		Need Pilot		Don't need pilot
Flight Controller		Default banned by government			Can be used by government
Flight time	<12min	12min-16min	16min-20min	20min-25min	25min-30min
FPV Camera	No				Yes

Table 19: Drone Criteria Levels

The selection of a drone depends entirely on our budget, if it is too expensive it won't even be possible to buy. So, the cost is weighted at 30% where anything above $4000 was heavily penalized. The obviously preferred option is the most powerful drone available, this generally increases flight duration and the margin of safety in operating in adverse conditions. With that, a drone that can't lift the minimum payload would heavily penalized, but a drone that heavily exceeds the minimum would be preferred. Logistics are important for the testing and integration phase, having to schedule a pilot and/or drone are the main logistical issues.

Criteria	**Weights**	**Swellpro**	**Tarot 650**	**650 w/FPV**	**Tarot X6**	**X6 w/FPV**	**DJI S900**
Cost	0.3	5	3	2	2	1	5
Payload	0.2	1	3	3	5	5	4
Logistics	0.2	5	5	5	3	3	1
Flight Controller	0.1	5	5	5	5	5	2
Flight Time	0.1	1	2	2	5	5	3
FPV Camera	0.1	5	1	5	1	5	5
Total	1	**3.8**	**3.9**	**3.7**	**3.6**	**3.4**	**3.5**

Table 20: Drone Trade Study

10.9.6 Positional Determination and Georeferencing Technique

Criteria	**Weight**	**Rationale**
Cost	10%	Cost should not prohibitively bar sensors capable of delivering necessary performance given the critical role of positional determination.
Weight	20%	Given the limited carrying capacity of drones, it is essential to maintain the lowest possible weight profile where possible to allow for maximum flexibility when selecting inevitably heavier components.
Horizontal Position Accuracy	15%	Accurate localization is of paramount importance, as positional awareness allows the bathymetry measurements to be accurately placed in relation to each other.
Vertical Position Accuracy	15%	Accurate localization is of paramount importance, as positional awareness allows the bathymetry measurements to be accurately placed in relation to each other.
Software Complexity	20%	A proper full scale implementation of a sensor fusion algorithm such as the Kalman Filter exceeds our skill and time budgets.
Hardware Complexity	20%	A localization unit requiring an extensive network of auxiliary devices should be strongly discouraged, given the limited spacial and skill budgets.

Table 21: PDG Criteria Definitions

Criteria	**1**	**2**	**3**	**4**	**5**
Cost	>$400	$350–$400	$300–$350	$250–$300	< $250
Weight	>100g	75g–100g	50g–75g	25g–50g	< 25g
Horiz. Positional Accuracy	>5cm	4cm–5cm	3cm–4cm	2cm–3cm	< 2cm
Vert. Positional Accuracy	>6cm	5cm–6cm	4cm–5cm	3cm–4cm	< 3cm
Software Complexity	Requires custom sensor fusion implementation		Requires software modification		Provided API
Hardware Complexity	Requires extensive supporting hardware				Integrated with existing hardware

Table 22: PDG Criteria Levels

Criteria	**Weights**	**U-Blox C94-M8P**	**Custom Sensor Fusion**	**Emild RS2 + Aceinna**	**Emild RS2 + U-Blox**
Cost	0.1	2	3	5	2
Weight	0.2	4	4	5	4
Horiz Positional Accuracy	0.15	4	1	4	5
Vert Positional Accuracy	0.15	4	1	5	5
Software Complexity	0.2	4	1	2	4
Hardware Complexity	0.2	4	2	1	4

Table 23: PDG Trade Study

10.9.7 Deployment Mechanism

10.10 SI-LSPIV Velocity Components

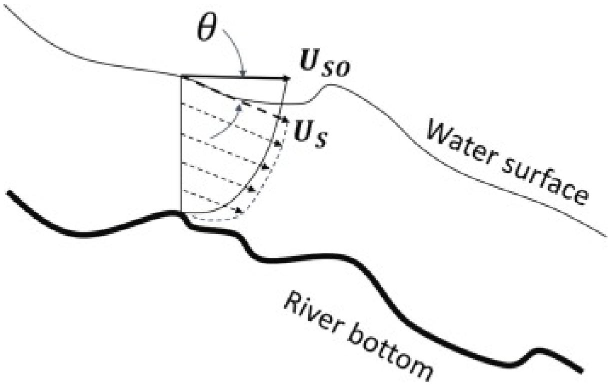


Figure 87: Schematic of velocity components of non-uniform water surface

Figure 87 from [8] shows the different components of flow velocity that can be measured from SI-LSPIV. Level 1 and 2 success of Streamflow measurements use U_{so}, the horizontal component, and level 3 will attempt to calculate U_s which is the true flow velocity.

10.11 Quantities Relevant to Boat Hydrodynamics

The following visualizations can be found in Molland [26].

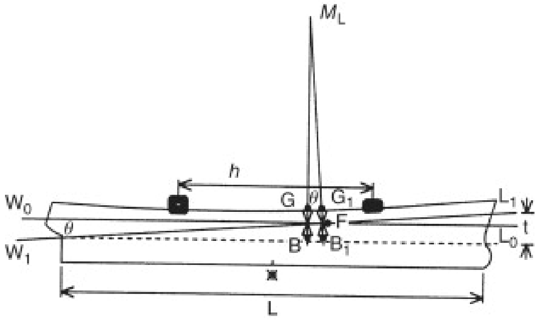


Figure 88: Longitudinal Boat Motions

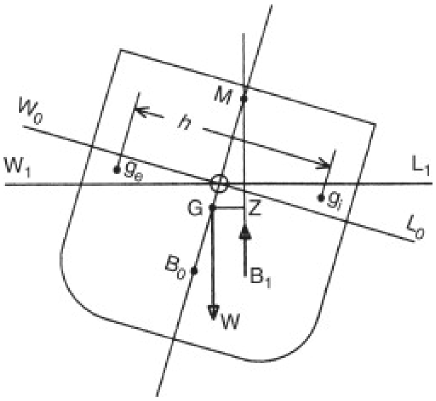


Figure 89: Lateral Boat Motions

10.12 Float Heeling Analysis

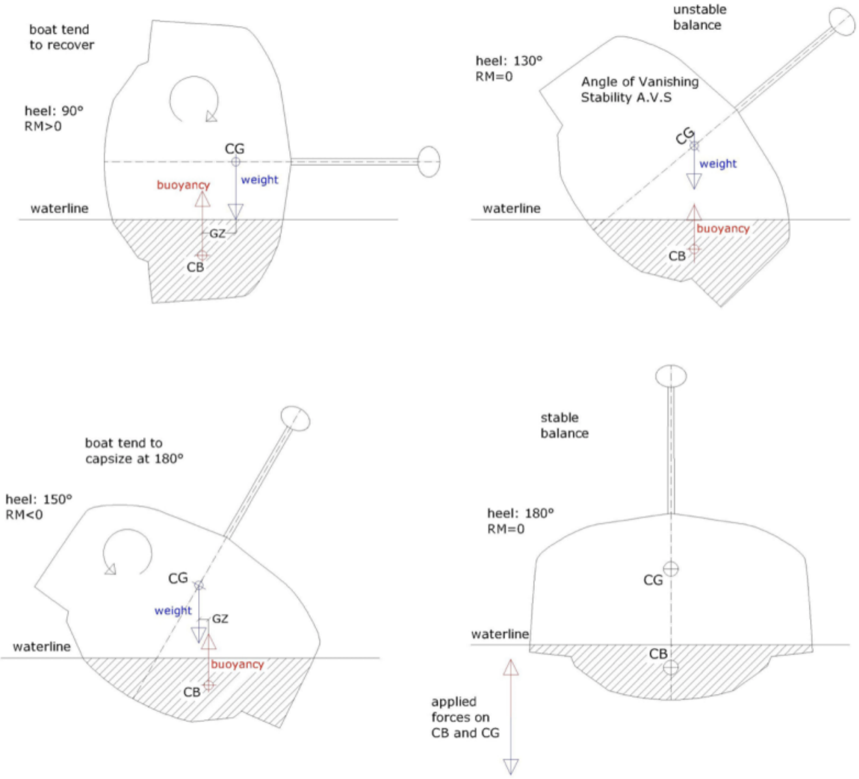


Figure 90: Typical behavior of a Boat Under Heeling

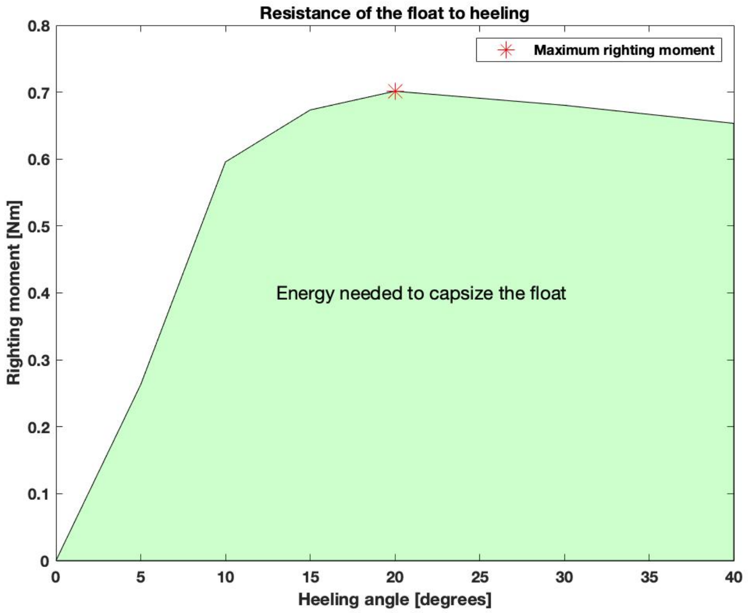


Figure 91: Heeling Analysis for the Selected Float Design

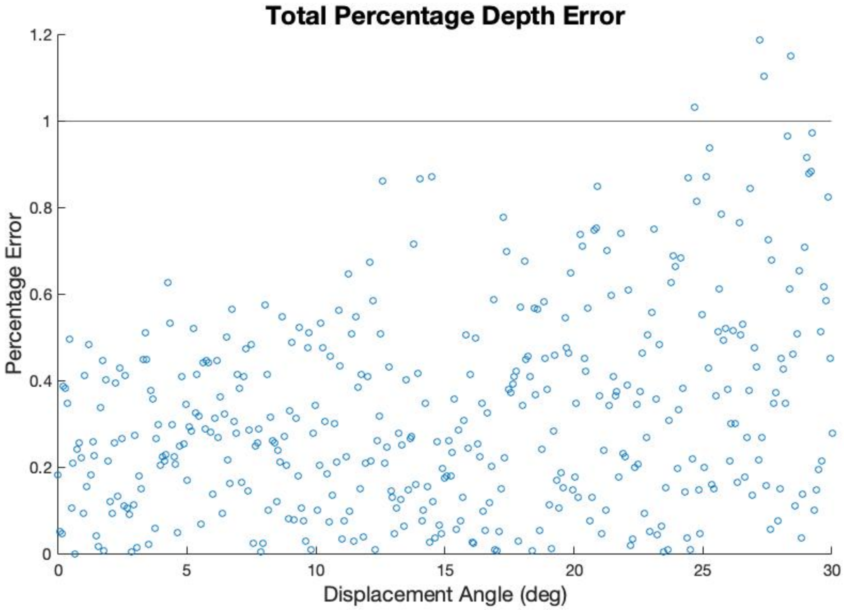


Figure 92: Error Associated Angular Displacement of SONAR Float

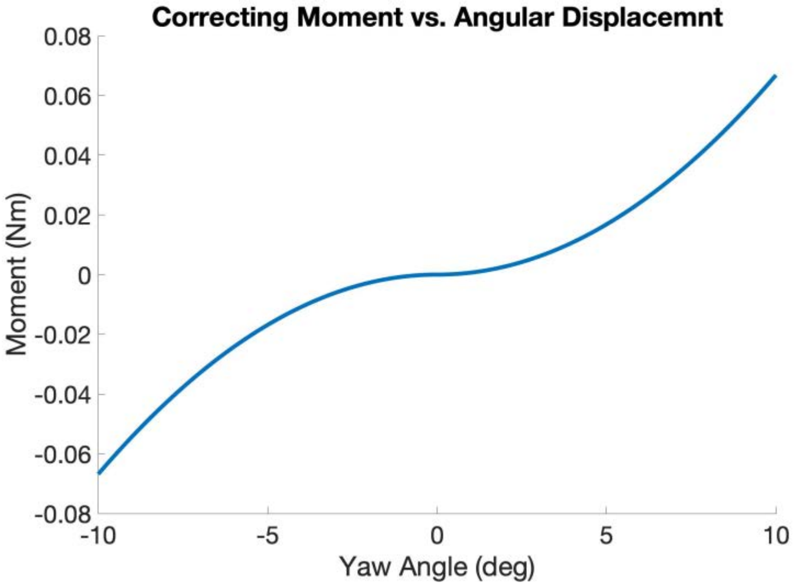


Figure 93: Corrective Moment Produced By Yaw Displacement of Float

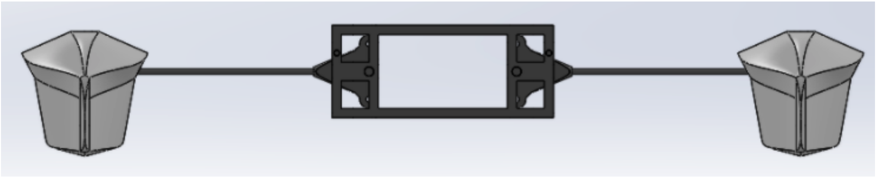


Figure 94: Folding Stabiliser Assembly

10.13 Electronics Box

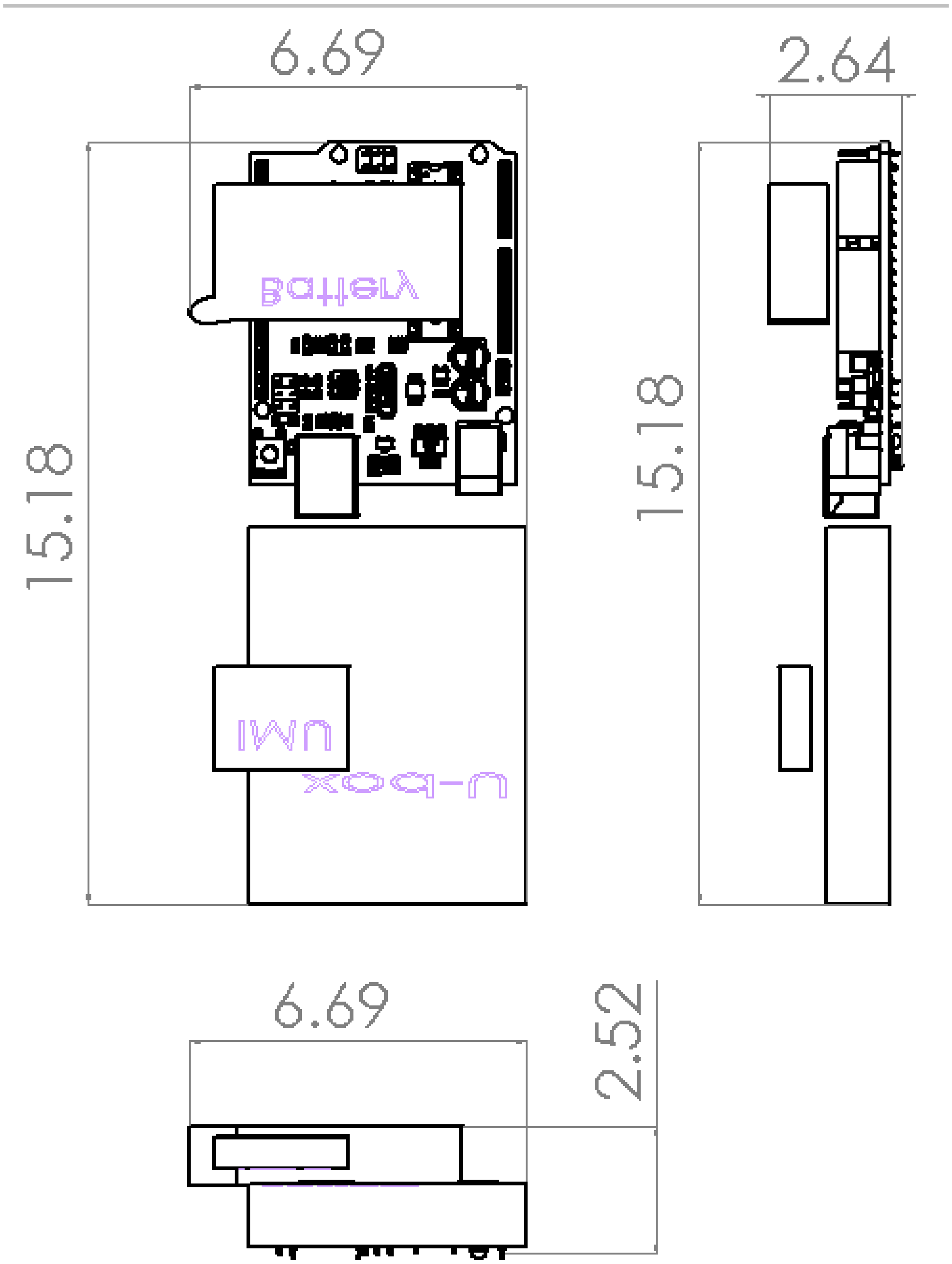


Figure 95: Electronics box potential layout, measurements in CM

10.14 Bow Design

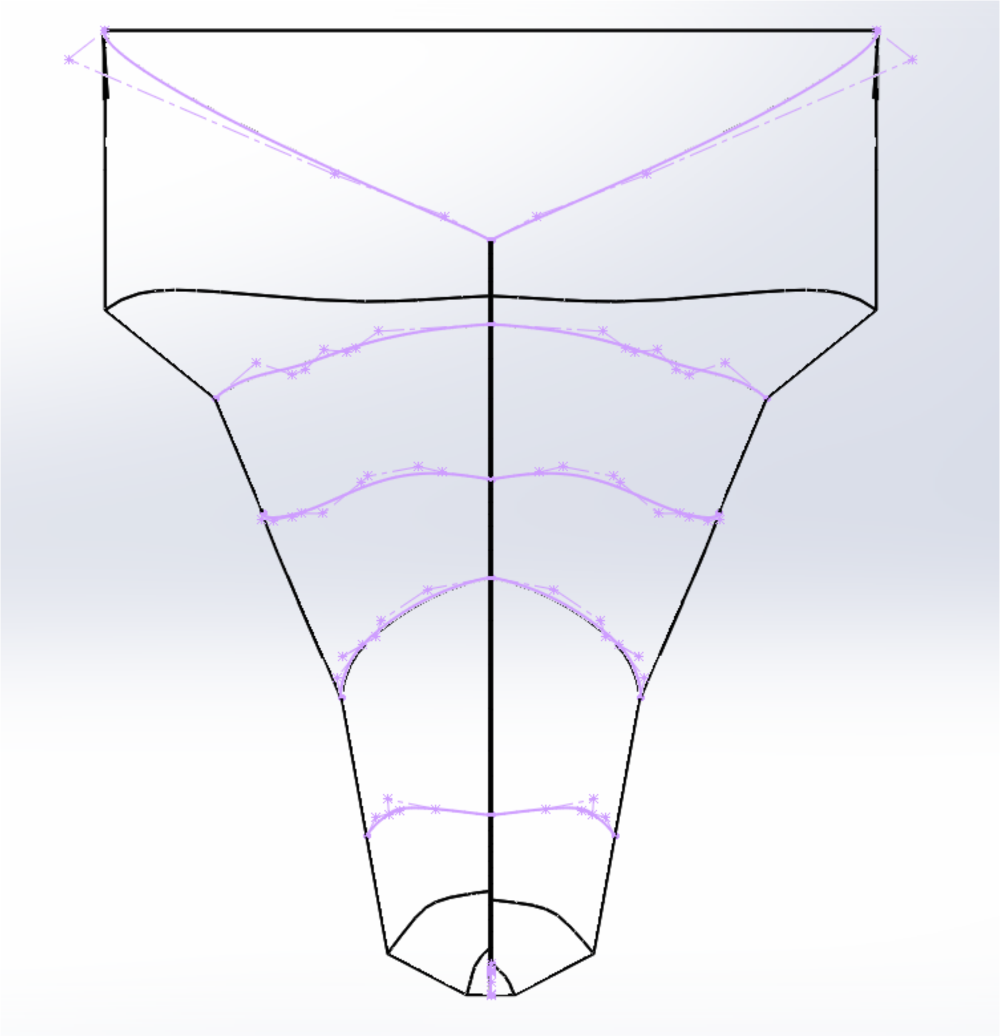
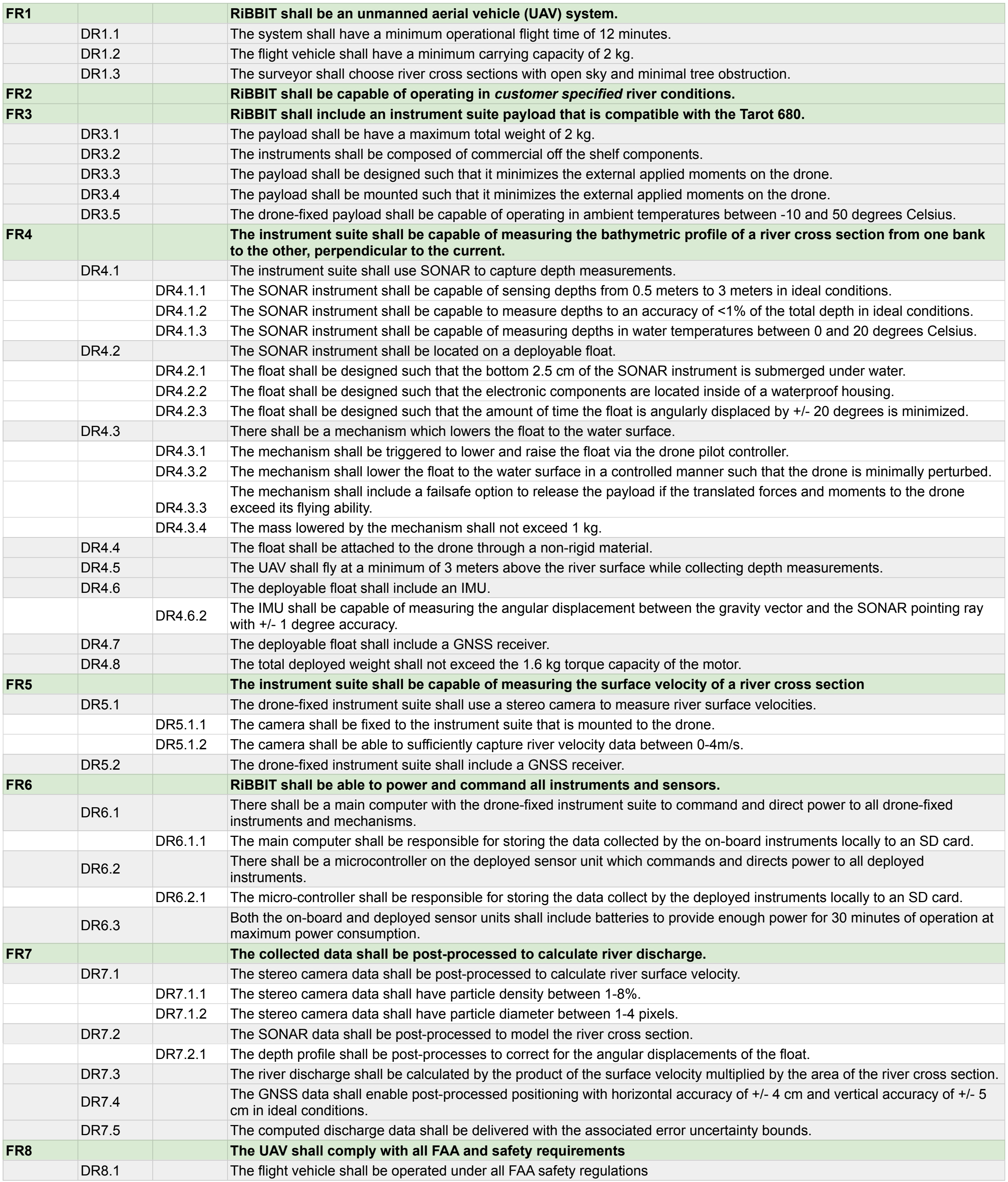
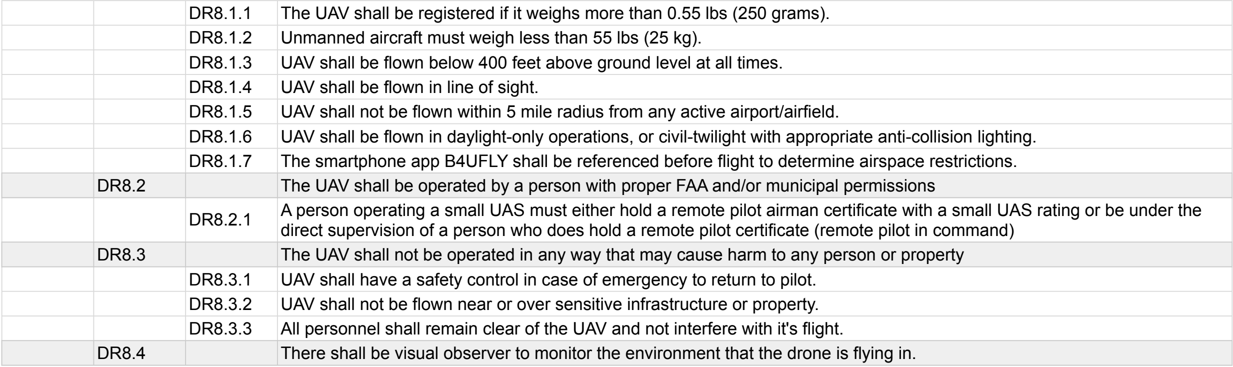


Figure 96: 3D Splines Used in the Bow Design

10.15 Functional & Design Requirements



FR1			**RiBBIT shall be an unmanned aerial vehicle (UAV) system.**
	DR1.1		The system shall have a minimum operational flight time of 12 minutes.
	DR1.2		The flight vehicle shall have a minimum carrying capacity of 2 kg.
	DR1.3		The surveyor shall choose river cross sections with open sky and minimal tree obstruction.
FR2			**RiBBIT shall be capable of operating in *customer specified* river conditions.**
FR3			**RiBBIT shall include an instrument suite payload that is compatible with the Tarot 680.**
	DR3.1		The payload shall be have a maximum total weight of 2 kg.
	DR3.2		The instruments shall be composed of commercial off the shelf components.
	DR3.3		The payload shall be designed such that it minimizes the external applied moments on the drone.
	DR3.4		The payload shall be mounted such that it minimizes the external applied moments on the drone.
	DR3.5		The drone-fixed payload shall be capable of operating in ambient temperatures between -10 and 50 degrees Celsius.
FR4			**The instrument suite shall be capable of measuring the bathymetric profile of a river cross section from one bank to the other, perpendicular to the current.**
	DR4.1		The instrument suite shall use SONAR to capture depth measurements.
		DR4.1.1	The SONAR instrument shall be capable of sensing depths from 0.5 meters to 3 meters in ideal conditions.
		DR4.1.2	The SONAR instrument shall be capable to measure depths to an accuracy of <1% of the total depth in ideal conditions.
		DR4.1.3	The SONAR instrument shall be capable of measuring depths in water temperatures between 0 and 20 degrees Celsius.
	DR4.2		The SONAR instrument shall be located on a deployable float.
		DR4.2.1	The float shall be designed such that the bottom 2.5 cm of the SONAR instrument is submerged under water.
		DR4.2.2	The float shall be designed such that the electronic components are located inside of a waterproof housing.
		DR4.2.3	The float shall be designed such that the amount of time the float is angularly displaced by +/- 20 degrees is minimized.
	DR4.3		There shall be a mechanism which lowers the float to the water surface.
		DR4.3.1	The mechanism shall be triggered to lower and raise the float via the drone pilot controller.
		DR4.3.2	The mechanism shall lower the float to the water surface in a controlled manner such that the drone is minimally perturbed.
		DR4.3.3	The mechanism shall include a failsafe option to release the payload if the translated forces and moments to the drone exceed its flying ability.
		DR4.3.4	The mass lowered by the mechanism shall not exceed 1 kg.
	DR4.4		The float shall be attached to the drone through a non-rigid material.
	DR4.5		The UAV shall fly at a minimum of 3 meters above the river surface while collecting depth measurements.
	DR4.6		The deployable float shall include an IMU.
		DR4.6.2	The IMU shall be capable of measuring the angular displacement between the gravity vector and the SONAR pointing ray with +/- 1 degree accuracy.
	DR4.7		The deployable float shall include a GNSS receiver.
	DR4.8		The total deployed weight shall not exceed the 1.6 kg torque capacity of the motor.
FR5			**The instrument suite shall be capable of measuring the surface velocity of a river cross section**
	DR5.1		The drone-fixed instrument suite shall use a stereo camera to measure river surface velocities.
		DR5.1.1	The camera shall be fixed to the instrument suite that is mounted to the drone.
		DR5.1.2	The camera shall be able to sufficiently capture river velocity data between 0-4m/s.
	DR5.2		The drone-fixed instrument suite shall include a GNSS receiver.
FR6			**RiBBIT shall be able to power and command all instruments and sensors.**
	DR6.1		There shall be a main computer with the drone-fixed instrument suite to command and direct power to all drone-fixed instruments and mechanisms.
		DR6.1.1	The main computer shall be responsible for storing the data collected by the on-board instruments locally to an SD card.
	DR6.2		There shall be a microcontroller on the deployed sensor unit which commands and directs power to all deployed instruments.
		DR6.2.1	The micro-controller shall be responsible for storing the data collect by the deployed instruments locally to an SD card.
	DR6.3		Both the on-board and deployed sensor units shall include batteries to provide enough power for 30 minutes of operation at maximum power consumption.
FR7			**The collected data shall be post-processed to calculate river discharge.**
	DR7.1		The stereo camera data shall be post-processed to calculate river surface velocity.
		DR7.1.1	The stereo camera data shall have particle density between 1-8%.
		DR7.1.2	The stereo camera data shall have particle diameter between 1-4 pixels.
	DR7.2		The SONAR data shall be post-processed to model the river cross section.
		DR7.2.1	The depth profile shall be post-processes to correct for the angular displacements of the float.
	DR7.3		The river discharge shall be calculated by the product of the surface velocity multiplied by the area of the river cross section.
	DR7.4		The GNSS data shall enable post-processed positioning with horizontal accuracy of +/- 4 cm and vertical accuracy of +/- 5 cm in ideal conditions.
	DR7.5		The computed discharge data shall be delivered with the associated error uncertainty bounds.
FR8			**The UAV shall comply with all FAA and safety requirements**
	DR8.1		The flight vehicle shall be operated under all FAA safety regulations



		DR8.1.1	The UAV shall be registered if it weighs more than 0.55 lbs (250 grams).
		DR8.1.2	Unmanned aircraft must weigh less than 55 lbs (25 kg).
		DR8.1.3	UAV shall be flown below 400 feet above ground level at all times.
		DR8.1.4	UAV shall be flown in line of sight.
		DR8.1.5	UAV shall not be flown within 5 mile radius from any active airport/airfield.
		DR8.1.6	UAV shall be flown in daylight-only operations, or civil-twilight with appropriate anti-collision lighting.
		DR8.1.7	The smartphone app B4UFLY shall be referenced before flight to determine airspace restrictions.
	DR8.2		The UAV shall be operated by a person with proper FAA and/or municipal permissions
		DR8.2.1	A person operating a small UAS must either hold a remote pilot airman certificate with a small UAS rating or be under the direct supervision of a person who does hold a remote pilot certificate (remote pilot in command)
	DR8.3		The UAV shall not be operated in any way that may cause harm to any person or property
		DR8.3.1	UAV shall have a safety control in case of emergency to return to pilot.
		DR8.3.2	UAV shall not be flown near or over sensitive infrastructure or property.
		DR8.3.3	All personnel shall remain clear of the UAV and not interfere with it's flight.
	DR8.4		There shall be visual observer to monitor the environment that the drone is flying in.

10. References

[1] Durand, M., Gleason, C. J., Garambois, P. A., Bjerklie, D., Smith, L. C., Roux, H., Rodriguez, E., Bates, P. D., Pavelsky, T. M., Monnier, J., Chen, X., Baldassarre, G. D., Fiset, J.-M., Flipo, N., Frasson, R. P. D. M., Fulton, J., Goutal, N., Hossain, F., Humphries, E., Minear, J. T., Mukolwe, M. M., Neal, J. C., Ricci, S., Sanders, B. F., Schumann, G., Schubert, J. E., and Vilmin, L., "An intercomparison of remote sensing river discharge estimation algorithms from measurements of river height, width, and slope," *Water Resources Research*, vol. 52, 2016, pp. 4527–4549.

[2] Ketchum, K., amp; Wellbots. (n.d.). Swellpro Splash Drone 3+ Waterproof Drone. Retrieved September 29, 2020, from https://www.wellbots.com/products/swellpro-splash-drone-3-plus-waterproof-drone?variant=23115155046448

[3] P-Themes. (n.d.). Tarot 650 V2.1 Ready To Fly. Retrieved September 29, 2020, from https://uavsystemsinternational.com/collections/all/products/tarot-650-ready-to-fly-drone?pos=2

[4] P-Themes. (n.d.). Tarot X6 V2.2 Ready To Fly. Retrieved September 29, 2020, from https://uavsystemsinternational.com/products/tarot-x6

[5] Buy Matrice 600 Pro. (n.d.). Retrieved September 29, 2020, from https://store.dji.com/product/matrice-600-pro?site=brandsite

[6] Turnipseed, D. Phil, and Vernon B. Sauer. "Discharge Measurements at Gaging Stations." Report. Techniques and Methods. Reston, VA, 2010. *USGS Publications Warehouse.* https://doi.org/10.3133/tm3A8.

[7] LiDAR POD compatible with aerial amp; bathymetric drones: Hélicéo. (n.d.). Retrieved September 13, 2020, from http://www.heliceo.com/en/produits-pour-geometres/lidar-technology/

[8] Li, W., Liao, Q., and Ran, Q. "Stereo-Imaging LSPIV (SI-LSPIV) for 3D Water Surface Reconstruction and Discharge Measurement in Mountain River Flows." Journal of Hydrology, Vol. 578, 2019, p. 124099. https://doi.org/10.1016/j.jhydrol.2019.124099.

[9] CEE ECHO- High Definition Single Beam Echo Sounder. (n.d.). Retrieved September 13, 2020, from http://www.ceehydrosystems.com/products/single-beam-echo-sounders/cee-echo/

[10] Kinzel, P. J., and Legleiter, C. J. "SUAS-Based Remote Sensing of River Discharge Using Thermal Particle Image Velocimetry and Bathymetric Lidar." *Remote Sensing*, Vol. 11, No. 19, 2019, p. 2317. https://doi.org/10.3390/rs11192317.

[11] US Department of Commerce, N. O. and A. A. *What Is Bathymetry?* https://oceanservice.noaa.gov/facts/bathymetry.html. Accessed Sep. 26, 2020.

[12] Crocker, T. R. "Near-Surface Doppler Sonar Measurements in the Indian Ocean." Deep Sea Research Part A. Oceanographic Research Papers, Vol. 30, No. 4, 1983, pp. 449–467. https://doi.org/10.1016/0198-0149(83)90078-X.

[13] Dugan, J. P., Anderson, S. P., Piotrowski, C. C., and Zuckerman, S. B. "Airborne Infrared Remote Sensing of Riverine Currents." IEEE Transactions on Geoscience and Remote Sensing, Vol. 52, No. 7, 2014, pp. 3895–3907. https://doi.org/10.1109/TGRS.2013.2277815.

[14] Teledyne RD Instrument. *Acoustic Doppler Current Profiler, Principles of Operation: A Practical Primer.* https://www.comm-tec.com/Docs/Manuali/RDI/BBPRIME.pdf. Accessed Sep. 26, 2020.

[15] UAV integration services: UgCS Drone Industrial solutions. (n.d.). Retrieved September 14, 2020, from https://industrial.ugcs.com/bathymetry

[16] What is Jason-3? (n.d.). Retrieved September 13, 2020, from https://www.nesdis.noaa.gov/jason-3/mission.html

[17] FAA. “Summary of small unmanned aircraft rule,” *Federal Aviation Administration Regulations*, Part 107, pp. 1-2.

[18] Hock, R., G. Rasul, C. Adler, B. Cáceres, S. Gruber, Y. Hirabayashi, M. Jackson, A. Kääb, S. Kang, S. Kutuzov, Al. Milner, U. Molau, S. Morin, B. Orlove, and H. Steltzer, 2019: *High Mountain Areas. In: IPCC Special Report on the Ocean and Cryosphere in a Changing Climate* [H.-O. Portner, D.C. Roberts, V. Masson-Delmotte, P. Zhai, M. Tignor, E. Poloczanska, K. Mintenbeck, A. Alegria, M. Nicolai, A. Okem, J. Petzold, B. Rama, N.M. Weyer (eds.)]. In press.

[19] Vliet, Michelle T. H. van, Wietse H. P. Franssen, John R. Yearsley, Fulco Ludwig, Ingjerd Haddeland, Dennis P. Lettenmaier, and Pavel Kabat. “Global River Discharge and Water Temperature under Climate Change.” *Global Environmental Change* 23, no. 2 (April 1, 2013): 450–64. https://doi.org/10.1016/j.gloenvcha.2012.11.002.

[20] NASA SWOT. “Home.” Accessed September 11, 2020. https://swot.jpl.nasa.gov/.

[21] “Malt Quality Lab,” *Malt Quality Lab - Barley Breeding Program | Montana State University*, from https://www.montana.edu/barleybreeding/malt-quality-lab/.

[22] sUAS Drone Thermal Vision. "VuIR Lepton - Cheapest Thermal System for drones for beginners." *VuIR Lepton*, from https://www.suas.com/product-page/vuir-lepton-cheapest-thermal-system-for-drones-for-beginners.

[23] Seek Thermal. "Micro Core". *Micro Core Specification Sheet*, from

[24] FLIR. *FLIR Vue Pro R*, from https://www.flir.com/products/vue-pro-r/.

[25] Young, D. S., Hart, J. K., and Martinez, K. “Image Analysis Techniques to Estimate River Discharge Using Time-Lapse Cameras in Remote Locations.” *Computers Geosciences*, Vol. 76, 2015, pp. 1–10. https://doi.org/10.1016/j.cageo.2014.11.008.

[26] Molland, A. F., Ed. Chapter 3 - Flotation and Stability. In The Maritime Engineering Reference Book, Butterworth-Heinemann, Oxford, 2008, pp. 75–115.

[27] S.g, S. “Safer and More Efficient Ship Handling with the Pivot Point Concept.” TransNav, International Journal on Marine Navigation and Safety od Sea Transportation, Vol. 10, No. 4, 2016.

[28] J. Gelling, "The Axe Bow: The Shape of Ships to Come," in 19th International HISWA Symposium on Yacht Design and Yacht Construction, Amsterdam, The Netherlands, 13 and 14 November 2006.

[29] Ulstein, "X-bow," Ulstein, 2005. [Online]. Available: https://ulstein.com/innovations/x-bow. [Accessed Nov 20 2020].