

Geo-information science and Remote sensing
Thesis report GIRS-2021-45

DEVELOPING A MOBILE WATER QUALITY SENSING PLATFORM FOR USV-BASED MONITORING

Developing a mobile water quality sensing platform for USV-based monitoring

Jurrian Doornbos

registration number 97 137 917

Supervisors:

dr. João Valente
dr. ir. J.G.P.W. (Jan) Clevers

A thesis submitted in partial fulfillment of the degree of Master of Science
at Wageningen University and Research
The Netherlands

7-7-2021
Wageningen, the Netherlands

Thesis code number: GRS - 80436

Thesis report: GIRS - 2021 - 45

Wageningen University and Research
Laboratory of Geo-Information Science and Remote Sensing
Social Artificial Intelligence Drone Lab

Acknowledgements

First and foremost I want to thank my supervisors dr. João Valente and dr. ir. Jan Clevers. João for the continuing faith in my capabilities in developing the system and the opportunities presented within this project. Jan for his efforts in setting up the meetings with the Water Quality chair group, next to keeping track of when the thesis would be ready to finish. Both have been incredibly helpful in going over the feedback whilst keeping track of the bigger picture.

I would like to acknowledge the help that Teun de Visser has been in the development of the USV-control side of the project. Without his efforts, the whole project would have taken longer and would have been less insightful.

Lastly, I would like to thank my parents Marc, Liesbeth, and my sister Arniek for helping me out by being both sparring partners and cheerleaders during the complete duration of this project.

Abstract

This work presents the development and experimentation with a low-cost water quality sensing platform mounted on a USV (Unmanned Surface Vehicle). The objective was to understand the complete picture of such a system: from the individual sensors to the final mapping capabilities. Such a system could help understand the intricate dynamics at play in a water body. The used sensors were TDS (Total Dissolved Solids), turbidity, temperature and pH. Only the temperature sensor was capable of taking reliable measurements for a longer period. The others could be improved with more intelligent software, and stable voltage referencing. Furthermore, the developed system showed that the findings from the sensors could be mapped real time, and be used for further analysis. The limitations however, lie in the quality of these sensors, and the technology used in these sensors in order to create reliable measurements. This larger picture was placed in a broader context, where it was revealed that every water quality sensing system is a trade-off between three important features. In order to understand how a USV could help understand the intricate dynamics in a water body, the trade-offs in the design of the USV need to be understood first.

Contents

1	Introduction	1
1.1	Water quality	1
1.2	Water sensors	2
1.3	Robotics for water quality sensing	3
2	Background	5
2.1	Fundamentals of environmental robotics	5
2.2	Fundamentals of water quality	6
3	Methodology	9
3.1	Sensing water	9
3.1.1	Turbidity	9
3.1.2	Acidity	10
3.1.3	Conductivity	11
3.1.4	Temperature	11
3.1.5	Depth	12
3.2	System Overview	13
3.2.1	Boxes	13
3.2.2	Wiring	16
3.2.3	Microcontroller programming	18
3.3	Calibration	18
3.3.1	Process	19
3.4	ROS	20
3.4.1	The Teensy Node	21
3.4.2	The Sonar Node	22
3.5	Experiments	23
3.5.1	Mapping the GAIA lake	24
3.5.2	Basic processing	25
3.6	Interpolation	26
3.6.1	Inverse Distance Weighting	26
3.6.2	Sampling	26
3.6.3	Accuracy	27
4	Results	29
4.1	Sensor costs and battery life	29
4.2	Calibration and accuracy	30
4.3	Interpolation	32
4.3.1	Continuous Sampling	32
4.3.2	Hold Location Sampling	36

5 Discussion	40
5.1 Sampling methods & interpolation techniques	40
5.2 Broader perspective	41
6 Conclusion	44
6.1 Recommendations	45

List of Figures

2.1 Example of a water quality sensing USV (Jo et al., 2019)	5
3.1 Product image and specification overview from the manufacturer (“Gravity: Analog Turbidity Sensor For Arduino”, n.d.)	9
3.2 Product image and specification overview from the manufacturer (“Gravity: Analog pH Sensor / Meter Kit For Arduino”, n.d.)	10
3.3 Product image and specification overview from the manufacturer (“Gravity: Analog TDS Sensor/Meter for Arduino”, n.d.)	11
3.4 Product image and specification overview from the manufacturer (“Gravity: Waterproof DS18B20 Sensor Kit”, n.d.)	12
3.5 Product image and specification overview from the manufacturer (“SONAR 2D-ENHANCED”, n.d.)	13
3.6 The schematic overview of the sensor array	14
3.7 Sensor box design and assembly	14
3.8 Electronics box design and assembly	15
3.9 Wiring diagram of the electronics box	17
3.10 Flowchart showing the steps executed in the calibration	20
3.11 Overview of the ROS nodes running when the sensor system is active	21
3.12 State of the system before starting the experiments	23
3.13 Dashboard screenshot for live monitoring	24
3.14 Timeline from the first half of 2021, when all the experiments on the GAIA lake took place	25
3.15 Interpolation pipeline	27
4.1 Results from calibration	31
4.2 Train and test split of the point data	32
4.3 Contextual information continuous interpolation	33
4.4 Cross validation results for the tenfold MSE search	34
4.5 Interpolated final maps for the continuous method	34
4.6 Accuracy scatter plot for the continuous method	35
4.7 Contextual information location hold interpolation	36
4.8 Cross validation results for the LOOCV MSE search for each sensor .	37
4.9 Interpolated final maps for the location hold method	38
4.10 Accuracy scatter plot for the location hold method	39
5.1 The trade-off for water quality robotics mental map	42

Chapter 1

Introduction

The surface of the earth is for 70 percent covered by water (Shiklomanov, 2000), 96.5 percent of this water is found in the oceans and seas. This means that only 3.5 percent of all water on earth is freshwater, most of it stored in glaciers. The last small percentages of water support almost all terrestrial life on this planet. Maintaining the quality of this natural resource is therefore essential (Oki & Kanae, 2006). Maintaining this resource is one of the main directives for the sustainable development goals (SDG) and integral to part 6 and 14 of the SDGs (United Nations General Assembly, 2015). In Europe, a framework for clean waters has been in place since 2000. This framework serves as a foundation to support sustainable management of the water basins inside Europe (European Commission, 2000). One of the key features in this framework is repeated monitoring of a variety of pollutants and water quality indicators.

Furthermore, good water quality is important not only to the surrounding ecosystem (Keeler et al., 2012), but also to the short and long term human health impacts (Graham et al., 2014; Leigh et al., 2010), either in the form of decreased effectiveness of antibiotics, or more directly with toxic cyanobacteria (Carmichael, 1994).

1.1 Water quality

According to Boyd (2000), pollutants are external sources that drain into the water surface, either from point sources, such as drainage pipes or factories or non-point sources which mainly consist of fertilization for crops, and pastures. These pollutants have a variety of complex chemical and biological effects on the water body, the most significant one is called eutrophication (Scheffer, 1997). Eutrophication is a process, where input of nutrients causes the water to change from a clear, nutrient-poor water body to a turbid, nutrient-rich water, the excess of nutrients can cause the plants to excel, or algal blooms to proliferate (Scheffer, 1997). Water quality processes such as eutrophication are complex to understand and monitor (Jones & Mulholland, 1999). Furthermore, once a surface water has been disturbed it is difficult to restore due to hysteresis (Scheffer, 1997). Hysteresis defines that the rate of change from stable state A to stable state B, is not the same as going back from B to A. This is due to past events in the history of the system. These past events have changed regulatory processes in the system, which inhibit a simple return to state A.

The field of limnology, the field of study on inland waters, concerns with this exact complex interaction, where nutrient inputs of carbon, oxygen, phosphorus, and

nitrogen in its different forms are converted into biomass, stored in the sediments or released into the atmosphere (Downing et al., 2008). This takes place at different scales, such as the chemical level where increasing acidity can dissolve shells and limestone (Ware et al., 1991). It could also occur in biological processes such as photosynthesis of plants creating biomass, and the subsequent benthic species that consume these plants (Scheffer, 1997). With this information, water quality indicators merely attempt to measure and monitor these highly dynamic processes by finding indicators that can measure the resulting chemical changes taking place in the water body (Li & Liu, 2019).

In recent years, limnology research has posed that smaller water bodies, such as ponds, have a disproportionately large impact on the surrounding ecosystem (Downing, 2010; Mullins & Doyle, 2019; Verpoorter et al., 2014). According to Downing (2010), this impact exists because the variety of species per kilometer is exponentially larger in smaller lakes; a similar pattern occurs for fish yield in differently sized lakes. Furthermore, research so far has shown that ponds serve as highly effective sinks for phosphorus deposition (Forbes et al., 2012). These first investigations are showing a high biogeochemical activity, as expected from the high biodiversity. This means that a pond provides services for an ecosystem that are essential in many terrestrial ecosystems (Mullins & Doyle, 2019).

However, the biogeochemical cycles inside these ponds have not been that well understood as for the larger bodies of water, which have seen most of the published research according to Marton et al. (2015). One of the reasons being the shortage of data on smaller ponds (Mullins & Doyle, 2019). As mentioned before, limnological research looks at a variety of dynamics from chemical, geophysical to biological ones. Dynamics such as these require multiple observations over time, with highly similar methods, making comparisons possible. This is a time and resource intensive method of acquiring data (Tuna et al., 2013). Getting a better understanding of the smaller ponds would require a method for getting information, with a reduction in time and a lower amount of resources required.

1.2 Water sensors

Environmental sensor technology has rapidly grown as both a research field and industry in the past two decades (Dunbabin & Marques, 2012). The main improvements have been the reduction in size of the sensor. For example, turbidity is the optical clarity of a fluid, which is usually measured in Nephelometric Turbidity Units (NTU). The Environmental Protection Agency (EPA) has defined this standard as the amount of light passing through the water, where a light source is placed 90 degrees from a photovoltaic sensor, which detects light centered at a specific wavelength: 780nm (Li & Liu, 2019). This standardized method has created many industrial designs with large tungsten lightbulbs and photo resistive sensors. More recently however, versions using a small LED and sensor have become available (Wang et al., 2018). Next to size, the cost of environmental sensors have gone down in the past two decades as well. Castell et al. (2017) have shown for air quality sensors that the newer, low cost, sensors are not as accurate in measuring the quality, but offer good, coarse information.

Robotics has been proven to be a useful tool in repeated monitoring of agricultural fields, harsh environments, and water bodies (Dunbabin & Marques, 2012).

Furthermore, Dunbabin and Marques (2012) pose that environmental analysis is becoming more and more sensor-focused, as research gets more data intensive, and broader conclusions can be made about the environment sensed, using these large datasets. They also mention that with more environmental sensors becoming accessible, robots can do more of the labor intensive, highly frequent monitoring of the environment.

Environmental science is focused on understanding the natural environment in relation to a changing planet (Carpenter et al., 1992). Robotics has been focusing on autonomy of different platforms, for a variety of functions, from self-driving trains to warehouse carts (Khairuddin et al., 2016). Robotics and environmental sciences are widely different fields of research. However, both fields share a similar interest in sensor technology and sensing the environment. For ecologists, the interest lies in the potential of data-driven environmental analysis (Martinez et al., 2004), in the case of robotics it lies in improved autonomous movement (Khairuddin et al., 2016). With the increasing interest in sensor technology from both fields. There is more overlap between these fields, making information on the quality of the sensors and different applications sensors can have more valuable.

1.3 Robotics for water quality sensing

Understanding the ecology of a pond requires detailed and frequent data collection, which is not feasible, due to existing methods being resource intensive. With the recent developments in sensor technologies, new opportunities arise to automate this data collection. Such sensor-platforms have already been developed such as in Jo et al. (2019), Kaizu et al. (2011), Koparan et al. (2018), and Romero-Vivas et al. (2015).

Jo et al. (2019) present an open-source water quality unmanned surface vehicle (USV) platform. This work emphasizes an open-source and low-cost approach to making such a vehicle. The USV itself was validated by placing it on a water body and checking the mobility, and whether the sensors sent information via wireless communication. The system itself was capable of measuring the water quality of a variety of ponds in the USA and Peru, but was missing calibration and understanding of the sensors themselves.

Kaizu et al. (2011) present a system where water quality sensors are placed on an autonomous USV. The paper emphasizes the control algorithm, in combination with the sampling method. The acquired data was interpolated on a regular grid, showing spatial difference in the observations. Kaizu et al. (2011) also show that some of the water quality parameters changed during the sampling, due to the time it took to map the lake. They also showed that different sampling paths, such as a spiral, might improve the accuracy.

Koparan et al. (2018) present an Unmanned Aerial Vehicle (UAV) to sample the water. The UAV lowers itself onto the water surface and thus lowers their pre-calibrated open source multiprobe meter (OSMM) into the water body. The measured values were stored on a SD card. The sensors were also compared to a commercial multiprobe meter by simultaneously lowering them from a kayak. The software running the OSMM, the parts used and the calibration procedure were not presented in this paper, making replication impossible. Furthermore, the 6 minutes flight time of the UAV makes longer sampling impossible, without battery changes.

None of the above examples show the complete approach. The complete approach would entail the costs, the internal working of the sensors, the mounting system, the software implementation of the sampling, the robotics implementation, the different calibrations and the specific interpolation steps to attain the final maps. This complete picture has the possibility to more specifically show where the shortcomings of such a system are, as well as explore the features of the system.

This thesis presents the development of a water quality sensing USV, from start to finish. By presenting the complete picture, the costs of such a system, the mounting options on a USV, the calibration accuracy, sampling methods, and interpolation results can be seen as an interrelated system. This in turn explores how a contribution from environmental robotics can add to the broader understanding of small water bodies. To structure this thesis, the main research question is presented below.

Does a low-cost USV for water quality measuring contribute to mapping small water bodies?

This question negates the notion that the development of such a platform will contribute to water quality research. Instead, it focuses on whether and how it might contribute to water quality monitoring processes. To answer this main research question, it is divided into the following sub-questions:

- *Which software and hardware implementation decisions play a role in developing the sensing system for a low-cost USV for water quality monitoring?*
- *How do interpolation and sampling techniques influence mapping the water quality indicators acquired by the USV?*
- *How does the processed data from the developed USV at a small pond contribute information about the water quality of the small pond?*

In order to answer these research questions with the correct context, the underlying theory of water quality and robotics is presented in chapter 2.

The extensive methodology, chapter 3, presents the acquired sensors, the steps executed to place them into a sensor array, in both software and hardware. This section gives an answer to first sub-question on implementation decisions by emphasizing which parts were used, how they were connected, as well as an in-depth look at the software implementation.

In order to answer the second sub-question on mapping these parameters, the final sections of the methodology go into the sampling and interpolation techniques that directly influence the final interpolated maps.

In chapter 4, results, the findings from both the sensor accuracy, as well as the interpolation is presented. This gives an answer to the third sub-question, on the contribution of such a system to information on the water quality of a pond, by delving into the calibration results and the interpolation results. In extension, the limitations of this system are presented in the discussion, chapter 5.

These main research questions are answered in the conclusion, chapter 6. Here, the takeaways and lessons from this thesis are presented in order to give the complete answer which implementation decisions lead to the best mapping results and what needs to change in order to contribute to mapping the water quality of small ponds.

Chapter 2

Background

This chapter creates the context around the sensor array to be developed, and places the sensor-array in the scientific debate. Because this thesis fits within aquatic ecology and robotics, this context also serves to clarify concepts from these different scientific disciplines.

This chapter is separated in two sub-chapters. The first chapter is an introduction into environmental robotics, the field of research where the field of robotics is applied to keep track, and interact with the natural environment. The second sub-chapter delves into the principles of water quality.

2.1 Fundamentals of environmental robotics

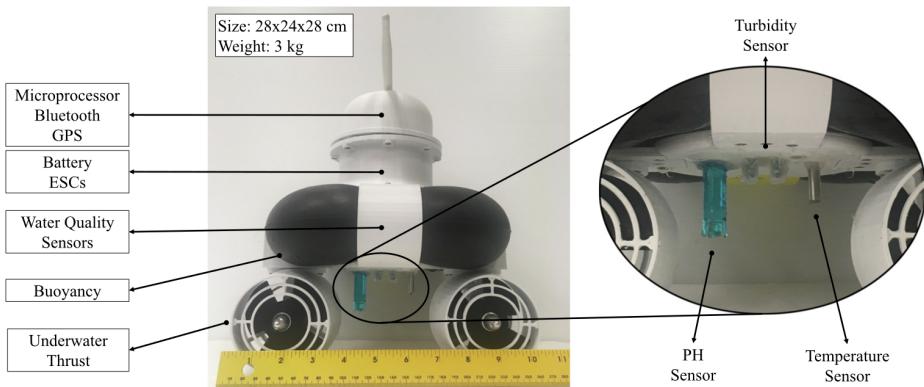


Figure 2.1: Example of a water quality sensing USV (Jo et al., 2019)
This system uses two motors for propulsion and has multiple water quality sensors below the hull touching the water. In the main compartment, the logic boards, GPS, and Bluetooth communication are present.

For this thesis, robots are understood as systems capable of reacting to their environment whilst executing tasks. This definition is broadly outlined in Siciliano and Khatib (2008), where the fundamentals of robotics are presented, from kinematics to autonomous control. Furthermore, this handbook divides a robot into its kinematics and mechanics, sensing capabilities and control algorithms (Siciliano & Khatib, 2008). The kinematics of a robotic system are the collections of poses that a robot can take on, based on the rigid bodies and joints that make up the

movement of the robot. This is a difference to the mechanics and dynamics, which entail the processes that can induce the movement of the robot (such as the motors in figure 2.1). The sensing part of the robot concerns estimating the state of the robot at each point in time. Furthermore, when using more sophisticated sensors, one can look into the perception of the robot, in which the sensors are used in space and time to facilitate the planning and execution of tasks (for the system in figure 2.1, Jo et al. (2019) suggest using the on-board GPS). Finally, there are the control algorithms, this is the set of algorithms concerning the complete control of the system, from motion planning to execution of the motion, or even broader, executing the task at hand. The complexity of these algorithms can range from a simple logic statement (if this, then that) all the way to reinforcement learning algorithms (algorithms which punish and reward based on pre-defined criteria, in order to facilitate self-learning).

Dunbabin and Marques (2012) introduce the concept of robotics for environmental monitoring. They explain that robotics are used increasingly in the exploration and understanding of the natural environment, from the deep oceans, to algal blooms, volcano craters, and even another planet. Furthermore, the increased pressure on natural systems requires precise and spatio-temporally dense information. Dunbabin and Marques (2012) identify robotics as a promising tool for acquiring this data, due to the repeatability of measurements and access to hostile environments. Different environments face different challenges, however the diversity in which robotics platforms exist, makes any environment one that can be observed. Measurements are done via sensors: encoder information, cameras, laser scanners and GPS can all be seen as sensors (Siciliano & Khatib, 2008). Furthermore, there are sensors that measure one exact feature, such as temperature, CO₂ or NH₄. This is also where the interaction between the robot and the environment lies. These sensors are capable of monitoring the natural environment, and the observations can be used in environmental analysis. This overlap between sensor application and usage for both robotics and environmental sensing is the field of environmental robotics.

2.2 Fundamentals of water quality

Before understanding water quality dynamics and its methodological counterpart, sensors, an overview of water quality is required. This is what this sub-chapter presents, using table 2.1 as a reference of different parameters that have been implemented in other USVs. Important to note is the difference between a water quality process and a water quality parameter. This section focuses on the different processes and how these parameters come into focus.

Water is always in motion. Water can move from mountaintops through rivers, to the oceans and evaporate into clouds. Water is cyclic in nature (Sitch & Drake, 2014). Sitch and Drake (2014) explain that these hydrological cycles can be seen as exchanges (fluxes) between water reservoirs. The fluxes themselves are the processes that entail the phase changes of the water such as condensation, precipitation and evaporation. When going from this global view to the local situation, many of these processes still apply. However, local effects are also at play. This is also where the complex local processes governing the quality of the water come into perspective (Scheffer, 2004a). The processes are different for each type of water reservoir. As this thesis focuses on small water bodies, it will omit the other reservoir types

Table 2.1: An overview of different water quality parameters.

Parameter	Used in other WQ-USVs	Used in thesis
Turbidity	Casper et al. (2012), Jo et al. (2019), and Kaizu et al. (2011)	x
EC	Casper et al. (2012), Kaizu et al. (2011), and Koparan et al. (2018)	
Temperature	Casper et al. (2012), Jo et al. (2019), Kaizu et al. (2011), and Koparan et al. (2018)	x
Acidity	Jo et al. (2019), Kaizu et al. (2011), and Koparan et al. (2018)	x
ORP		
DO	Casper et al. (2012), Kaizu et al. (2011), and Koparan et al. (2018)	
TDS		x
Chlorophyll-a	Casper et al. (2012) and Kaizu et al. (2011)	

EC = Electrical conductivity; ORP = Oxygen reduction potential; DO = Dissolved oxygen; TDS = Total dissolved solids. The table shows which parameters have been used in other studies. It also shows which parameters were used in this thesis. Dissolved Oxygen is used in most other system, but not in this project. TDS is the replacement parameter for EC.

such as oceans and clouds. The small water body in this thesis are the shallow lakes and ponds found numerously across the Netherlands (Scheffer, 2004a). In this reservoir type, the water remains longer, and thus favours big changes in water quality. The chemical, physical and biological characteristics of the water body is what defines water quality (Li & Liu, 2019). In order to understand water quality, these characteristics need to be understood.

The physical characteristics of water quality can broadly be subdivided into light, sediments and nutrients. When looking at light in the water, what is most important is the clarity of the water, called turbidity. Clear water without sediments floating around can pass a lot of light into the deeper regions; so as turbidity increases, the amount of light goes down. Sediments in turn are the particulates on the bottom of the water and floating around in the water column. Thus, it contains the soil at the bottom, which can be loosened and disturbed by fish and phytoplankton and gets mixed in the water. The bottom soil contains a lot of important nutrients for the species in the water. This means that when it gets well mixed in the water column, there are a lot of nutrients floating around, to be used by the species that require them. The most important nutrients are phosphorus, nitrogen and oxygen. Each nutrient has its own seasonal cycle it goes through (Wang et al., 2019). To understand the type of interactions that each of these nutrients has on the water quality, phosphorus is taken as an example. Phosphorus particles naturally originate from organic matter, such as excretion from fish. This phosphorus binds to the sediment particles suspended in the water column and slowly drift to the bottom (Carstensen et al., 2014). Phytoplankton can take up this suspended phosphorus and grow, as more sunlight hits the earth during the spring and summer months and thus intensifies the photosynthesis process. There is one more important factor to have in mind when looking at lakes. As phosphorus is bound to sediments, it can be stored in the bottom sediment layers of the water body (Carstensen et al., 2014). Thus, a sudden increase in phosphorus in the water can be stored in the bottom

sediment layers. This buffer of phosphorus can be released back into the water, but this exact sediment-water interaction is not perfectly understood (Scheffer, 2004c). Generally, iron has a very strong impact on the phosphorus being able to bind to sediments, and be released. However, this only works when there is enough oxygen present. If the water becomes anaerobic, this iron-bound phosphorus is released (Scheffer, 2004c). Furthermore, the acidity of the water can have a strengthening or dampening effect on this interaction. This shows that when looking at an individual nutrient, the whole chemistry of the water matters.

The biological level looks at the flora and fauna of the water body, such as phytoplankton, fish and vegetation. Phytoplankton is a micro-organism that lives in the water and grows by photosynthesis. This species thus takes in the nutrients in the water and combines it with sunlight to increase in biomass. This once again means that there is an interaction between the turbidity of the water, and thus the amount of light passing through at certain depths, and the possibility of biomass creation by phytoplankton (Scheffer, 2004b). An important distinction must be made within this group, which are the cyanobacteria. They are most commonly known as blue-green algae, taking over the lake in sunny periods of the year, and reducing the fish stocks and vegetation due to the lack of light, which is blocked by the density of this algae at the upper surface. This is a harmful algal bloom (HAB), these blooms can even be toxic to humans (Scheffer, 2004b).

This then makes external nutrient loading, such as wastewater disposal and fertilization, a large problem. These sources of pollution disturb the finely tuned interactions, and transform the system into a eutrophic state.

However, one cannot see the whole image without taking everything in account. That is where the term ecosystem comes in, the interactions between these scales are captured and understood within the larger system. Furthermore, it acknowledges the non-linear interactions between the species, which has troubled water quality management over the decades.

Chapter 3

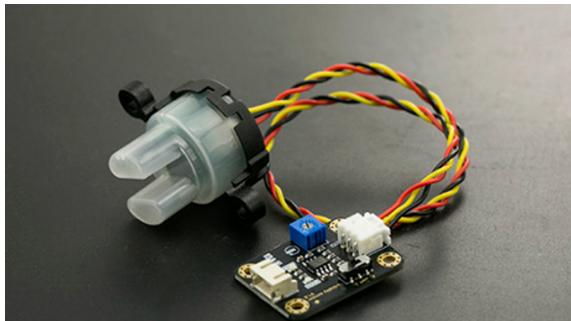
Methodology

This methodology serves to explain each of the water quality sensors, introduce the system architecture of the sensor on the USV and give an insight into the test-runs of the USV. Finally, the interpolation processing for the sensor data is presented.

3.1 Sensing water

For environmental measurement, an understanding of the accuracy of the sensor is required. For robotics, the limitations of the sensor need to be understood in order to implement it in the robot system. Therefore, an explanation of the inner-working of each environmental sensor used in the system is important for both perspectives.

3.1.1 Turbidity



(a)

(b) Turbidity sensor specifications

Range	0 – 4.5 V
Operating temperature	5 – 90 °C
Accuracy	±4.88 mV
Response time	≤ 500 ms
Operating Current	40 mA

Figure 3.1: Product image and specification overview from the manufacturer (“Gravity: Analog Turbidity Sensor For Arduino”, n.d.)
for measuring turbidity

The suspended solids in a water body is a broad indication for the quality of this water. An optical method in quantifying these solids, is turbidity. The basis of this method is that murky water has more particles in it that will scatter the light around, and thus make the source of light less visible (Li & Liu, 2019). The methods used to measure the turbidity can yield large differences, as the type of light, and light measurement technique can vary. Each standard of water quality can have its

own version of a turbidity quantification. There are two implementations. The first is the nephelometric method, in which a light source is at a 90 degree angle of the photovoltaic sensor (a sensor sensitive to light). This is deemed as the optimal angle for light scattering (Li & Liu, 2019). The limitation being that it is only optimal to around 40 NTU with higher turbidities, the linear relationship between turbidity and the light scattering is non-linear. On the other hand the attenuation method uses a 180 angle to measure the light coming in, with the light source directly opposite of the sensor. This method however, means that sensors are highly susceptible to the colour of different light sources, which make them less suitable for use in water quality standards.

This thesis uses the SEN0189 by DFRobot (“Gravity: Analog Turbidity Sensor For Arduino”, n.d.), an attenuation-style sensor. According to Hakim et al. (2019) this sensor is capable of distinguishing large differences in turbidity. This sensor works by emitting an infrared light emitting diode (LED) onto the phototransistor on the other side. This phototransistor will increase or decrease the output voltage, based on the amount of light that hit the transistor. In the paper by Hakim et al. (2019), they tested the SEN0189 by placing it in water with a known turbidity between 0 and 1000 NTU. In this experiment, they found that the output voltage of the SEN0189 linearly correlates ($R^2 = 0.92$) to the NTU values. This shows that in a controlled environment, the sensor is capable of measuring the difference in turbidity of the water.

Correlating NTU to an attenuation-style sensor is problematic because of the inherent difference in measurement method. Due to the linearity of the results from Hakim et al. (2019), from around 0 NTU all the way to 1000 NTU, the experiment shows that this sensor is capable of distinguishing large differences in turbidity, but lacks the precision a nephelometric meter will have in the lower regions.

3.1.2 Acidity



(a)

(b) pH sensor specifications

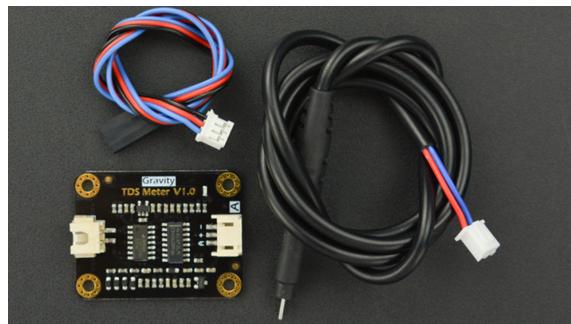
Range	0 – 14 pH
Operating temperature	0 – 60 °C
Accuracy	±0.1 pH
Response time	≤ 1 min
Operating current	Not provided

Figure 3.2: Product image and specification overview from the manufacturer (“Gravity: Analog pH Sensor / Meter Kit For Arduino”, n.d.)
for measuring pH

Acidity, and its counterpart alkalinity, is determined by the activity of hydrogen ions. This is quantified using the pH scale, which is a number between 0 and 14, with the number 7 meaning a neutral solution. This scale is logarithmic, meaning that change of 1 pH, leads to a 10 fold change in acidity.

Measuring acidity can be done in a variety of ways, such as the colour changing litmus paper. This thesis uses the electrode method, using the pH Sensor by DFRobot (“Gravity: Analog pH Sensor / Meter Kit For Arduino”, n.d.). This sensor uses an electrochemical sensor, with a measuring and reference electrode. The probe is also used by Jo et al. (2019), with moderate success. The measuring electrode is made of a glass which is highly sensitive to hydrogen ions (Li & Liu, 2019), it is also filled with a pH neutral solution. When placed into a solution, a change in voltage occurs between the two electrodes. This change in voltage is amplified using external circuitry, and the voltage can be transformed to a pH value by calibrating the sensor.

3.1.3 Conductivity



(a)

(b) TDS sensor specifications

Range	0 – 1000 ppm
Operating temperature	0 – 60 °C
Accuracy	±2 ppm
Response time	≤ 500 ms
Operating Current	3 – 6 mA

Figure 3.3: Product image and specification overview from the manufacturer (“Gravity: Analog TDS Sensor/Meter for Arduino”, n.d.)
for measuring TDS

Conductivity is a property of all materials, and explains the amount of electrons that can pass through the material. In the case of water, conductivity is determined by the amount and type of minerals present in the water. Dissolved salt for example, increase the potential of charges to flow through the water. There are sensors that can directly infer the electrical conductivity (EC) of a solution.

Total Dissolved Solids (TDS) is a measurement of hardness of the water. When you take a sample of water, and evaporate all the water, the particles that are left over are the dissolved solids. This means that EC and TDS are closely related, but not the same. TDS can be converted to EC, when the conductive property of this material is known. When water has a high TDS value, it has more solid particles in it. In general, the cleaner the water, the lower this value. It is measured by measuring the voltage going across two prongs, with a known distance apart, as more voltage can get through, the water has more solids in it. The DFRobot implementation of this method was used in this thesis (“Gravity: Analog TDS Sensor/Meter for Arduino”, n.d.).

3.1.4 Temperature

In almost all chemical reactions, temperature plays a significant role, due to the activity of the particles. Temperature is physically understood as the vibration of



(a)	
(b) Temperature sensor specifications	
Range	-55 – 125 °C
Operating temperature	-55 – 125 °C
Accuracy	±0.5 °C
Response time	≤ 1 min
Operating current	3µA

Figure 3.4: Product image and specification overview from the manufacturer (“Gravity: Waterproof DS18B20 Sensor Kit”, n.d.)
for measuring temperature

the H₂O particles. Densely packed H₂O particles which do not vibrate, are the solid form, ice. The loosely packed shaking particles are gas. In between is fluid water. The celsius scale of temperature is determined by these properties of water. At 0 degrees Celsius, water freezes. At 100 degrees it vaporizes into its gaseous form, water vapour. It is important to keep in mind that pressure also plays an important role, water can vaporize without a change in temperature but purely a change in pressure. Therefore, the Celsius scale is determined at one atmosphere of pressure. Temperature sensors are available in a wide variety of options, for a wide variety of applications.

This thesis uses the DS18B20 Waterproof by DFRobot (“Gravity: Waterproof DS18B20 Sensor Kit”, n.d.). The DS18B20 is a small temperature sensor made with its own analog to digital conversion and digital 1-Wire communication protocol. The DS18B20 has an accuracy of around 5% with a range between -55°C and + 125°C.

The temperature sensing on the DS18B20 is done by using the bandgap temperature method. This method directly correlates the voltage level at the output of a bipolar junction transistor (BJT) to the temperature (Makinwa, 2010). The precision and reliability from this method comes from the design of the DS18B20, in which a temperature dependent and temperature independent voltage source are sampled, and thus the gap between these can be related to the temperature measurement. The BJT, analog-to-digital converter, temperature calculation circuitry and the 1-wire interface is all baked into this small temperature sensor package, meaning that a power, ground and communication wire is all that is required to create reliable temperature measurements (Maxim Integrated, 2019).

3.1.5 Depth

Another useful parameter of a water body is the morphology: the shape of the water body. Light will quickly scatter in water, meaning that a radar or camera is not feasible in deeper waters. So to understand the bottom surface of the water body, an echo-sounder can be used (Abraham, 2019), in combination with a digital system to attain the useful information, together called a sonar. An echo-sounder sends out a tone which passes through the water at around 1500 m/s, and when it hits a surface, it is scattered. Some of these scattered sound waves return to



(a)

(b) Sonar specifications

Range	0 – 14 pH
Operating temperature	–5 – 60 °C
Accuracy	±1 cm
Echo rate	≤ 50 hz
Beamwidth ‘	10 °
Operating current	500 mA

Figure 3.5: Product image and specification overview from the manufacturer (“SONAR 2D-ENHANCED”, n.d.)

the echo-sounder. The return time of this sound wave is directly proportional to the depth of the water. Different sonar implementations exist, such as a multi-beam echo-sounder (MBES), single-beam echo-sounder (SBES) and a side-scan sonar (SS). A single beam echo-sounder uses a single pulse of sound and listens for the return signal, this can be directed downward and as such give insight into the depth of the surface below. Important here is to understand the shape of the sound that is sent out. As it passes through the water the cone of sound increases. And thus the area that it interacts with is larger and less defined.

For this thesis the Kogger Sonar Enhanced was used (“SONAR 2D-ENHANCED”, n.d.). This sonar is a single beam echosounder with a digital interface, for communication with a UART capable machine, such as a computer or microcontroller.

3.2 System Overview

The system is one part hardware and one part software. The design decisions for both are presented in this section. A broad overview is shown in figure 3.6. This overview has three subsections: the boxes, the wiring and the microcontroller programming.

Presented in these sections are the end-result of continuously testing and improving upon decisions previously made. The first designs were not the final designs for the system, these iterations are shown in more detail in the experiments section (section 3.5).

The designs resulted in two boxes, one is a submersible sensor box and the other is exclusively for all the electronics. The 3D files are available as .stl files on the thesis GitHub and GitLab (“jurriandoornbos/wqdrone”, 2021).

Furthermore, the different connections between all the parts are covered in the subsection on wiring. This is followed by an overview of the microcontroller programming that took place. The source code for every part is also available in this repository.

3.2.1 Boxes

The sensor box is a 3D-printed enclosure with holes to fit all the sensors, this can be seen in figure 3.7a (file: sensor_box.stl), designed in AutoDesk Fusion 360 (Computer

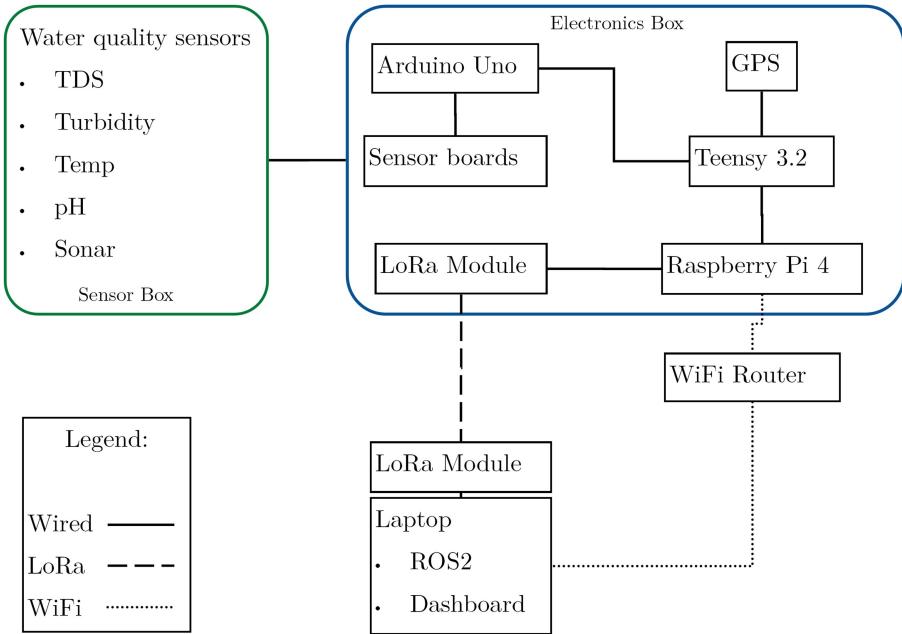


Figure 3.6: The schematic overview of the sensor array

The main parts in the sensor system are the following: the sensors touching the water; the sensor boards connected to a microcontroller: an Arduino Nano (“Arduino Nano”, n.d.); the Teensy 3.2 (“Teensy ® 3.2 Development Board”, n.d.), the GPS module (UBlox-M8N) (“NEO-M8”, 2021) connected to the Teensy; LoRa communication (Long Range, low power wireless communication protocol and hardware) (“Semtech LoRa SX1276 UART Interface Module”, n.d.); a Raspberry Pi 4B (a single board computer running Ubuntu 20.04) (Raspberry Pi Foundation, n.d.); a WiFi router (“Xiaomi Mi WiFi Router Mini”, n.d.), and the power delivery.

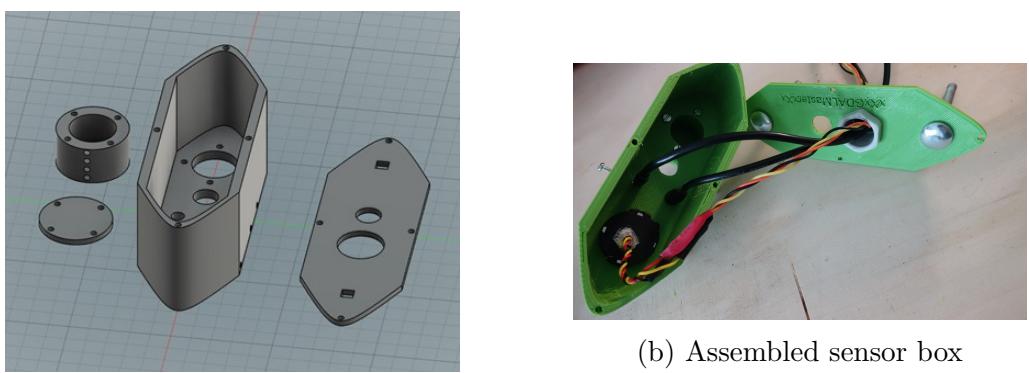


Figure 3.7: Sensor box design and assembly

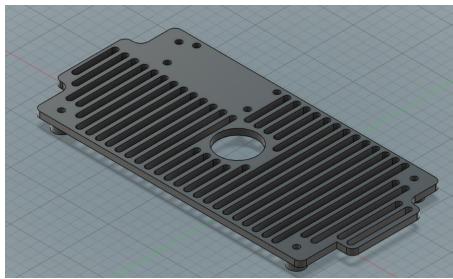
In figure a, the box itself, the lid on the right, and the sonar-mount is shown. The holes in the box serve to mount the sensors, the holes in the lid are for the mounting screws and the cables to pass through. The sonar mount ensures a solid connection between the sonar and the sensor box.

Everything can be screwed together with M3 bolts.

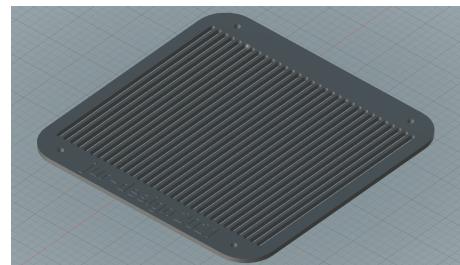
Assisted Design software) (Autodesk, 2021).

Required features were the following: 1. the ability to securely connect the sensors; 2. make sure as little as water as possible enters the enclosure; 3. create a mounting system that can be used on a variety of platforms.

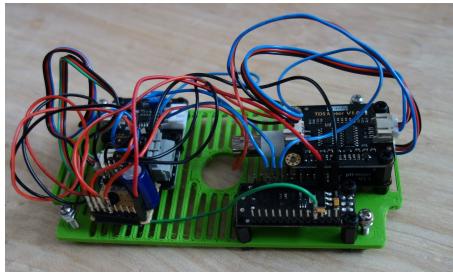
This resulted in the boat-shaped box design. Perfectly sized holes in the bottom were designed to fit the sensors. The side-mounted screw-holes function to ensure the sensors stay in place. The lid also has holes to ensure the pass-through of the longer pH sensor and the wires. It has square holes to accommodate M6 threaded nuts and bolts for mounting. All the wire ends of the sensors were terminated using GX-20 connectors to ensure the waterproof connection between sensors and electronics.



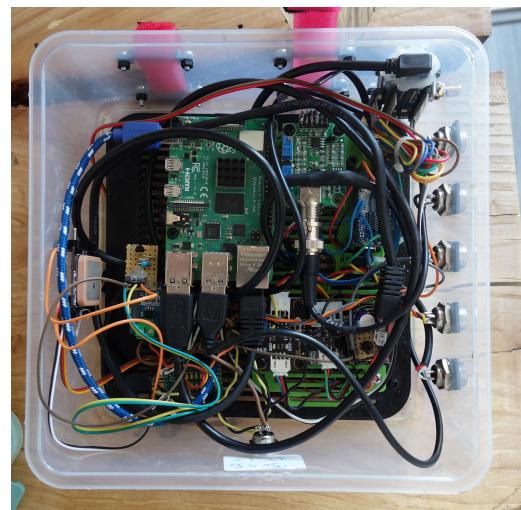
(a) CAD design of the sensor electronics plat-



(b) CAD design of the electronics assembly



(c) Final assembly of the sensor platform



(d) Final assembly of the electronics box

Figure 3.8: Electronics box design and assembly

Figure a shows the sensor electronics platform which houses the sensor boards and the Arduino Nano. In a similar fashion, figure b shows the platform housing the sensor platform and all the other electronic parts. These platforms are 4mm in width, with lines across. These lines across mount the different boards with standoffs and screws, but also makes sure that parts are replaceable. In figure c, the different sensor board, power distribution and the Arduino are mounted and wired. This is then placed on the larger platform. The larger platform can be placed directly in the main electronics box in figure d, with the GX-20 connectors on the right hand side. The wiring done in this image can be found in figure 3.9.

The electronics box was an off-the-shelf food container which can resist some water (figure 3.8d). To facilitate the various electronics-boards, two mounting surfaces were designed and 3D-printed: the sensor board platform (figure 3.8a) and the main assembly platform (figure 3.8b).

On the sensor board platform (figure 3.8c) the Arduino Nano and the four different sensor-processing boards were placed. Power to these sensors is being delivered via the internal 5V header on the Arduino, distributed via a distribution board with a 1000uF capacitor eliminating any potential noise. All connections were made with pin-header cables, directly soldered and secured with additional hot-glue.

The main assembly in figure 3.8b facilitates the sensor board platform, the Raspberry Pi, Teensy GPS, LoRa platform and the internal power distribution. On the walls of this box the GX-20 connectors and the LoRa antenna connector can be found, right hand side in figure 3.8d.

The power box has a XT-60 and Dean connector for powering from a LiPo-battery, and sends out 5v into the sensor box.

3.2.2 Wiring

The electronical parts in the system require power and data connections. This sub-chapter will introduce the wiring diagram in figure 3.9 and go over the connections between the components.

Power

Each component required power, however different parts require different voltages. The power rail starts at the LiPo battery at the outside of the case (right-middle in 3.9). The battery power is therefore allowed to vary from 7.4v to 18v, as it gets converted to 5v by a buck converter (XL4005). This 5v rail enters the box via a GX-20 connector and gets send to the Raspberry Pi (bottom-middle in 3.9), Teensy (top-right in 3.9) and Kogger Sonar (bottom-left in 3.9) and Arduino Nano (middle-top in 3.9).

The decision to power these from the XL4005, was made because these modules require the most power and they were not restricted to voltage regulators built inside the Raspberry (0.5A per USB port) or Arduino (0.5A total).

The sensors use the 5v regulated signal from the Arduino, via a distribution board (top-left in 3.9). The GPS gets power from the regulated 3.3v from the Teensy. The LoRa module is powered via the USB port from the Raspberry Pi.

Data

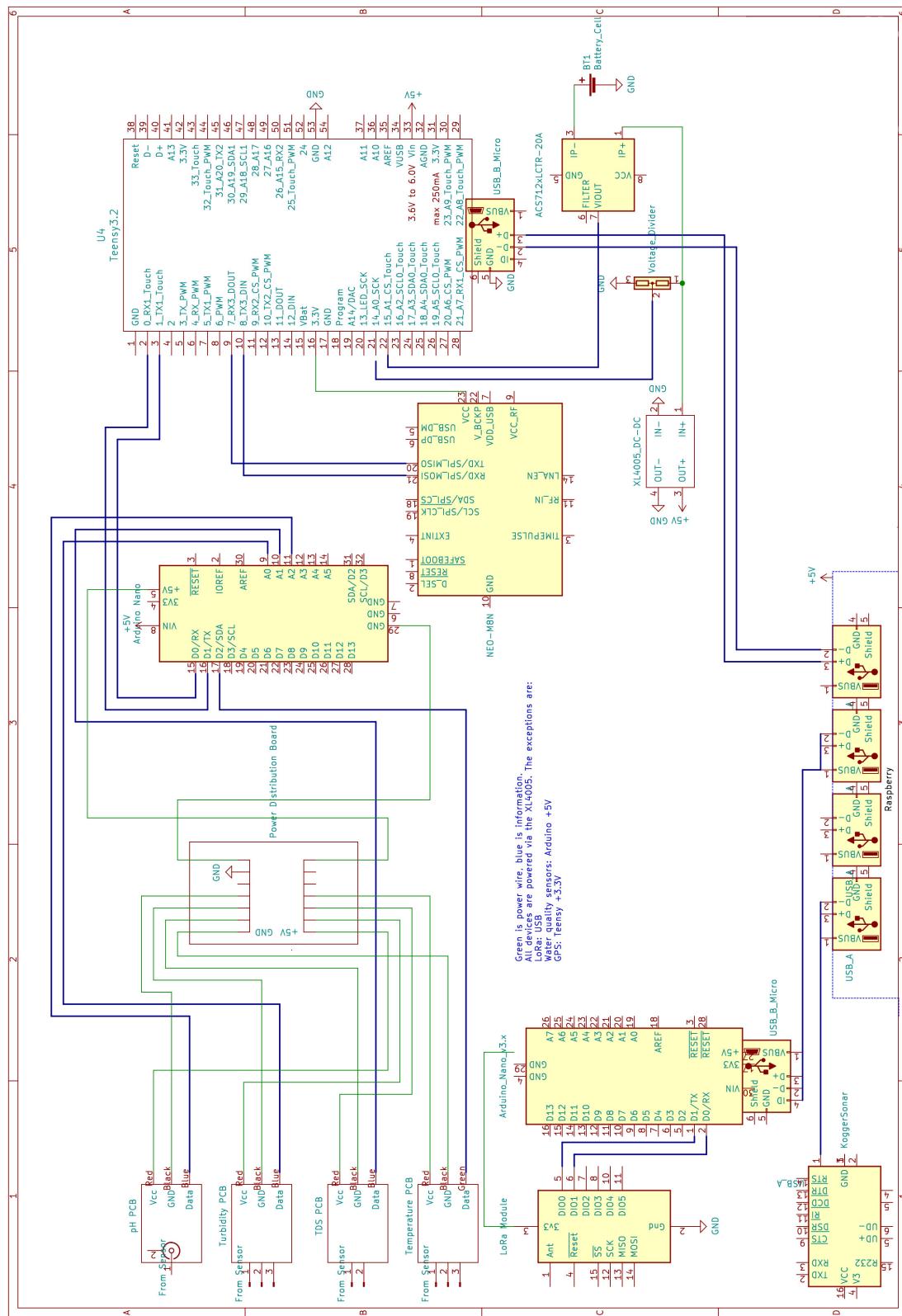
Data transfer occurs between the Teensy, Arduino, sensors and Raspberry Pi.

The turbidity and pH probes use analog signals between 0v and 5v to communicate the incoming turbidity and acidity levels to the Arduino Nano. The Arduino has a built-in analog-to-digital converter (ADC) converting these analog voltage readings to digital bytes.

The TDS board has a digital signal to the Arduino. It is digital due to some storage and conversion taking place on the TDS board.

The Temperature probe uses the 1-Wire digital protocol (Maxim Integrated, 2019).

The Arduino was connected to the serial input on the Teensy. The Teensy also handles the incoming GPS serial signal, and sends this information out over USB to the Raspberry (with the power from the USB cable cut off). The Sonar was connected to the Raspberry via USB, where a USB to UART converter does the



conversion of logic speed and logic level. Finally, the LoRa module exclusively sends its data directly to the USB port of the Raspberry Pi.

3.2.3 Microcontroller programming

As mentioned before, there are two microcontrollers in the system, the Arduino and the Teensy. Both can be programmed to execute a few tasks, repeated over and over again. An Arduino is powerful enough for light on-board processing. The implementation here takes a reading from the sensor, stores it in an array, and takes the mean of that array, to be sent out over the serial port on the Arduino.

The sensors all acquire their data slightly differently. For the temperature probe, the function `getTemp()` is used, which reads the actual temperature, as signalled over the OneWire connection to the Arduino and converts it from hexadecimal to a temperature reading. The TDS library from DFRobot has a built-in function which samples, and adjusts the acquired value for the temperature of the water (“Gravity: Analog TDS Sensor/Meter for Arduino”, n.d.).

The turbidity and pH voltages are read from the analog pin and converted to a digital reading. This means that the exact linear relation of the turbidity and pH voltages to their corresponding values still needs to be calibrated.

With all the samples in the array, sampled at every 0.04 seconds for 0.8 seconds, it takes the mean of this array. This filtered value, is printed over the serial connection, by formatting it as “Sensor: xx Value;”. This is repeated for all the sensors. This simplifies understanding the incoming data stream onto another system. This string of information is started and ended with a < or > to denote the end of the line.

The Teensy has two functions: The first is to read the incoming sensor-array information, and the second one is to read the GPS information. All are coming in via serial ports. This is sent to the Raspberry Pi over USB.

The GPS information is parsed using the NeoGPS library (“NeoGPS”, 2018). Which is a low-resource intensive Arduino library for reading GPS data, made for a variety of GPS receivers. It also comes with a variety of default scripts to use. One of these default scripts was used for the Teensy, with some slight alterations. This script reads the incoming NMEA GPS data and publishes it over the USB port.

Reading the incoming water quality serial data is done by finding the start and end markers of the serial line (< and >), as programmed in the Arduino Nano. After it has found these, it prints the whole information over the USB port.

3.3 Calibration

Calibration is the process of measuring values from a device, by comparing it to a device that has a known accuracy.

The sensors are not ready to measure out of the box. The chosen water quality sensors require programming and interfacing, which is explained in the system overview section. This means that changes in voltage from the power sources, software filtering decisions, and even wire distance can cause issues. Meaning that these sensors are unreliable. To understand the extent of reliability, and thus validity of these sensors, calibration was required. In this thesis, water quality sensors from the Water Quality Chair group at the WUR were used as the devices with a known accuracy. The features of these field meters are presented in table 3.1.

Table 3.1: Specifications of the field meters

Features	TN100	pH320	Cond315i
Range	0 – 2000 NTU	–2.0 – 20.0 pH	0 – 500 mS/cm
Resolution	0.1 NTU	0.1 pH	0.1 mS/cm
Accuracy	±2 % (0 – 500 NTU)	0.1 pH	±0.5 %
Response Time	≤ 6 seconds	15 seconds	10 seconds
Operating Temperature	0 – 50 °C	–10 – 55 °C	–10 – 55 °C
Cost (EUR)	600,00	600,00	400,00 (used)

The resolution of these meters change on what kind of precision it is measuring, the Cond315i for example has an auto-ranging function, where the measurement can go all the way down to 0.01μ Siemens. This means that the accuracy of these meters is also based on the value of the parameter it is measuring. Values are taken from the manufacturer websites and operating manuals (“Thermo Scientific Eutech TN-100 Waterproof Turbidimeter”, n.d.; WTW, 2004, 2008). The costs were mostly hidden, due to it being price-on-request products. Estimations have been made using the numbers available.

These field meters are well-tested out of the factory, and have regular re-calibration. The accuracy and resolution values are determined by the range in which it is measuring. For this table, middle values of accuracy and resolution from the manual have been chosen to indicate that these meters are accurate enough for outdoor water quality sampling. These meters were not made with high laboratory precision in mind, where 0.01μ changes matter. The idea of these meters, and the water quality sensors in the system, is that they can give a broad overview of the state of a pond. These field meters will therefore provide enough accuracy to represent this.

The temperature and TDS sensor give digital information to the microcontroller. Whereas the turbidity and pH return voltage readings, to be converted to water quality measurements. These last two therefore first require a two point calibration. This calibration is done by measuring the voltage in two known calibration solutions. Then calculating the slope and offset of these voltages in order to convert them to actual values. This is done by using the two equations below.

$$\text{Value} = \text{CalibrationHigh} - (\text{VoltageHigh} - \text{Observedvoltage}) * m$$

$$m = \frac{\text{CalibrationHigh} - \text{CalibrationLow}}{\text{VoltageHigh} - \text{VoltageLow}}$$

3.3.1 Process

Table 3.2: The content of the solutions

Temperature	TDS	Acidity	Turbidity
Tap water	<i>Tap water</i>	Tap water	<i>Tap water</i>
Outdoor water	10 salt crystals	<i>4.01 pH solution</i>	1:10
Fridge water	20 salt crystals	<i>7.0 pH solution</i>	1:20
	<i>30 salt crystals</i>	2 droplets lime-juice	1:40

The salt crystals were mixed with 100mL tap water. The lime juice mixed with 100mL tap water. The 1:10, 1:20 and 1:40 refer to the ratio between white acrylic paint to tap water: 1 part paint to 10 parts water, etc. The cursive values are the high and low points of the initial calibration.

The corresponding EC, pH, NTU and temperature values from the solutions in the table above were measured with the field meters.

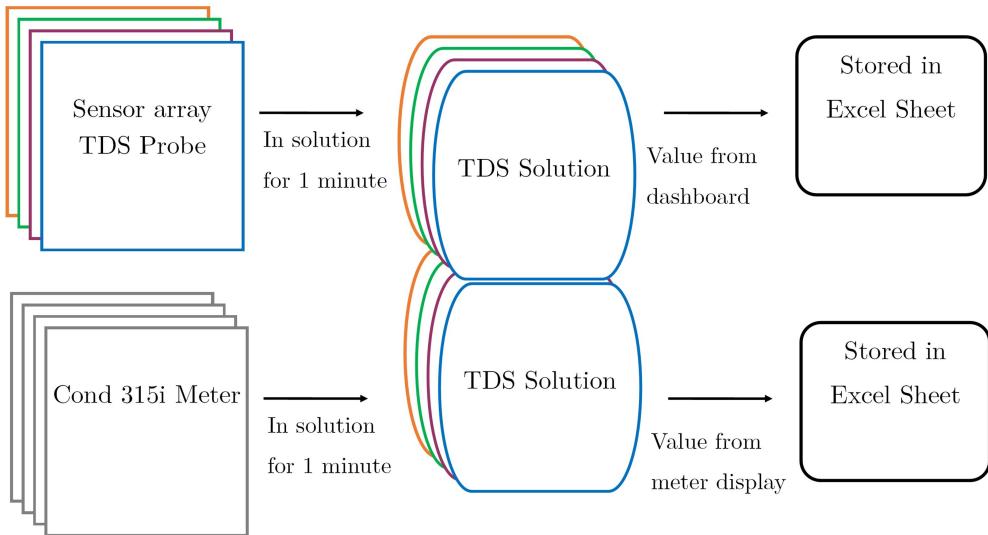


Figure 3.10: Flowchart showing the steps executed in the calibration

The different sensors on the array all went through calibration, in the exact same manner. The field meters were used to measure the exact parameters, such as the level of acidity and assumed to be the truth value. This truth value was compared to the value that the array sensors would measure.

Then, calibration was executed using the steps as can be seen in figure 3.10.

First, the voltages for the high and low calibration values were measured. This set the baseline for the calibration for each of the meters, apart from the temperature probe, as this already present the degrees Celsius. TDS was converted to EC this way, and the voltages to pH and NTU. This was done using the equation above.

With the calibrated slope and offset in place, the other test-samples were sampled. For pH this was done by placing the probe in the solution for 30 seconds, as the sensor requires time to settle.

The turbidity sensor was placed in a small container, where the two prongs could fit, and most of the external light was covered from hitting the photo-voltaic sensor. For TDS, the sensor was placed in the solution for 30 seconds, as the TDS is dependent on temperature.

For the temperature probe, it was placed in the solution for around 30 seconds, waiting for the temperature to settle.

This process resulted in an Excel file. In this excel file, the solutions, and their corresponding field meter truth values were saved in tangent with the measurements from the values measured by the sensor array. This data was analyzed using Python Pandas (The pandas development team, 2020).

3.4 ROS

This subchapter presents the implementation of the whole system into ROS (Robotics Operating System) (Open Robotics, 2021). Which opens up the measured water quality data points for the broader USV system. The overview of the ROS architecture can be found in figure 3.11.

The basis of ROS, is the ability to create or use previously made nodes. These nodes can be started, and will execute a task. Such as publish data from all sorts of

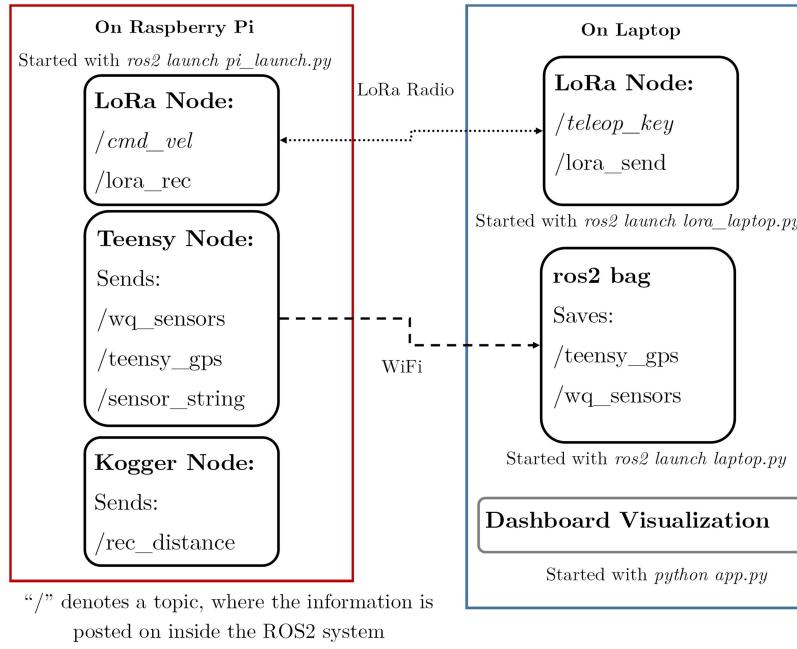


Figure 3.11: Overview of the ROS nodes running when the sensor system is active

The diagram shows the system split into the Raspberry Pi running nodes, and the laptop running nodes. The cursive nodes are the ones that are standard ROS2 packages, while the others are specifically made for this project. The messages that these nodes produced are published on the /topics. These topics can be subscribed to by other nodes. As long as they are in the same network, in this case a WiFi network. The LoRa nodes produce control messages in case the WiFi network is out of range, thus serving as a backup line of control.

sensors, it could be a service to parse the information to another format. But the most important function that ROS has, is handle callback timers and synchronicity, next to having a wide array of default data formats (“common_interfaces”, 2021). All of this is done inside C++ or Python written nodes. Data used in these nodes is published on topics.

ROS requires a complete operating system in order to work. In this case, the Raspberry Pi, running Ubuntu 20.04 and ROS2, was chosen for this task. The Raspberry is connected to the Teensy, the Teensy is constantly sending GPS and water quality information over USB. The Raspberry also interacts with the sonar, this sonar sends depth information. In the ROS system, the water quality measurements, sonar observations and the GPS all have their own nodes.

The nodes have been written in python, and mainly uses the pyserial package (Liechti, 2021), which handles the incoming data streams from the USB port.

3.4.1 The Teensy Node

As presented in the microcontroller programming subsection, the water quality measurements and GPS observations from the Teensy are sent over serial to whatever is connected on the other side of the USB port. This means that the ROS2 node only needs to reads the serial input and publishes it as a String (text) topic on ROS2.

Then there are two other nodes, a gps and a water quality parser. These publish to the `\teensy_gps` and the `\wq_sensors`.

For the GPS, it takes the whole line, converts it from bytes to a string, and splits

it on the “,” separator. Each of the items in this new list corresponds to one of the NMEA protocol datapoints, as created by the NeoGPS library on the Teensy. This information can be parsed into correct formats. The latitudes and longitudes on the WGS-84 coordinate system have seven digits of floating-point precision. This information is published with the default NavSatFix message from ROS (“common_interfaces”, 2021), this message predefined that there is a latitude, longitude and altitude and some covariances. This is then published on the \teensy_gps topic.

For the water quality sensors. The incoming line is read to its completion (line end /n) and then converted to a python string. This string is parsed using regular expressions, to acquire all of the numbers from the different sensors. Regular Expressions are a robust way to extract information from a string. These extracted numbers are then converted from strings to floating-point numbers. These numbers are placed in a list, and this list is published on a water quality sensors topic. It uses a Float64MultiArray default message from ROS2 (“common_interfaces”, 2021).

3.4.2 The Sonar Node

The Sonar node reads the incoming binary data from the sonar at USB port, and publishes this as a UInt32 message on a ROS2 topic.

The Kogger communication protocol is structured as a frame (Kogger, 2020), the frame contains different sections, and one of these sections is the actual depth information. Which means that the sonar node needs to read the incoming stream, stop at the correct length, split it up into the different sections, extract the depth information and post it on a sonar topic.

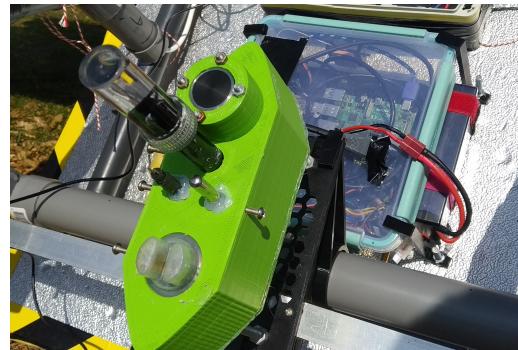
The first step is to open the serial port and await incoming information. Whenever a byte is coming in, it is placed into a string, the payload. The length of this string is always the same, as the distance-frames are always the same length. By checking whether the length of this payload is the same length as the distance-frame the total payload can be acquired. If the data has completely arrived, the distance information (in bytes) is converted into the unsigned 32-bit integer, representing the millimeters of depth that the sonar has measured. All of this is done each time the ROS2 timer fires, and this sequence runs again.

These nodes are started on the Raspberry Pi simultaneously, using a ROS2 launch command, with a set of launch parameters, eg. which node to launch, how to call the topics used in the nodes, and which USB ports the node should open. All the nodes start, and publish data on the ROS2 topics, available for application in any other ROS2 system.

In this case, a laptop on shore subscribes to the relevant topics, and saves the messages in a SQL database (Hipp, 2020), using the ros2 bag command. This database can be replayed later, or stored and accessed later on to do visualizations and analysis. The whole water quality sensing system is started and monitored from this base-station laptop. As long as the WiFi network is the same as the Raspberry Pi, everything can be controlled with this laptop, the only operation is turning the power to the electronics box on.



(a) Updated version of the USV



(b) Underside mounting of the sensors

Figure 3.12: State of the system before starting the experiments

Figure a shows the state of the USV at the time of doing the experiments. The main change in comparison to the USV presented in Borreguero et al. (2018), is skid steering. This method of steering has two motors on either side of the frame, when they move in opposite directions, the USV turns. When they move in the same direction it goes forwards, or backwards. This makes it better in holding its position in comparison to the previous generation of this USV. Figure b shows the location of the sensors probes, in the green box, and the electronics above it in the transparent container.

3.5 Experiments

The system was tested on a water surface. This was done to show how the water quality sensors worked in a real life setting and to explore spatial differences in water quality parameters. The system was mounted on an Unmanned Surface Vehicle, capable of carrying a payload. This USV is presented in Borreguero et al. (2018), more recently it has gone through a rework. This was done in tangent with the development of the water quality sensors. This new version is presented in figure 3.12a. At the time of writing, the rework added skid-steering and ROS control. However, it is still under development and waypoint following, and UAV landing capabilities are still being developed.

Setting up the whole system starts by mounting the sensors to the USV, with the use of M6 bolts. The electronics box fits with clips to the underside of the hull of the USV. The battery can be connected to the power connector on the power box, which turns on all the electronics on board. With a WiFi router also powered on, the Raspberry Pi, and thus ROS, can be accessed via a SSH connection (Secure Shell, a remote connection to another computer). From here, the Raspberry Pi nodes can all be launched with a single launch command. Finally, on the laptop a ros2 bag command can be started, which stores the water quality, GPS and sonar topics into a database. This data is used in a monitoring dashboard.

The dashboard was created in Plotly Python, a dashboarding framework for Python (Hossain, 2019). This reads the database, visualizes the temperature readings in a spatial map of points. The pH, turbidity, TDS and depth are visualized as time-series plots. The most recent observations are shown as floating point numbers (figure 3.13). This dashboard updates every 4 seconds, which correlates to around 2 or 3 new points being measured at each update.

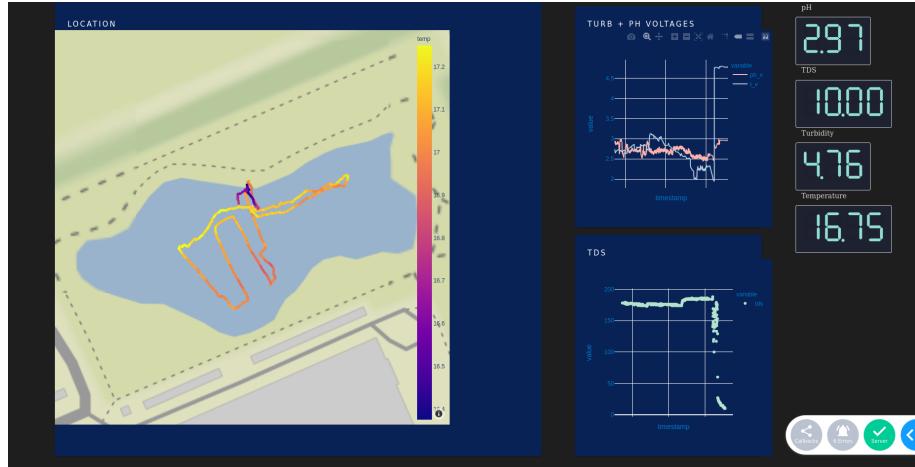


Figure 3.13: Dashboard screenshot for live monitoring

The dashboard shows a live map of the current location of the USV. The plots on the middle-right show the last 50 observations of the different sensors, over time. On the right side, the latest observations of the sensors are presented. This dashboard accesses the ros2 bag information and visualizes it directly. It was built on the software by Plotly Dash (Hossain, 2019).

3.5.1 Mapping the GAIA lake

The used lake is a small, shallow, artificial pond, behind the Wageningen University Lumen/GAIA building (WGS84: 51.988408; 5.666714). This pond was chosen because of the ease of accessibility.

The USV was placed in the water and until all the sensor values have settled, the system is kept idle. After idling for around a minute, the ros2 bag command was started and the USV started storing the water quality information on the laptop.

The goals of the experiments on the GAIA lake were two-fold. The first was to improve the system itself, from physical robustness, stability and software reliability. The goal was to make sure a complete battery could be emptied before something in the system would fail. The second side was to spatially measure the GAIA lake, with the GPS coordinates linked to the water quality observations. This data would be useful for later analysis.

The first tests were executed in February through March 2021. In this iteration, the sensors were on its own floating platform. This floating platform was pulled along by the USV, shown in the second image of figure 3.14. However, the stability of this system would not be able to survive a larger wave. Furthermore, the electronics box was not large enough to house all the final parts, and the connections from the sensors to the board were permanent. These findings led to the creation of the larger electronics box, with the option of underside mounting, labelled GX-20 connectors and increased waterproofing of the whole system.

This renewed system was tested in March. Mainly focused on resilience of the system. The system needed to be able to send information throughout the run, without overheating, power loss, connection loss or some system-breaking bug. These tests resulted in the external battery box, a replacement of Raspberry Pi and changes in ROS for the string parsing of the Teensy. Performance improvements to the dashboard were also made, increasing the visualization rate.

All these changes led to the first complete test on the lake. In the experiments of April, the USV was manually controlled to grid out the water body. The GPS

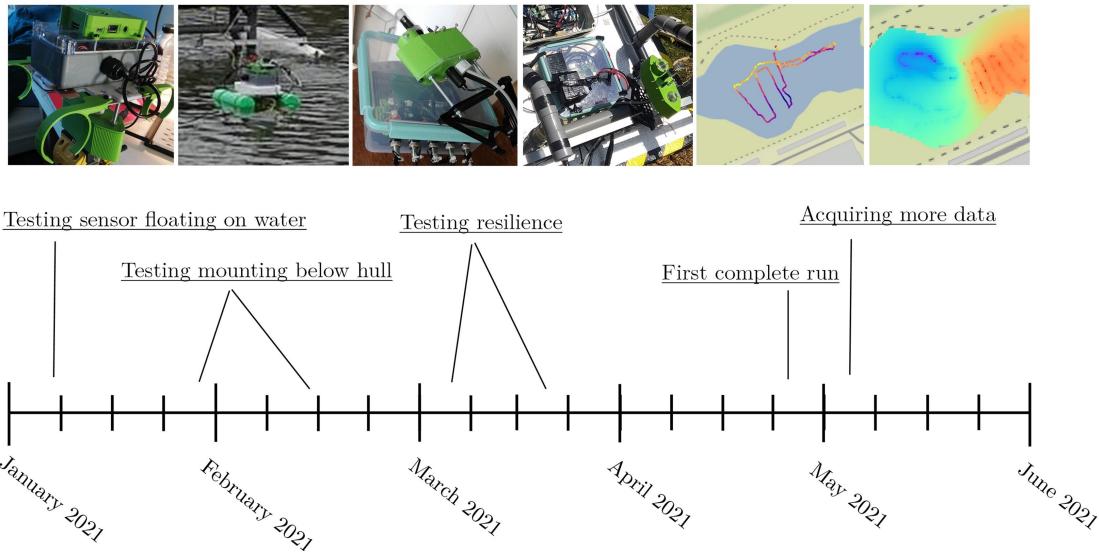


Figure 3.14: Timeline from the first half of 2021, when all the experiments on the GAIA lake took place

The images on top show the state of the system at that point. This shows the evolution of the system during these experiments. Until the point that it was capable of working until the battery would run out.

locations and the number of points are shown in the second to last image. The results of this first complete test clarified the need for more data in order to create interpolations of the pond.

On tests halfway through May, the data acquisition was repeated. With slower movement and more systematic sampling. Added was a second test run where the USV stood still for around 1 minute on different locations in the lake. In order to understand differences between continuous slow movement sampling and the position hold method.

3.5.2 Basic processing

The acquired data was processed in Python. Just as the dashboard, the data was read from the SQLite3 database, and reformatted into Geopandas dataframes (Jordahl et al., 2020).

This was done by first extracting all the raw data from each topic in the ros bag, converting the byte-arrays created by ROS2 into python usable integers, strings or floating point numbers, and placing them into their own timestamped lists. The data was harmonized by taking the nearest timestamp where there is a GPS point available. This creates a geolocated, timestamped observation for each sensor in the system.

In the other testing of the pond, a sampling hold method was used to obtain the information. In this method, the USV stood still in a variety of sections of the lake for around 1 minute. All the points where the change in GPS coordinate was large, meant movement. These points were removed. The distances between points were calculated once more. All the points without a large gap in between them were clustered into a single point. Using the mean location and mean value of

the sensor on that cluster of points. This whole pipeline is available on the thesis Github (“jurriandoornbos/wqdrone”, 2021).

3.6 Interpolation

The water quality data is attached to the GPS observations. This results in water quality observations with latitude and longitude information. This format is not very useful for water dynamics, as a water is a continuous surface. This requires that the different points are placed onto a continuous grid, a raster.

Creating this raster requires interpolation. Interpolation is the general term for inferring information between two, or more, observations by just using the available information. There is a large variety of methods available.

An important distinction can be made between the fast method of Inverse Distance Weighting (IDW), and a stochastic method of kriging. The basics for both methods analyze the distance between the known observation and the unsampled location. IDW weighs all the known observations to a linear distance from the unknown point (Murphy et al., 2010). Kriging is built on statistical assumptions underlying the data, and is therefore a parametric approach to interpolation. There are a variety of models available in kriging. However, research has shown that the best method is study and site specific (Murphy et al., 2010). This requires validation data and methods to understand the accuracy of the interpolation. The IDW python implementation from geosardine was used (“geosardine”, 2021) to interpolate the points.

3.6.1 Inverse Distance Weighting

IDW is based around the following function from Shepard (1968):

$$\lambda_t = \frac{1/d_i^p}{\sum_{i=1}^n 1/d_i^p}$$

In this function, the weights of each point are calculated, which is the inverse distance to the known point. d_i is the distance to the other points in the dataset. p is the power parameter and n the number of known points. The power factor has the highest influence on the results. As this determines the strength of the nearest points. As the distance increases, this effect is reduced, making the reduction scaled by p . According to Li and Heap (2008b), the value of p is arbitrary, usually kept at 2, called Inverse Distance Squared. However, it can also be found with error measurements techniques, such as a cross-validation. The resolution of each cell is around 30 by 30 cm. This resolution was set in order to best define the outline of the GAIA lake, not resulting in large squares, while not taking a long time to process.

3.6.2 Sampling

Different sampling methods will result in different interpolation results, and could require different models (Li & Heap, 2008a). In order to keep the sampling and interpolation defined, IDW will be applied to two sampling methods, the hold method and the continuous method.

In the hold method, the USV will hold its position for around 1 minute, and sample a variety of sections.

In the continuous method, it will slowly move across the whole pond.

3.6.3 Accuracy

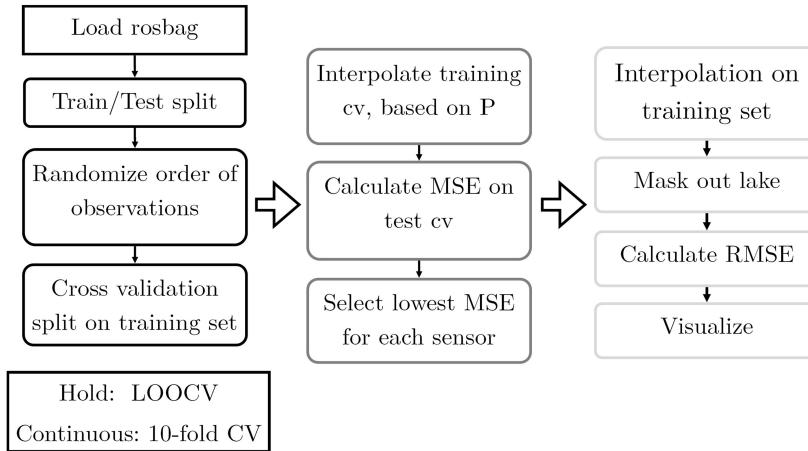


Figure 3.15: Interpolation pipeline

The process is shown here in three large steps. The first is pre-processing, the second is the cross validation, and the last one is the final interpolation. Pre-processing consists of loading the data, and creating the initial training and testing split of the data. The training data is then split once more, after the order of observations have been randomized. This subset of the training data is placed into the cross validation. In this cross validation, different configurations of IDW parameters were tested. From these interpolations, the MSE was calculated. The IDW parameters with the lowest MSE were chosen to create a final interpolation. These final results were masked to an outline of the lake and visualized.

The final step in the interpolation process was accuracy assessment. Due to the modelling nature of interpolation, techniques from statistical learning can be applied to understand the model performance (Li & Heap, 2008c). The interpolation pipeline is presented in figure 3.15.

The two sampling methods have been analyzed using two different cross validation techniques: leave one out cross validation (LOOCV) and 10-fold cross validation. In cross validation the dataset is split into a training and testing set, not once, but multiple times. Each set consists of a different subset of the data. This reduces the random error that can exist when using a single split to assess error, as the different subsets of the data ensure that this randomness evens out (James et al., 2017). The size of each subset k depends on the method used. LOOCV takes a single observation for a hold out set, and uses the others for training. This means that most of the data is used in training the model, which creates a better model. In general LOOCV is best used for smaller datasets where computation time is low.

For the continuous observation method, 10-fold-cv was used. In this method, the input data is split into 10 parts, where 9 parts are used for training, and 1 is a hold out set. This is more applicable to the larger data-set, which is more computationally expensive for the IDW algorithm. To minimize the time of calculation, LOOCV is not applied, as it would result in n computations of the IDW algorithm. The created

interpolations would then be tested on the hold out set, and the Mean Squared Error (MSE) would be calculated (middle part in figure 3.15), to find the best value for p . The values 1, 2, 3, 5 and 10 were used for p . MSE is calculated by sampling the interpolated map at the hold out point, and subtracted from the actual value that the hold out set has for that point. This is then squared, and finally the mean is taken from all the squared errors.

This is then repeated for each training/hold out set, and the mean is taken from that. This is repeated for each value of p . Resulting in a set of MSE values for each value of p , in which the lowest MSE shows the best value for p . This is then placed into a final interpolation on the whole training set. RMSE was calculated by taking the testing dataset observation and taking a sample from the interpolation raster, this sample is then directly compared to the testing observation. In the formula below, the y_i are the truth values from the testing dataset, and x_i are the interpolated samples.

$$RMSE = \sqrt{\left(\frac{1}{n}\right) \sum_{i=1}^n (y_i - x_i)^2}$$

Chapter 4

Results

In this chapter, the findings are presented. These findings explore the strengths and weaknesses of the sensor system. The findings are subdivided into two parts. First is the sensor quality, where the sensor calibration, sensitivity and costs of the sensors are presented. The point observations of the USV were made into raster images using interpolation. This was done in two methods, a continuous sampling, where the USV did not pause, but slowly moved across the water surface, and a location hold sampling, where the USV held its location at different points on the pond. The second section is about the interpolation results and the accuracy of those methods.

4.1 Sensor costs and battery life

Table 4.1: Costs of the sensor array

Part	Price (EUR)
Temperature sensor	6.56
pH sensor	25.81
TDS sensor	10.32
Turbidity sensor	8.66
Kogger Sonar	150.00
U-Blox NEO M8N GPS	15,00
Arduino Uno	10.00
Teensy 3.2	30.00
Raspberry Pi 4	60.00
XL4005 DC-DC converter	5.00
Wires	5.00
Total	316,35

The table is subdivided into three sections. First, the top section shows the cost of the sensors themselves. These sensors are highly affordable, compared to the meters used in calibration. In this study, the sensors are not expected to outperform these, but to function well enough to measure the result within an acceptable accuracy. Second, the middle section shows the logic devices used to create the observations and communicate in the wider ROS system. Third, the lowest section is about the external parts required to make the system work, such as wires and a power converter.

This paragraph presents the specifications of the sensor array, in terms of cost and battery life. Furthermore, the calibration results are presented in section 4.2.

The sensor array costs are presented in table 4.1. With the sonar taken out of the equation, less than 170,00 euro was spent on the whole sensor array. Making it considerably cheaper than the field probes and test-equipment to which these sensors were calibrated, for details see section 3.3. Part of these costs were also the Raspberry Pi, which serves to communicate the sensor information the wider ROS system.

The system was mounted on a catamaran style USV. During testing, the current draw was stable with current peaks around 1500mA. This means that with the 3300mAH, 7.2V battery used in the tests, the system will last at least 3 hours.

4.2 Calibration and accuracy

In order to understand the accuracy of the acquired sensors, a calibration was executed. This is presented in figure 4.1. This figure shows that the temperature sensor is the most accurate sensor, with the pH sensor showing the other best linear response. The TDS and turbidity did not perform well with the linear combination executed in the calibration.

A two point calibration was executed for TDS, turbidity and acidity. The temperature sensor was not further calibrated since it already returned calibrated values.

According to the calibration figure, the temperature probe shows the best results (figure 4.1a), where the measured temperature is closely related to the actual temperature, dotted line vs orange line. The RMSE means that 0.8 °C of deviation in the data remains unexplained.

The calibration for TDS consisted of recalculating the TDS value to electrical conductivity, in millisiemens. Interesting in the scatter-plot (figure 4.1b) is the non-linear relation between the observations. The sensors hit a ceiling in measuring at around 800 mS. Anything above 800mS will all show up as 800mS, making it unusable for higher value EC measurements. Values below 800mS seem to be usable.

The pH probe highly sensitive to small changes, as each step in pH is around a 100mV increase in voltage level in the water. This means that drifting in input voltage and temperature of the electronics and the water can affect the measurements a lot. The created water samples were hard to read for the meter. This can be seen on the right hand side in figure 4.1c, these two samples were 6 and 7 pH respectively, whereas the the calibrated probe measured around 7 for both. At the left hand side, the sensor is measuring too high. For this sample, the meter measured a pH of 4.5, while the sensor measured less than 4 pH. The pH sensor is the least reliable of the sensors, due to the above mentioned issues. Furthermore, the pH sensor has issues with drifting voltages and thus getting a reliable measurement. Drifting is where the values slowly drift away over time from their original point. The calibration points, high and low, were measured first. But over time, when sampling the other two solutions, the values are off.

The turbidity measures the amount of light coming in the sensor, at a specific wavelength. Calibrating this to a NTU meter, suggests that a non-linear relationship is present (figure 4.1c). This is seen in the scatter-plot. As turbidity of the sampled water increases, the measured voltage at the probe increases non-linearly. A linear transformation, such as the two-point calibration from voltage to NTU therefore does not seem to make much sense in this case. Controlling for external light is also crucial for this probe.

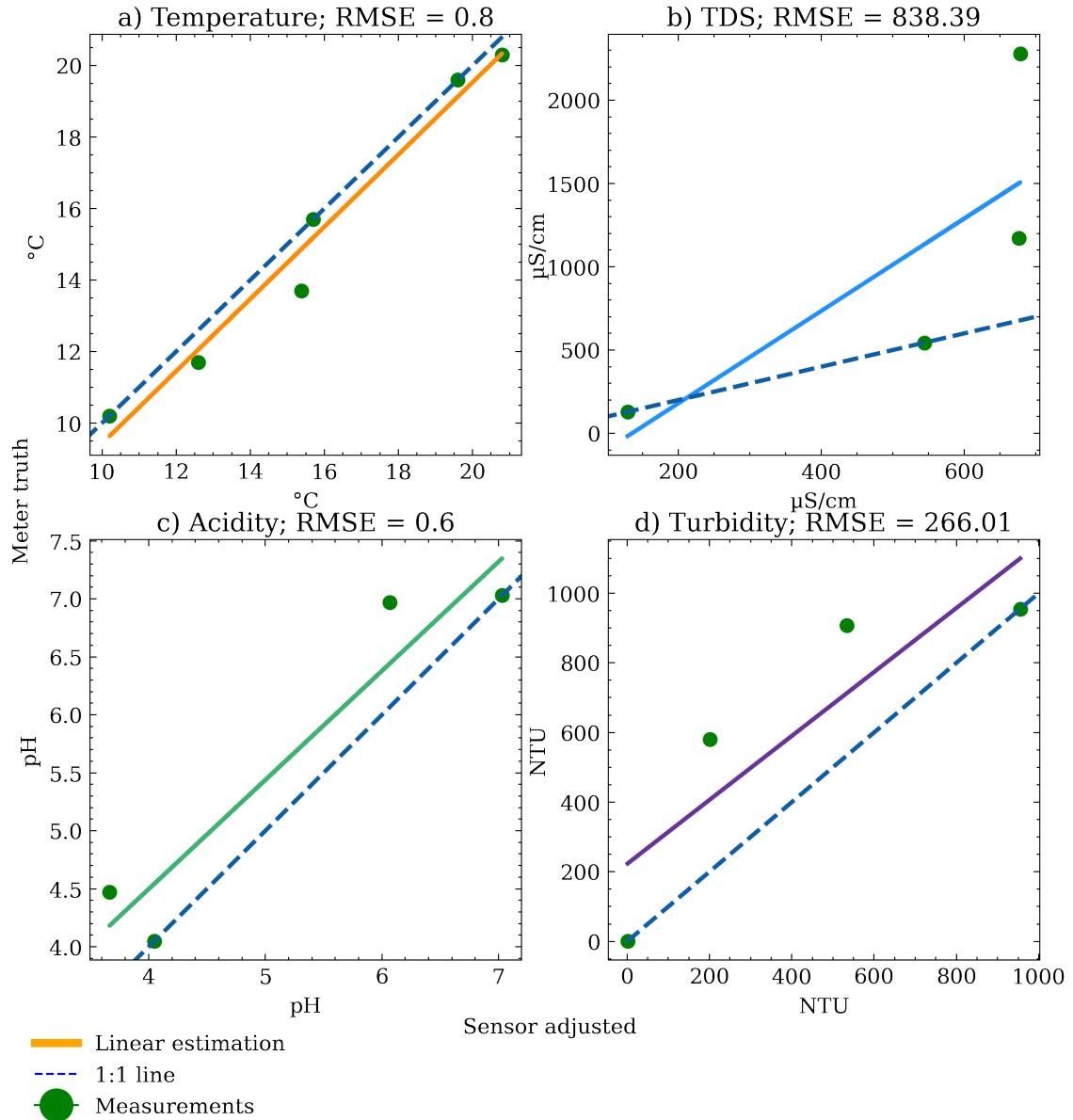


Figure 4.1: Results from calibration

The figure consists of four panels, each one representing another sensor. The X-axis shows observations for the sensor array, the Y-axis the observations from the meters, assumed as the ground-truth observations.

The scatter-plots are the measurements of the sensors compared to the meters. The coloured line is the linear estimation of the relation between the sensors and the meters. The dotted line is the 1:1 line, this is the expected relation between the sensor and the meter. The more different the linear estimation is to this dotted line, the less accurate the sensor is compared to the meter. The two-points in the two-point calibration are the measurements where the dotted line crosses the measurement.

A note on the sonar

When testing, the kogger sonar was not capable of measuring the very low depth that is present in the lake, at less than 50cm. This made using the sonar in these situations less than ideal. Furthermore, the sonar was sent back to Kogger for checking why it was not responding properly to turning on.

4.3 Interpolation

To get the most out of the sampled data, the values were interpolated on a regular grid. This was executed on two datasets, a continuous sampling and a location hold sampling method. This was done to understand how different sampling techniques can have an impact on the accuracy of the result. Some of the sensors require a few seconds to take a measurement, and continuous moving might inhibit this. The observations in these two datasets are shown in figure 4.2.

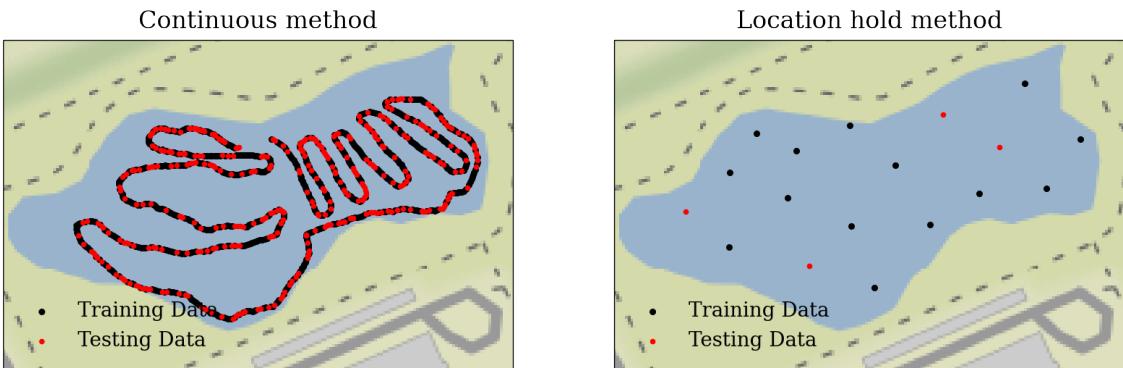


Figure 4.2: Train and test split of the point data

The two figures here are the two different sampling methods, visualized as the map of input points. The black points show the training dataset used in the interpolation, the red points are the points used in the accuracy assessment.

For each of these methods, a time-series, histogram, cross validation, interpolation and accuracy assessment is presented. First the continuous sampling method is presented, afterwards the location hold method is presented.

4.3.1 Continuous Sampling

In the continuous sampling technique, the USV continuously sampled the water as it went over the surface with a speed between 0.1 and 1 m/s (3.6km/h). The histograms of observations and the time-series are shown in figure 4.3.

The distribution of voltages between the pH and the turbidity sensors are different. The pH voltages are roughly between 3 and 3.2 volts, while the turbidity is between 2.6 and 3.8 volts. The figures in 4.3a, b, c and d, show the values of the probes over time, during the continuous sampling. The temperature probe varies (19.6-21.3 °C), while the TDS sensor shows signs of drifting upward continuously. The pH begins low but rises in the first minute. After around 5 minutes it drops

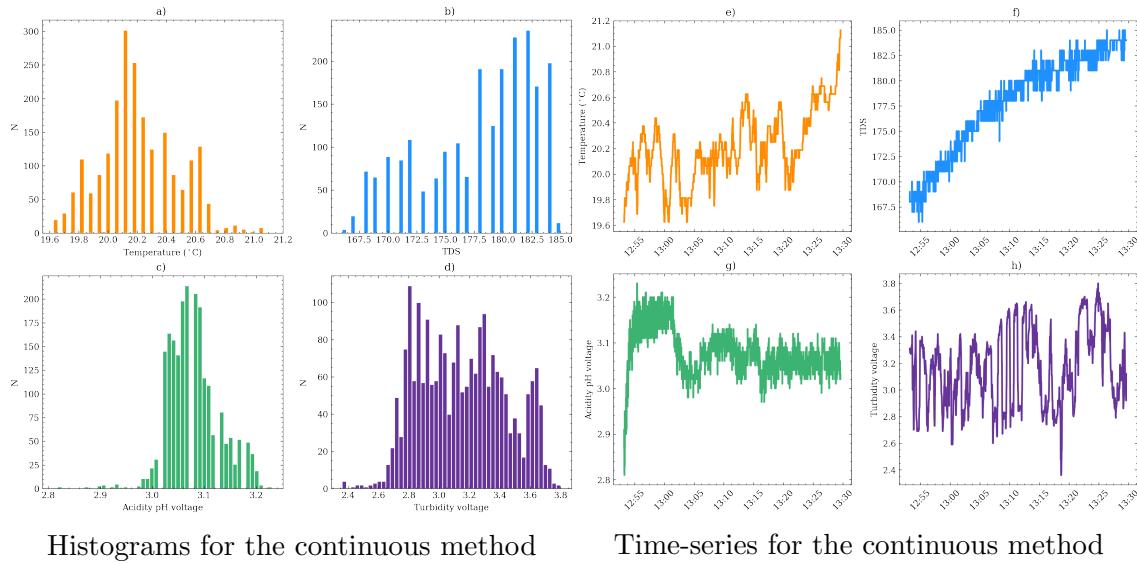


Figure 4.3: Contextual information continuous interpolation

The histograms show normally distributed data. The TDS and turbidity are not as clearly distributed as temperature and acidity. In the TDS and temperature histogram, clear bins are visible, due to missing observations between numbers. This shows that something from the microcontroller program was rounding it to specific numbers. The pH and turbidity show the voltage levels instead of the calibrated values. As seen above, when calibrating the voltages, the voltages, and thus the values, were not more reliable. It was therefore chosen to keep the voltage levels to show a relative change in pH between observations.

again where it remains stable. The turbidity goes up and down continuously, showing that the light coming in at the photovoltaic sensor increases and decreases.

The continuous sampling training data was placed into a tenfold cross validation, in order to find the best value for p for each probe. The results of the cross validation are presented in figure 4.4.

The power value p has impact on the slope between two points that are placed in the interpolation. The higher the p value, the stronger each points maintains its value as distance increases. The results show that a p value of 1 did not result in a low MSE. The optimum value was between 2 and 3. The turbidity shows a slightly different curve (figure 4.4d), rapidly increasing after 2. These results show that the turbidity observations are unusually distributed in space, and do not follow the same p value results of the other sensors.

With the cross validation executed, the p values with the lowest MSE were chosen to run the interpolation on for each sensor. This final interpolation is shown in figure 4.5.

In the interpolations the original data is still visible in the pattern of the interpolation. This pattern is partly due to the raster size that was chosen, so the original value is always kept in the overlapping raster pixel.

Furthermore, the interpolation shows that for TDS and pH (figure 4.5b and c) a large difference between the eastern and western side of the pond is visible. Temperature remains similar in both sides. Furthermore, the variation in temperature is a lot less ($19.9 - 20.3^{\circ}\text{C}$), compared to the histogram range of input data (fig 4.3.a). This shows that the extremes in the input datasets have been reduced by around 0.4°C on both the minimum and maximums.

In the turbidity interpolation, the values change between 2.8 and 3.4 volt for each

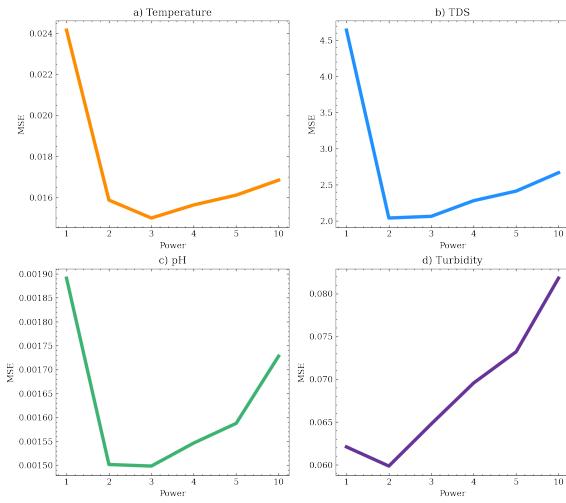


Figure 4.4: Cross validation results for the tenfold MSE search
The MSE values for different values of p . Found by executing a tenfold cross validation on the training dataset. This was done for each of the sensors in the system.

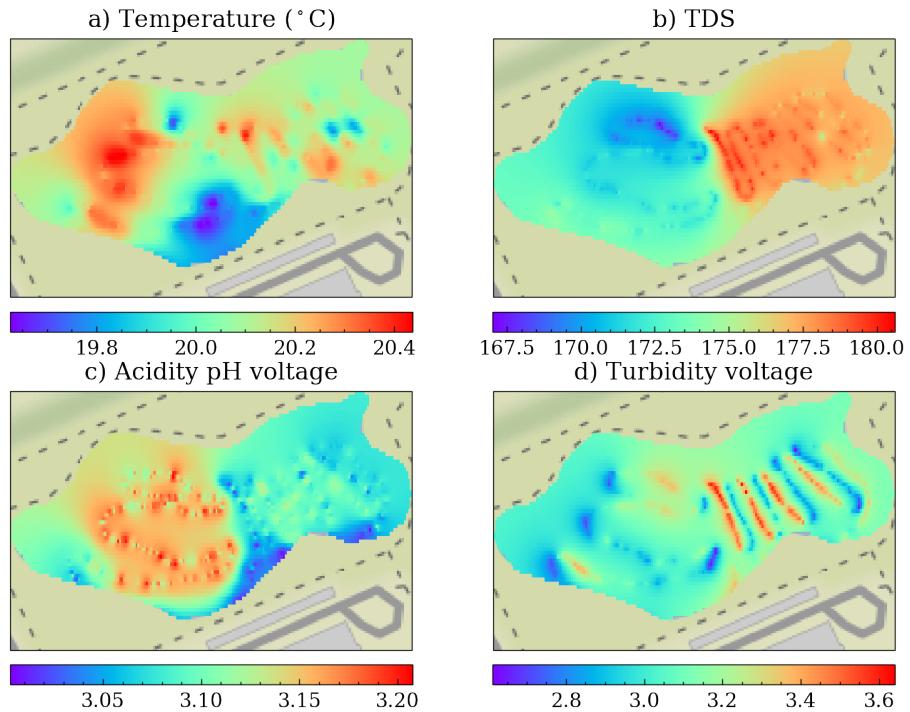


Figure 4.5: Interpolated final maps for the continuous method
In these figures the raster images resulting from the interpolation are shown. These rasters were masked with a polygon of the GAIA pond, and visualized with a basemap. All of the interpolation clearly show the underlying route that was followed.

path. According to the calibration, this is between 800 and 600 NTU respectively, which is a larger difference than is expected for the pond. This means that there was a large variation in input light between the forward and backwards path. This was caused by the sun directly shining onto the photovoltaic sensor. It shows that in-situ turbidity measurement is very sensitive to the direction the turbidity sensor is facing in relation to the sun.

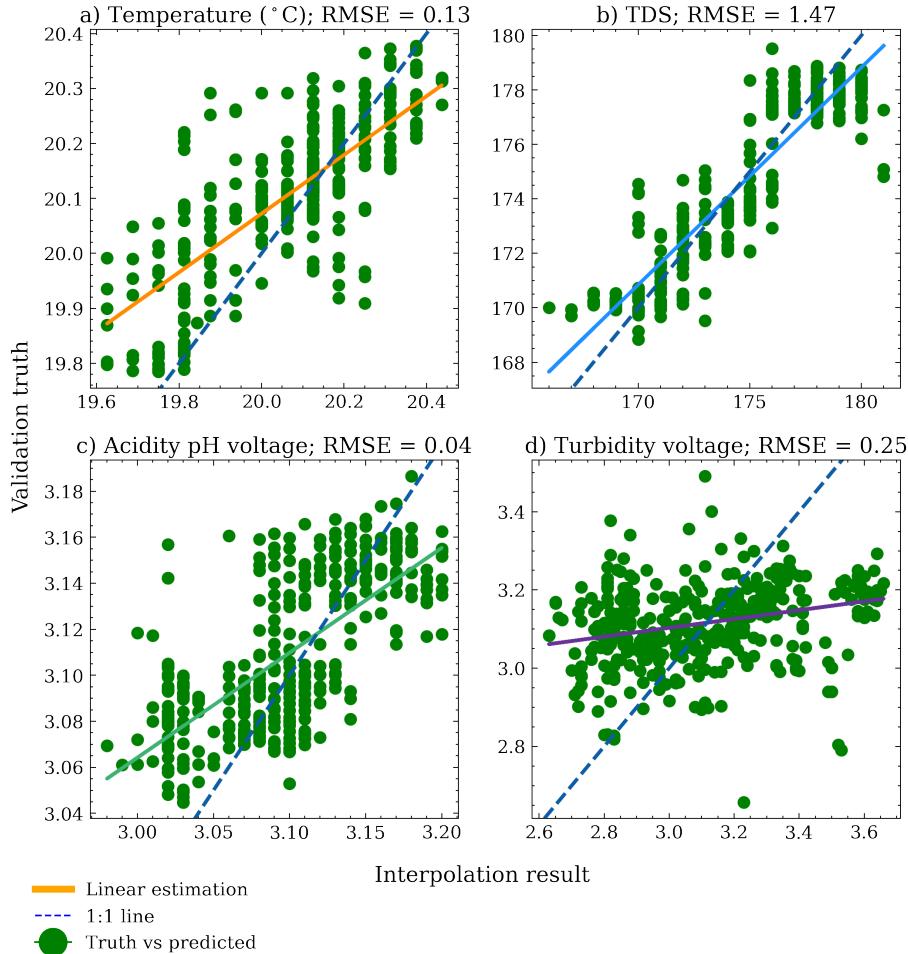


Figure 4.6: Accuracy scatter plot for the continuous method

The test dataset, red points in figure 4.2a, were used as truth observations for measuring the accuracy of the final interpolation. Finally, the RMSE here is calculated from the truth and observed values. The difference in ranges of validation truth observation values and ranges of interpolated values is that in the interpolation dataset, the minimum and maximum values were higher, thus slightly overfitting on the training set. This is compounded by the higher values of p for pH and temperature.

These final interpolations were also tested against the testing dataset from figure 4.2a. In figure 4.6 the scatter plot shows the interpolated values versus the actual measurements, the estimated linear fit of the points, and the RMSE. The scatter-plots show that when comparing the interpolations to the actual sampled values at the locations, the interpolation is not very comparable. Whilst the RMSE might be quite low, as well as the TDS and temperature linear estimations seem good at capturing the linear relation between the interpolation and ground truth. The 1:1 line, which shows the expected relation between truth and interpolation, the linear estimation and the 1:1 line do not share a similar trend. Across the board,

the 1:1 line is more steep than the linear estimation, which is partly explained by the reduction in minimum and maximum observations. Leading to less extreme, or steep, data.

4.3.2 Hold Location Sampling

Due to the slower update speed of some of the sensors, they are not as well suited to a moving platform, as can be seen in the tables in the methodology chapter, section 3.1. This means that a location hold sampling method might be more fitting in reliable measurements.

With the hold location sampling path, the data was filtered a few times to remove all the movement, and only kept the spots where the USV held its location. This removed around 99% of the input observations. Only 18 observations remained of the original data. The final observations can be seen in figure 4.2b.

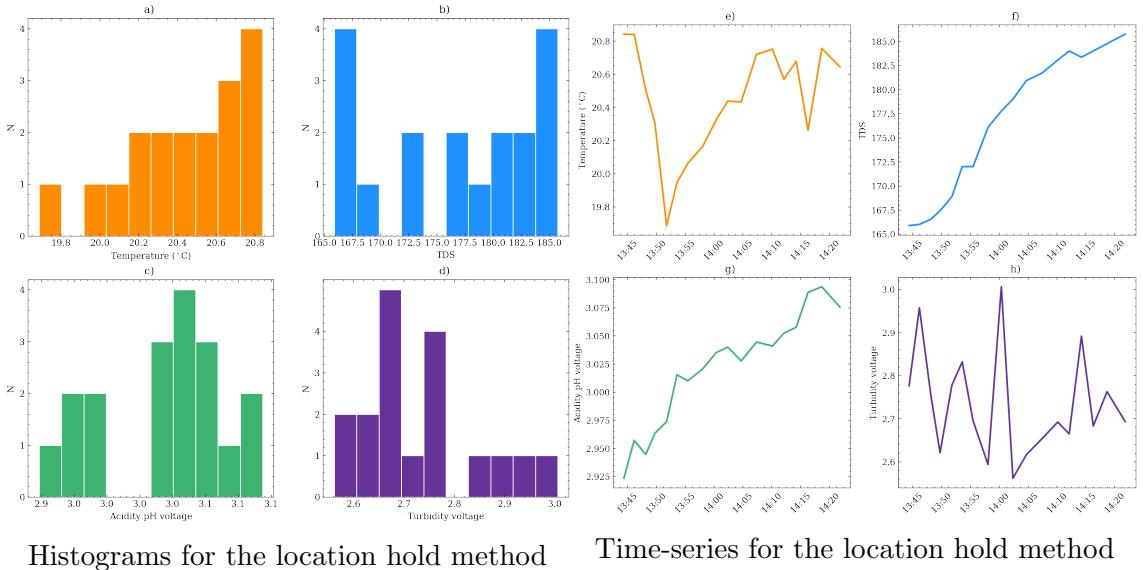


Figure 4.7: Contextual information location hold interpolation

The four figures on the left show the histograms for the 18 observations for each sensors. The lower number of observations also make it harder to observe whether these datasets are normally distributed. TDS seems to have two clear peaks however. The figures on the right show the time series for each of the sensors. TDS and pH are clearly drifting upwards in measurement.

The reduction of data is immediately visible in the histograms in figure 4.7. The size and number of bins is a lot less. Furthermore, the normal distributions are also less visible than in the continuous method, but have not completely disappeared. The two peaks of TDS are still visible (figure 4.7b). Furthermore, the TDS time series also has the slow increase in values that the continuous dataset has. Either showing a drift of values, or a difference between east and west TDS values in the pond itself (figure 4.7f and g). The time series for pH also shows an increase, while the turbidity seems random, and the temperature shows a large dip at the start (figure 4.7a).

Furthermore in the time-series, the difference in range of observations between the continuous sampling method (figures 4.3e, f,g and h) and the location hold method is quite large. Where the continuous sampling method has larger ranges for temperature (range of 1.4 vs 1 °C), pH (range of pH 0.4 vs 0.075pH voltage) and

turbidity (range of 1.2 vs 0.4 turbidity voltage), TDS show a similar range (17.5 vs 20 TDS). TDS can be explained by the fact that the fast update speed of the TDS sensor is less affected by the movement of water.

This means that the slower updating sensors, temperature, turbidity and pH are measuring less noise in the data, and have a benefit in the sampling and hold method.

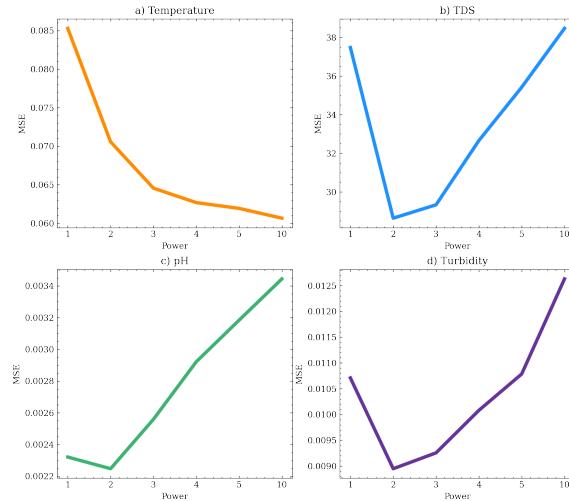


Figure 4.8: Cross validation results for the LOOCV MSE search for each sensor
The cross validation results show that the optimum values for p are lower than for the continuous sampling method, except for the temperature sensor.

The location hold training data was cross validated with a leave one out cross validation. The results are shown in figure 4.8. These figures show a different result to the cross validation from the continuous sampling. All the lines show an increase of error with an increase in p . Apart from the temperature observation, where an optimum p value of 10 is shown. Furthermore, in the time series (4.7f and g) it was shown that TDS and pH are drifting up over time. The turbidity has a confounding effect with the angle of the sensor to the sun, as seen in 4.5d.

The resulting interpolation for these optimum values of p is presented in figure 4.9. The interpolation for temperature has a large p value (10), which resulted in a clear distinction between the different sampling points, these show up as the patchwork-like visualization.

The TDS also has the east-west shift that the continuous interpolation showed. Acidity has a similar east-west difference. The hot-spot in the TDS and pH interpolation however is the last observation from the dataset, leading to the conclusion that the TDS values were drifting when being in water for a longer time. The temperature probe has had more time to settle on a measurement in this sampling technique, which is visible in a wide range of values. Interesting here is the difference between the eastern and western most points of the lake. At the eastern most point, a water outlet was present which was turned on right at the time of sampling. Finally turbidity seems to be going up and down randomly, but it is likely that the variance in this dataset can be explained by the sun illuminating the photo-voltaic sensor at the heart of the system.

Furthermore, this dataset has less observations than the continuous dataset has. Due to the lower number of observations, the results are more prone to overfitting

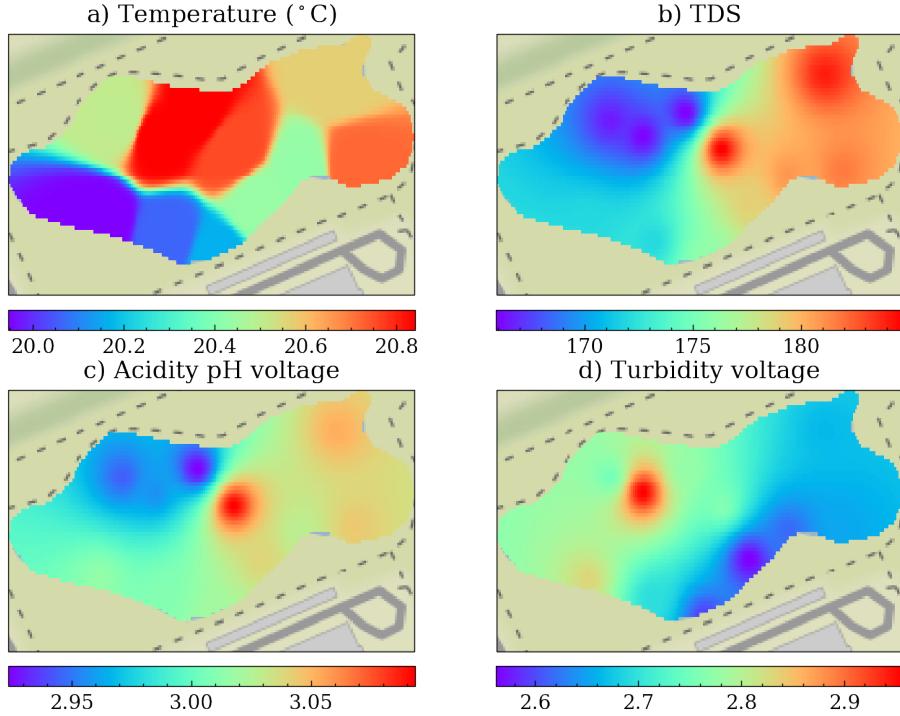


Figure 4.9: Interpolated final maps for the location hold method

In these figures the raster images resulting from the interpolation are shown. These rasters were masked with a polygon of the GAIA pond, and visualized with a basemap. The interpolations in b, c and d show the impact of the lower p values in the visible input points, whilst the higher p resulted in the patchwork like input-points.

on the test dataset. By finding the interpolated value for the truth values in the testing dataset, the accuracy scatter plot was created (figure 4.10).

The interpolation results of the location hold method show a similar accuracy to the continuous sampling. Comparing the ideal 1:1 line versus the linear estimation however shows that these estimations are off. Turbidity is worst off, due to the samples being essentially random. Finally, the RMSE of these values are very similar compared to continuous sampling, apart from TDS. In TDS, the higher number of input points, in combination with the east-west, time-based changes in the data, the interpolation performed better, as this temporal pattern is also visible in the testing data.

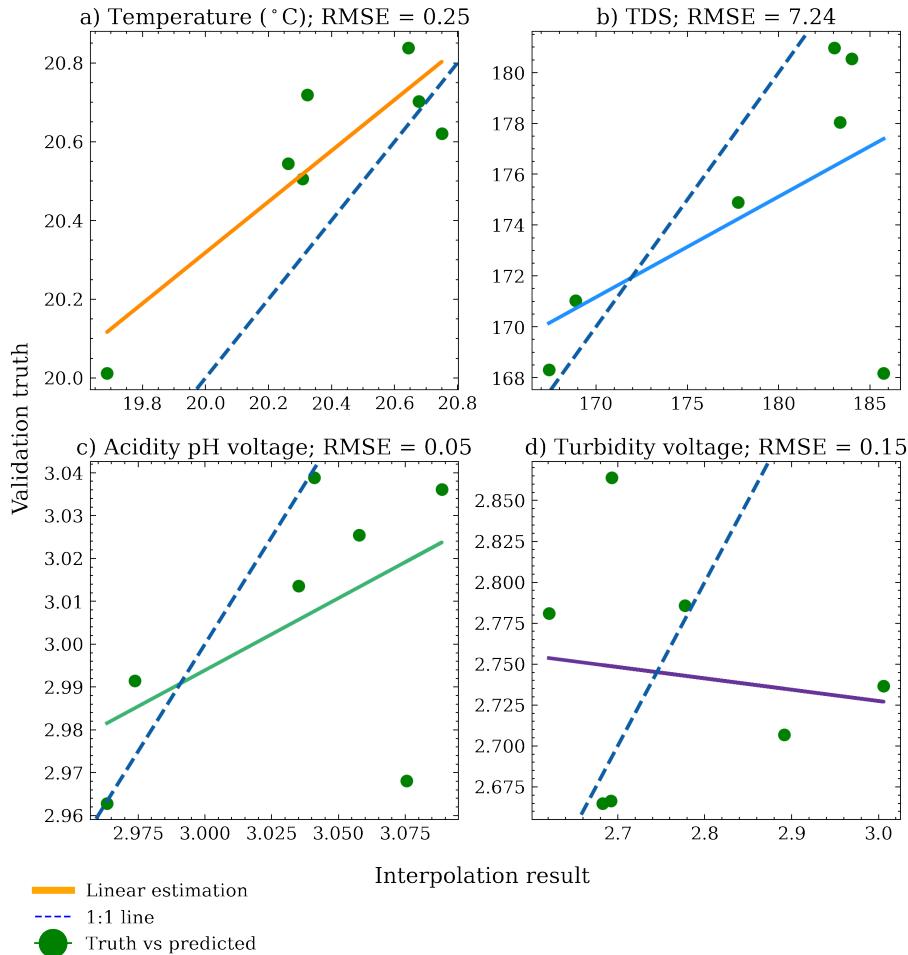


Figure 4.10: Accuracy scatter plot for the location hold method

Only four observations remained in the test dataset. This means that these RMSE values are calculated with a low observation count. The linear estimation of the temperature interpolation shows the highest linearity, which the other observations do not. Comparing the linear estimations to the ideal 1:1 line show that the interpolation did not result in accurate interpolation. Furthermore, the range of values has been reduced by the interpolation. Finally, the RMSE here is calculated from the truth and observed values.

Chapter 5

Discussion

This chapter starts by contextualizing the results with the use of literature. Afterwards a broader perspective is presented, in which the sensor system, other systems from literature and future systems can be categorized.

5.1 Sampling methods & interpolation techniques

Interpolation for water quality parameters has been an active field, from both geo-information science and water quality analysis.

The findings from this study are focused on the sampling techniques, comparing a continuous sampling technique to the location hold method. The main findings can be summarized as the following:

In continuous sampling, the larger number of observations are limiting the generalization of the results. The path the USV took on the water surface is very clearly seen in the interpolation. Furthermore, the sensors themselves also show problems when interpolated on a map. The case of turbidity is the clearest, where the angle of the sun directly impacts the observations. The temperature seems the most reliable, whilst the pH and TDS suffer from a east-west difference which could be explained by a reduction in input voltage over time. The temperature sensor is most reliable in interpolating.

The low number of observations in the location hold method might not be sufficient in drawing the conclusion that the location hold method is better. However, the benefits of having less noise in the data due to the increased time that the sensors have to acquire a reliable reading are a plus for this method of sampling.

Other water quality robotics systems also have executed interpolations, such as in Koparan et al. (2018) and Kaizu et al. (2011). In Koparan et al. (2018), these interpolations were exclusively used for the final visualization, and IDW was chosen as the interpolation algorithm. However, no accuracy was presented. Furthermore, the relation of path planning and interpolation was also not acknowledged. This thesis adds in this aspect the fact that the interpolation itself was assessed, which helped further understand the chosen water quality sensors. This makes the system a step closer to integrated assessment of the complete picture, where everything is seen as an interrelated system.

In Kaizu et al. (2011), the path planning and sampling has all been designed in order to execute an interpolation on it afterwards. The conducted experiment consisted of moving to a centre of a point. When arriving at the centre of the point, the

sensors were dropped in the water, and the sampling started. The interpolation was the kriging method from ArcGIS 9.3.1 (Esri, 2008). As an accuracy assessment, the coefficient of variation was used. From their result, the chlorophyll-a and turbidity sensors showed the highest coefficient of variation, which they attribute to the drifting of the USV itself on the water surface, impacting their readings. While the study conducted in this thesis presents that another option might be possible: the optical sensors can be directly affected by sunlight or drifting voltages. Furthermore, the temperature readings in Kaizu et al. (2011) were slowly going upwards, their tests occurred in the morning, and the solar radiation slowly heated up the water surface. They did not conclude that the sensors themselves were attributing to variations in the final maps. They stated that the variation was due to environmental aspects, such as sun and wind. This shows that by using higher quality sensors, the notion that the sensors themselves might be a factor in the interpolation is negated.

In USV water quality parameter interpolation, the laid out pattern and interpolation technique need to be determined before doing the experiment. This was an oversight in this thesis. The best would be to use equally spaced sampling points, such as in Kaizu et al. (2011). Different spatial patterns, and the density of these patterns yield different results, even though the same interpolation technique is applied (Li & Heap, 2008b). Furthermore, the sensor data needs to be of high quality with little noise (Li & Heap, 2008b). This study reinforces this perspective. In the location hold method, the noise of the data was less, at the cost of data points. However, a limitation to the location hold method was not defining a sampling grid beforehand. Not only will this improve the interpolation accuracy, such patterns and grids will inform the path planning when selecting the optimum route.

5.2 Broader perspective

The trade-off when using robotics in environmental analysis as explored in this study can be seen from the perspective presented in figure 5.1. It presents three important aspects to water quality USV development, or perhaps environmental robotics in general. It presents that there is a trade-off between costs, accuracy and real time capabilities.

Starting with the results from this system, the system presented in this study is best described by the right hand side, affordable robotics. It was developed with the main goals being ROS integration, and being low-cost. This has caused limitations in the accuracy aspect. This has become clear from the calibration results (eg. figure 4.1b), or the interpolation results from the turbidity sensor (figure 4.5d).

However, benefits exist in this system, such as the real time dashboard that was used during the tests and calibration. Furthermore, there are possibilities to implement the sensors into a odometry-based navigation algorithm, albeit in a proof-of-concept type system.

An important finding in this research, as visualized in the triangle, is that trying to implement all three aspects into a single system is not yet possible. However, there are signs of it becoming possible, such as the used temperature sensor, which showed comparable accuracy to the temperature meter, whilst simultaneously being low-cost.

In literature similar systems can also be categorized according to this trade-off triangle. The USV in Kaizu et al. (2011) consists of high-quality water sensors,

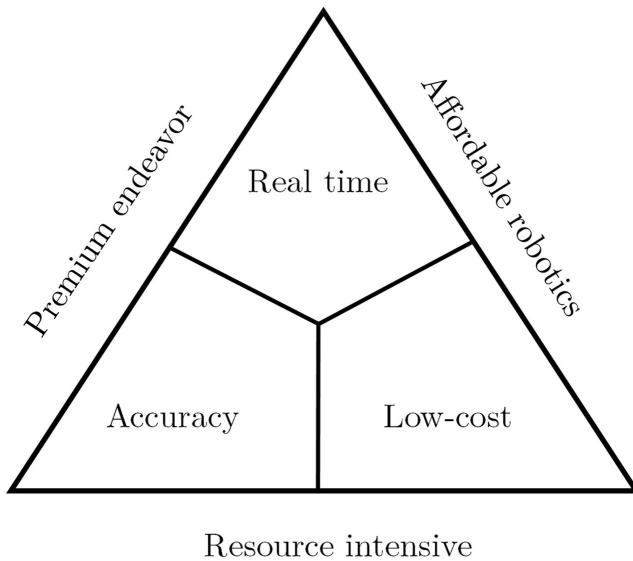


Figure 5.1: The trade-off for water quality robotics mental map

There are three aspects to this image, but only one side can be chosen for any environmental system. The upper section is the real time capabilities of the system. This includes live dashboards, navigation inputs from the sensors. The right section is about the cost of the system, where the low cost indicates the price of the sensors, but also the computational resources and knowledge of the system required in order to use it. The final section is the bottom left, accuracy, the accuracy of the system has to do with the quality of the sensors themselves, the effort required for calibration and therefore the usefulness of the samples in academic research. The sides are named in such a way to provoke an image to what the system would entail.

and robotics integration to optimize the path following capabilities of the USV. The identified limitation of the system is the slow pace it has in order to map out a complete lake. In 2011, such an autonomous vehicle was not yet applied to water quality assessment, making it a leading development. Furthermore, the sensors themselves were not identified as being a limiting factor. This fits within the premium endeavor categorization, as cost was not one of the main points of the system.

The manually controlled USV in Casper et al. (2012), shows the potential of in-depth water quality analysis enabled by a USV. This system was controlled manually, with no feedback to the user, only being line of sight. Their mapping of a riparian environment enabled analysis at different scales of ecosystem dynamics, such as local differences in plants, all the way up to the effect of the sea on the water quality at large. The lack of real time features, high quality sensors and manual data transfer and processing, make it a good fit for the resource intensive side of the triangle.

The RIVERWATCH marsupial team, presented in Pinto et al. (2014), shows the capabilities of multiple autonomous systems interacting. Even though the sensors consisted of camera's for perception, and a sonar for mapping the bottom surface of lakes and rivers. The cost was not a limiting factor for this system. The presented findings on robotic interaction pushed the envelope of environmental robotics as a whole with this project. This system is another example of a premium endeavor. Where cost is less of a limiting factor, and the accuracy of the findings, reliability of the system, and real time perception were the main goals to focus on.

The UAV in Koparan et al. (2018) shows that autonomous water quality analysis is not limited to USV platforms. It presents an open-source, more affordable, water

quality sensors and an assessment of sensor accuracy and interpolation. A large limiting factor was the flight time of around 6 minutes, making it impossible to map a larger water surface. Furthermore, a lack of real time mapping capabilities shows that this system fits in the resource intensive part of the triangle.

The final water quality USV in Jo et al. (2019) shows that water quality mapping is not limited to resource intensive or expensive systems. The low-cost sensors, combined with a Bluetooth connection and app, made real time water quality inquiry possible. However, no accuracy assessment was executed. This is a candidate for the affordable robotics side, where the real time capabilities of a robot and low cost are the main focus of the system.

In summary, the triangle can inform where different environmental robotics systems lie in this spectrum, but it is not rigid. Systems can change with more development, making it more real time or having higher accuracy. This makes the triangle not a definitive structure, but a tool for guiding design decisions and clarify which trade-offs present itself in the development process.

Figure 5.1 also can give an informed decision in what is ahead in the field of water quality robots, from the three presented sides of the triangle.

Starting with affordable robotics, sensor development is very rapid. The niche sensors, such as pH, are slightly slower than the broadly implemented temperature sensors. This means that systems developed in the future will slowly improve in accuracy, and over time be as accurate and reliable as the expensive systems today. This is currently visible in the most accurate sensor in this system, the temperature sensor. Furthermore, these systems will lead the way to inform the development and testing for the more valuable systems, as failures will be less costly on these platforms.

On the bottom side, the low-cost is still present, but accuracy is maintained, resulting in a reduction in real time mapping and odometry options. This leads to the resource intensive systems. This then is the future of systems taking samples, but not analysing those samples on the fly. Further development would be water-sampling mechatronics at different depths for example.

Finally the left hand side of figure 5.1, valuable enterprise, systems that are less-cost prohibited than the others. These systems have high-quality sensors on-board in order to take reliable samples that can both inform ecologists, and the control algorithms on the USV system itself. These systems will have the ability to identify point sources of pollution on the fly, reliably. Furthermore, these systems will be the leading driver in understanding water quality dynamics, with the availability of reliable, repeatable, high resolution measurements in both 3D-space and time.

Chapter 6

Conclusion

In this thesis, the development of a water quality sensing platform was presented. This system was created in order to simplify water quality measurements, and understand the sensors for implementation into a wider USV control system. The methodology presented the sensor technology, and explain the different subsystems that make up the sensor platform. In the results, the accuracy of the sensors was presented, and a test run of mapping a small pond was executed. In the discussion, these results were compared to other USV interpolation studies, and the trade-off in water quality robotics was presented.

This leads to answering the main research question concerning the contribution of a low-cost USV for water quality measuring. To answer this main research question, the subquestions concerning the implementation decisions, mapping the indicators and final processed data need to be answered first. This chapter presents the answers to these questions, and concludes by giving recommendations for similar endeavors into this field of study.

The project started from a low-cost, multi-variable mapping perspective. The goal was to understand how to implement a system capable of mapping water quality parameters, and how good such a system is in measuring the variables. A set of low-cost water quality probes were chosen.

The temperature sensor. The DS18B20 is a widespread, simple to implement and accurate temperature sensor. Intelligent calibration, voltage referencing is all built-in. Temperature has a widespread use for a variety of industries and is further along in price and size, compared to other environmental sensors. This makes the waterproof temperature probe a good probe for this system. Furthermore, the calibration showed good linearity to the reference meter, while the interpolation showed that nothing stood out, which is expected of temperature variety on a water surface.

The other probes were less successful, the TDS sensor drifted when measuring for a longer period. This could have been caused by a variety of things, from an internal voltage reference not being stable, or one of the microchips on the DFRobot support-board being sensitive to temperature changes in the electronics box. The pH sensor suffered from the same issue, but was also highly sensitive to movement in the water. This made the measurements unreliable for in-situ measuring. Finally, the turbidity sensor was very sensitive to outside light, which was clearly seen in the continuous interpolation, where the angle of the USV relative to the sun impacted the measurement. The setting for these low-cost sensors is better fit in a laboratory setting. In a laboratory, light, the movement of water and continuous calibration

can be controlled.

In-situ measurement then has proven not quite possible with these low-cost sensors, due to drifting and environmental outside effects. This showed that for in-situ measurement, more basic, widely adopted low cost sensors are more usable, such as the temperature sensor, or higher quality sensors such as used in Casper et al. (2012),

Making a USV do the sampling, implementing the sensors in ROS, creating a monitoring dashboard, and having the data available for post-processing and interpolation contributes to the information on a small water body. However, this does not lead to the conclusion on contributing to academic understanding of this small water body. This is due to the limitations in the sensors, as explained above.

As seen in Casper et al. (2012), Jo et al. (2019), Kaizu et al. (2011), and Koparan et al. (2018), similar sensor systems exist. This thesis added to it by understanding of the sensor technology and accuracy and its relation to interpolation, by presenting the complete picture. Implementing a water quality sensing system, from software to hardware is easier to do than understanding the underlying technology and accuracy of such a system. This leads to answering the final research question, how a low-cost USV could contribute to mapping small water bodies. The presented trade-offs in figure 5.1 show that low-cost, real time possibilities and accuracy of the system are a trade-off. In order for USVs to contribute to understanding small water bodies, this trade-off needs to be kept in mind. Scientific inquiry requires high quality observations, meaning that a low-cost USV must give in on the real time capabilities, or requires more software and hardware work on the low-cost sensors. The options here are then sampling methods in which the robot itself does not sense, but takes a sample of the water, for analysis in laboratory conditions. Furthermore, the trade-off will be able to guide future designs of water quality USVs, by defining the most important features of the system, and which trade-offs to take. This means that the design decisions are reduced and make for an easier development cycle.

6.1 Recommendations

The continuing development in USV water body mapping is an exciting field; the findings from this study have resulted in the following recommendations for similar systems.

Spatial sensing could be useful for water bodies, but it must measure something that changes with space, or has an inherent location. In an isolated pond, the spatial variance is low, and thus the added value of a map is low. When the spatial aspect is important such as the varying coastal environment, depth, algal bloom sites, plastic waste or point-source pollution a spatial perspective is an added value.

Furthermore, vertical measurement in the water column has more value for water quality analysis than spatial measurements have. Larger differences can be present, when the water body is stratified. This also ties into the sampling technique, due to sensor needing to stabilize into a reading, and resulting in better interpolation results, a location hold methodology is better than continuous for water quality sampling.

The method of in-situ sampling can also be changed to a method in which the water itself is captured by the USV, such as in Koparan et al. (2018). The reliability of the measurement can be ensured in a laboratory setting.

When developing other USV systems, or improving upon those that are present, it is recommended to keep the trade-off triangle in mind. In order to be clear what the goal should be: scientific inquiry, proof of concepts, marsupial teams, point-source identification or anything else. The triangle can help identify where the system lies in this spectrum, and which features ought to be negated, and which should be optimized.

Bibliography

- Abraham, D. A. (2019). *Underwater Acoustic Signal Processing*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-92983-5>
- Arduino Nano. (n.d.). <https://store.arduino.cc/arduino-nano>
- Autodesk. (2021). Autodesk Fusion 360. <https://www.autodesk.com/products/fusion-360/overview?term=1-YEAR>
- Borreguero, D., Velasco, O., & Valente, J. (2018). Experimental design of a mobile landing platform to Assist Aerial Surveys in fluvial environments. *Applied Sciences (Switzerland)*, 9(1), 38. <https://doi.org/10.3390/app9010038>
- Boyd, C. E. (2000). Overview of Water Pollution. *Water quality: An introduction* (Second Edition, pp. 315–317). Springer International Publishing.
- Carmichael, W. W. (1994). The Toxins of Cyanobacteria. *Scientific American*, 270(1), 78–84. <https://www.jstor.org/stable/24942554>
- Carpenter, S. R., Fisher, S. G., Grimm, N. B., & Kitchell, J. F. (1992). Global Change And Freshwater Ecosystems. *Annual Review Ecological Systems*, 23. www.annualreviews.org
- Carstensen, J., Conley, D. J., Bonsdorff, E., Gustafsson, B. G., Hietanen, S., Janas, U., Jilbert, T., Maximov, A., Norkko, A., Norkko, J., Reed, D. C., Slomp, C. P., Timmermann, K., & Voss, M. (2014). Hypoxia in the Baltic Sea: Biogeochemical cycles, benthic fauna, and management. *Ambio*, 43(1), 26–36. <https://doi.org/10.1007/s13280-013-0474-7>
- Casper, A. F., Dixon, B., Steimle, E. T., Hall, M. L., & Conmy, R. N. (2012). Scales of heterogeneity of water quality in rivers: Insights from high resolution maps based on integrated geospatial, sensor and ROV technologies. *Applied Geography*, 32(2), 455–464. <https://doi.org/10.1016/j.apgeog.2011.01.023>
- Castell, N., Dauge, F. R., Schneider, P., Vogt, M., Lerner, U., Fishbain, B., Broday, D., & Bartonova, A. (2017). Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates? *Environment International*, 99, 293–302. <https://doi.org/10.1016/j.envint.2016.12.007>
- common_interfaces. (2021). https://github.com/ros2/common_interfaces
- Downing, J. A., Cole, J. J., Middelburg, J. J., Striegl, R. G., Duarte, C. M., Kortelainen, P., Prairie, Y. T., & Laube, K. A. (2008). Sediment organic carbon burial in agriculturally eutrophic impoundments over the last century. *Global Biogeochemical Cycles*, 22(1). <https://doi.org/10.1029/2006GB002854>
- Downing, J. A. (2010). Emerging global role of small lakes and ponds: little things mean a lot. *Limnetica*, 29(1), 9–24.
- Dunbabin, M., & Marques, L. (2012). Robots for environmental monitoring: Significant advancements and applications. *IEEE Robotics and Automation Magazine*, 19(1), 24–39. <https://doi.org/10.1109/MRA.2011.2181683>
- Esri. (2008). ArcGIS Desktop: Release 9.

- European Commission. (2000). Directive 2000/60/EC of the European Parliament and of the Council establishing a framework for Community action in the field of water policy. <https://www.eea.europa.eu/policy-documents/directive-2000-60-ec-of>
- Forbes, M. G., Back, J., & Doyle, R. D. (2012). Nutrient transformation and retention by coastal prairie wetlands, upper gulf coast, Texas. *Wetlands*, 32(4), 705–715. <https://doi.org/10.1007/s13157-012-0302-z>
- geosardine. (2021). <https://github.com/sahitono/geosardine>
- Graham, D. W., Collignon, P., Davies, J., Larsson, D. G., & Snape, J. (2014). Underappreciated role of regionally poor water quality on globally increasing antibiotic resistance. <https://doi.org/10.1021/es504206x>
- Gravity: Analog pH Sensor / Meter Kit For Arduino. (n.d.). <https://www.dfrobot.com/product-1025.html>
- Gravity: Analog TDS Sensor/Meter for Arduino. (n.d.). <https://www.dfrobot.com/product-1662.html>
- Gravity: Analog Turbidity Sensor For Arduino. (n.d.). <https://www.dfrobot.com/product-1394.html>
- Gravity: Waterproof DS18B20 Sensor Kit. (n.d.). <https://www.dfrobot.com/product-1354.html>
- Hakim, W. L., Hasanah, L., Mulyanti, B., & Aminudin, A. (2019). Characterization of turbidity water sensor SEN0189 on the changes of total suspended solids in the water. *Journal of Physics: Conference Series*, 1280(2). <https://doi.org/10.1088/1742-6596/1280/2/022064>
- Hipp, R. D. (2020). SQLite. <https://www.sqlite.org/index.html>
- Hossain, S. (2019). Visualization of Bioinformatics Data with Dash Bio. <https://doi.org/10.25080/Majora-7ddc1dd1-012>
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017). 5: Resampling Methods. In G. Casella, Fienberg. S/, & I. Olkin (Eds.), *An introduction to statistical learning with applications in r* (8th ed., pp. 175–184). Springer Science + Business Media. <https://doi.org/10.1007/978-1-4614-7138-7>
- Jo, W., Hoashi, Y., Paredes Aguilar, L. L., Postigo-Malaga, M., Garcia-Bravo, J. M., & Min, B. C. (2019). A low-cost and small USV platform for water quality monitoring. *HardwareX*, 6, e00076. <https://doi.org/10.1016/j.hwx.2019.e00076>
- Jones, J. B., & Mulholland, P. J. (1999). *Streams and ground waters*. Elsevier.
- Jordahl, K., Van den Bossche, J., Fleishmann, M., McBride, J., Wasserman, J., Gerard, J., Garcia Bardacco, A., Snow, A. D., Tratner, J., Perry, M., Farmer, C., Hjelle, G. A., Cochran, M., Gillies, S., Culbertson, L., Bartos, M., Caria, G., Eubank, N., sangarshanan, ... abonte. (2020). geopandas/geopandas: v.0.9.0. <https://doi.org/10.5281/zenodo.3946761>
- jurriandoornbos/wqdrone. (2021). <https://github.com/jurriandoornbos/wqdrone>
- Kaizu, Y., Iio, M., Yamada, H., & Noguchi, N. (2011). Development of unmanned airboat for water-quality mapping. *Biosystems Engineering*, 109(4), 338–347. <https://doi.org/10.1016/j.biosystemseng.2011.04.013>
- Keeler, B. L., Polasky, S., Brauman, K. A., Johnson, K. A., Finlay, J. C., O’Neill, A., Kovacs, K., & Dalzell, B. (2012). Linking water quality and well-being for improved assessment and valuation of ecosystem services. *Proceedings of*

- the National Academy of Sciences of the United States of America*, 109(45), 18619–18624. <https://doi.org/10.1073/pnas.1215991109>
- Khairuddin, A. R., Talib, M. S., & Haron, H. (2016). Review on simultaneous localization and mapping (SLAM). *Proceedings - 5th IEEE International Conference on Control System, Computing and Engineering, ICCSCE 2015*, 85–90. <https://doi.org/10.1109/ICCSCE.2015.7482163>
- Kogger. (2020). Open Serial Binary Protocol specification. <https://github.com/koggertech/Kogger-Protocol>
- Koparan, C., Koc, A. B., Privette, C. V., & Sawyer, C. B. (2018). In Situ Water Quality Measurements Using an Unmanned Aerial Vehicle (UAV) System. *Water*, 10(3), 264. <https://doi.org/10.3390/w10030264>
- Leigh, C., Burford, M. A., Roberts, D. T., & Udy, J. W. (2010). Predicting the vulnerability of reservoirs to poor water quality and cyanobacterial blooms. *Water Research*, 44(15), 4487–4496. <https://doi.org/10.1016/j.watres.2010.06.016>
- Li, D., & Liu, S. (2019). Sensors in Water Quality Monitoring. *Water quality monitoring and management* (pp. 1–54). Elsevier. <https://doi.org/10.1016/b978-0-12-811330-1.00001-6>
- Li, J., & Heap, A. D. (2008a). *A Review of Spatial Interpolation Methods for Environmental Scientists*. Geoscience Australia.
- Li, J., & Heap, A. D. (2008b). Chapter 2: Interpolation methods. *A review of spatial interpolation methods for environmental scientists* (pp. 4–26). Geoscience Australia.
- Li, J., & Heap, A. D. (2008c). Chapter 4: Assessment Measures. *A review of spatial interpolation methods for environmental scientists* (pp. 4–26). Geoscience Australia.
- Liechti, C. (2021). pySerial. <https://github.com/pyserial/pyserial>
- Makinwa, K. A. (2010). Smart temperature sensors in standard CMOS. *Procedia Engineering*, 5, 930–939. <https://doi.org/10.1016/j.proeng.2010.09.262>
- Martinez, K., Hart, J. K., & Ong, R. (2004). Environmental sensor networks. *Computer*, 37(8), 50–56. <https://doi.org/10.1109/MC.2004.91>
- Marton, J. M., Creed, I. F., Lewis, D. B., Lane, C. R., Basu, N. B., Cohen, M. J., & Craft, C. B. (2015). Geographically isolated wetlands are important biogeochemical reactors on the landscape. <https://doi.org/10.1093/biosci/biv009>
- Maxim Integrated. (2019). DS18B20 Programmable Resolution 1-Wire Digital Thermometer. <https://www.maximintegrated.com/en/products/sensors/DS18B20.html>
- Mullins, M. L., & Doyle, R. D. (2019). Big things come in small packages: Why limnologists should care about small ponds. *Acta Limnologica Brasiliensis*, 31. <https://doi.org/10.1590/s2179-975x4119>
- Murphy, R. R., Curriero, F. C., & Ball, W. P. (2010). Comparison of Spatial Interpolation Methods for Water Quality Evaluation in the Chesapeake Bay. *Journal of Environmental Engineering*, 136(2), 160–171. [https://doi.org/10.1061/\(ASCE\)EE.1943-7870.0000121](https://doi.org/10.1061/(ASCE)EE.1943-7870.0000121)
- NEO-M8. (2021). https://www.u-blox.com/sites/default/files/NEO-M8-FW3-Datasheet_UBX-15031086.pdf
- NeoGPS. (2018). <https://github.com/SlashDevin/NeoGPS>

- Oki, T., & Kanae, S. (2006). Global hydrological cycles and world water resources. <https://doi.org/10.1126/science.1128845>
- Open Robotics. (2021). ROS 2 Documentation. <https://docs.ros.org/en/foxy/>
- Pinto, E., Marques, F., Mendonca, R., Lourenco, A., Santana, P., & Barata, J. (2014). An autonomous surface-aerial marsupial robotic team for riverine environmental monitoring: Benefiting from coordinated aerial, underwater, and surface level perception. *2014 IEEE International Conference on Robotics and Biomimetics, IEEE ROBIO 2014*, 443–450. <https://doi.org/10.1109/ROBIO.2014.7090371>
- Raspberry Pi Foundation. (n.d.). Raspberry Pi 4 Model B. <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>
- Romero-Vivas, E., Von Borstel, F. D., Pérez-Estrada, C. J., Torres-Ariño, D., Villa-Medina, J. F., & Gutiérrez, J. (2015). On-water remote monitoring robotic system for estimating the patch coverage of *Anabaena* sp. filaments in shallow water. *Environmental Sciences: Processes and Impacts*, 17(6), 1141–1149. <https://doi.org/10.1039/c5em00097a>
- Scheffer, M. (1997). 1: The story of some shallow lakes. *Ecology of shallow lakes* (1st, pp. 1–4). Kluwer Academic Publishers.
- Scheffer, M. (2004a). Introduction. *Ecology of shallow lakes* (pp. xiv–xix). Springer Netherlands. <https://doi.org/10.1007/978-1-4020-3154-0>
- Scheffer, M. (2004b). Phytoplankton. *Ecology of shallow lakes* (pp. 76–118). Springer Netherlands. <https://doi.org/10.1007/978-1-4020-3154-0>
- Scheffer, M. (2004c). The abiotic environment. *Ecology of shallow lakes* (pp. 20–70). Springer Netherlands. <https://doi.org/10.1007/978-1-4020-3154-0>
- Semtech LoRa SX1276 UART Interface Module. (n.d.). <http://www.dorji.com/products-detail.php?ProId=61>
- Shepard, D. (1968). A two-dimensional interpolation function for irregularly-spaced data. *Proceedings of the 1968 23rd ACM National Conference, ACM 1968*, 517–524. <https://doi.org/10.1145/800186.810616>
- Shiklomanov, I. A. (2000). Appraisal and Assessment of world water resources. *Water International*, 25(1), 11–32. <https://doi.org/10.1080/02508060008686794>
- Siciliano, B., & Khatib, O. (2008). Introduction. *Springer handbook of robotics* (pp. 1–4). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-30301-5__1
- Sitch, S., & Drake, F. (2014). The changing water cycle. In J. Holden (Ed.), *Water resources* (pp. 19–48). Routledge New York.
- SONAR 2D-ENHANCED. (n.d.). <https://kogger.tech/product/sonar-2d-enhanced/>
- Teensy ® 3.2 Development Board. (n.d.). <https://www.pjrc.com/store/teensy32.html#specs>
- The pandas development team. (2020). pandas-dev/pandas: Pandas. <https://doi.org/10.5281/zenodo.3509134>
- Thermo Scientific Eutech TN-100 Waterproof Turbidimeter. (n.d.). <http://www.eutechinst.com/pdt-para-turbidity-tse-tn100.html>
- Tuna, G., Arkoc, O., & Gulez, K. (2013). Continuous monitoring of water quality using portable and low-cost approaches. *International Journal of Distributed Sensor Networks*, 2013. <https://doi.org/10.1155/2013/249598>
- United Nations General Assembly. (2015). *Transforming our world: the 2030 Agenda for Sustainable Development* (tech. rep.). United Nations General Assembly.

- Verpoorter, C., Kutser, T., Seekell, D. A., & Tranvik, L. J. (2014). A global inventory of lakes based on high-resolution satellite imagery. *Geophysical Research Letters*, 41(18), 6396–6402. <https://doi.org/10.1002/2014GL060641>
- Wang, M., Xu, X., Wu, Z., Zhang, X., Sun, P., Wen, Y., Wang, Z., Lu, X., Zhang, W., Wang, X., & Tong, Y. (2019). Seasonal Pattern of Nutrient Limitation in a Eutrophic Lake and Quantitative Analysis of the Impacts from Internal Nutrient Cycling. *Environmental Science and Technology*, 53(23), 13675–13686. <https://doi.org/10.1021/acs.est.9b04266>
- Wang, Y., Rajib, S. M. M., Collins, C., & Grieve, B. (2018). Low-Cost Turbidity Sensor for Low-Power Wireless Monitoring of Fresh-Water Courses. *IEEE Sensors Journal*, 18(11), 4689–4696. <https://doi.org/10.1109/JSEN.2018.2826778>
- Ware, J. R., Smith, S. V., & Reaka-Kudla, M. L. (1991). *Coral reefs: sources or sinks of atmospheric CO₂* (tech. rep.).
- WTW. (2004). Hanheld meter Cond 315i Operating manual. https://www.labunlimited.co.uk/pdf/wtw_lab_man_cond_315i.pdf
- WTW. (2008). pH 3210 Operating manual. <https://usermanual.wiki/Pdf/WTWpH3210.1732600204.pdf>
- Xiaomi Mi WiFi Router Mini. (n.d.). <https://xiaomi-mi.com/wifi-routers/xiaomi-mi-wifi-router-mini-white/>