Vision Impaired Swim Aid

Design Document

Team 5

Advisor: Lealand Harker

Carson Kneip, Conor Albinger, Jake Sieverding, Nathan Mortenson, Paden Uphold, and Timothy Steward

sdmay20-05@iastate.edu

http://sdmay20-05.sd.ece.iastate.edu/

Revised: April 25, 2020

Executive Summary

Development Standards & Practices Used

- For prototyping
  - Arduino (Programmed in C)
  - Raspberry Pi
  - Image Detection
    -

  - IR sensors
    - Sharp GP2D12

  - Ultrasonic/Sonar
    - Vexilar Sonar
    - Parallax PING)))™

    - MaxBotix MB7072-200

    - BlueRobotics Ping Sonar Altimeter and Echosounder

Summary of Requirements

- All components waterproof
- Device should be completely operable by blind and vision impaired users
- Device should detect the user when they are near the end of the
- Device should warn the user when they are near the end of the pool

Applicable Courses from Iowa State University Curriculum

- CPR E 288
- EE 224/324
- EE 201/230
- EE 321
- EE/CPR E 185
- COMS 327

New Skills/Knowledge acquired that was not taught in courses

- Using sensors in the water
- Image detection
- Soldering
- Researching commercially available parts/devices

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Acknowledgement

Our advisor, Lee Harker, is the overall manager of the project and all the products being purchased have to be confirmed and funded by the ETG.

The ISU swim coach, Micheal Peterson, has helped by providing information about how competitive swimmers compete and what accommodations that are made to a visually impaired swimmer. We also reached out to Brandon Schellhorn, the swim coach for Heartland AEA, and he gave us good information about how the swimmer would be able to set up the device, who it would be helpful for, and why this would be a useful device.

Contact Information

Micheal Peterson, Coach for ISU swim team, Iowa State University, coachmike@swimacac.com

Brandon Schellhorn, Coach for the Visually Impaired, Iowa Braille School/IESBV - Heartland AEA, 515-249-6312

1.2 Problem and Project Statement

Problem

Swimmers who are visually impaired cannot see the lines at the bottom of the pool to indicate to a lap swimmer when they are close to the wall. Because of this, those who cannot see the lines are forced to have assistance from another person who taps them on the head with a stick when it is time to turn around. This forces vision impaired people to have to work around an additional person's schedule to find time to swim.

Project Statement

To allow visually impaired people to be able to swim without other's assistance, we set out to create a device that would allow the swimmer to independently swim laps on their own time. This device will tell the swimmer when they are getting close to the end of the pool, so they know to turn around. We planned to create a device that will be easy to set up and takedown, and be accommodating for vision impaired people.

Goals

- Deliver a complete final product that allows vision impaired swimmers to swim laps independent of another person
- Improve our understanding of the complete engineering design process
- Gain further technical insight beyond what we have learned in our previous coursework

1.3 Operational Environment

- The headphones and radio receiver on the swimmer will be used for extended periods of time in chlorinated pool water. They will need to be completely sealed and waterproof. They will need to be able to withstand the effects of any chemicals in pool water. They will need to be powered by a rechargeable battery that can power them for the duration of a swim.

- The control boxes and sensors will be placed at the ends of the pool. They may be used in both indoor and outdoor pools. They need to withstand splashes from the pool, the humid climate of an indoor pool, sunlight/UV rays, and light-moderate winds.
- The environment for this project will be outside in a pool or inside for lap swim competitions

1.4 Requirements

- All components waterproof
- Device should be completely operable by blind and vision impaired users
- Device should detect the user when they are near the end of the lane
- Device should warn the user when they are near the end of the pool
- Device should detect the user while they are performing any of the four main competitive swimming strokes
- All instructions and indications should be helpful and accommodating for vision impaired users

1.5 Intended Users and Uses

The intended user would be vision impaired lap swimmers. We are designing this product to help vision impaired lap swimmers know when they are getting close to the edge of the pool so they could turn around before hitting their head while at the same time not having two other people tap their shoulders when they get close.

1.6 Assumptions and Limitations

Assumptions

- The product will primarily be for vision impaired lap swimmers
- The vision imparied will have someone available to help them set it up
- The product will have two sensors on each side of the pool
- Product will be used in a freshwater lap swimming pool
- The pool will be no longer than 50 meters

Limits

- Vision impaired and blind people will need to be able to use it
- The system needs a sensor on both ends to detect swimmer, otherwise pool length is too long
- The system needs to be wireless and operate in wet conditions
- The system needs to be battery powered

1.7 Expected End Product and Deliverables

The end products are going to be:

- Control Boxes with sensors attached
    - These boxes will be mounted on each end of the swimming lane. Each box will have an attached sensor to detect when the user is nearing the end of the lane, as well as an FM transmitter to communicate and audio signal to the user.

- Sensor Selection
  - For this project we wanted to find a sonar or IR sensor that can reliably detect the user as they are swimming towards the end of the lane. We then found out that computer vision worked best for detecting the swimmer so we decided on that. The sensor should work for all the main competition swimming strokes.
- Wireless headphones/earbuds
  - This device will allow the user to receive the audio signal from the control boxes when it is necessary for the user to turn around.

2. Specifications and Analysis

2.1 Proposed Design

The first thing we did in our design process was to come up with a high-level overview of the whole system. The overview is shown below in Figure 1.

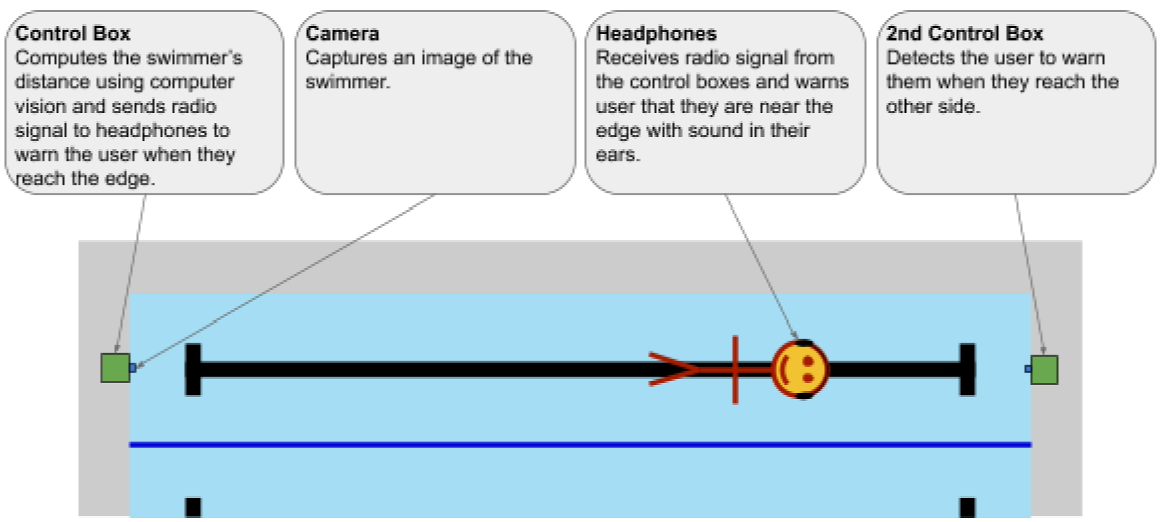

Figure 1: Overview of the system

2.2 Design Analysis

- We started off by going to the pool to test the different sensors
- Different types of sensors like IR and Sonar are used to test how they react to the swimmer on the surface of the water
- The first sensors we used from the ETG
- We later found that computer vision worked better then the IR and Sonar sensors
- The microcontroller that is being used is the Arduino Uno
- After we get a good tracking of the swimmer, then we can focus on the communication from the control box to the swimmer

2.3 Development Process

It is hard to imagine how disabled people go about their everyday needs. We met with the Student Accessibility Services office here at Iowa State to reach out to members in our community that are disabled. We want to determine some norms that vision impaired swimmers might follow. Our goal is to make our product as user friendly as possible. We also reached out to the coach from the Iowa Braille School and he said that most vision impaired have someone there to help set up the equipment. He also mentioned that about 10-20% of his students get tapped on the shoulder, and a lot of his swimmers would benefit from it when they do the backstroke. After we establish some more norms that need to be followed, then we just need to get the system to work. Right now our biggest issue is tracking the swimmer. We have obtained good data from our IR sensor that would work kind of like a break beam to detect the swimmer. After working with computer vision though we found we could track the swimmer without interruptions from other swimmers. We have established that we can communicate well over FM, and now need to integrate the output audio with our sensors and controller.

2.4 Design Plan

We planned to use an IR or sonar sensor to detect the swimmer that will be connected to an Arduino inside the control box, but found that computer vision worked best for our product. When the sensor detects that the swimmer is at the T then Arduino will send a signal through an FM transmitter to a pair of FM headphones that the swimmer would be wearing. For it to work, we would need the same control box on both sides of the pool and a transmitter that does not reach the other side of the pool so the FM headphones can connect to the FM transmitter that is closer.

2.5 Engineering Standards and Design Practices

We are using the IEEE Standard for Floating-Point Arithmetic (IEEE 754) in our computer vision computations. We are using the Recommended Standard 232 (RS-232) for serial communication between the Arduino and Raspberry Pi.

3. Statement of Work

3.1 Previous Work And Literature

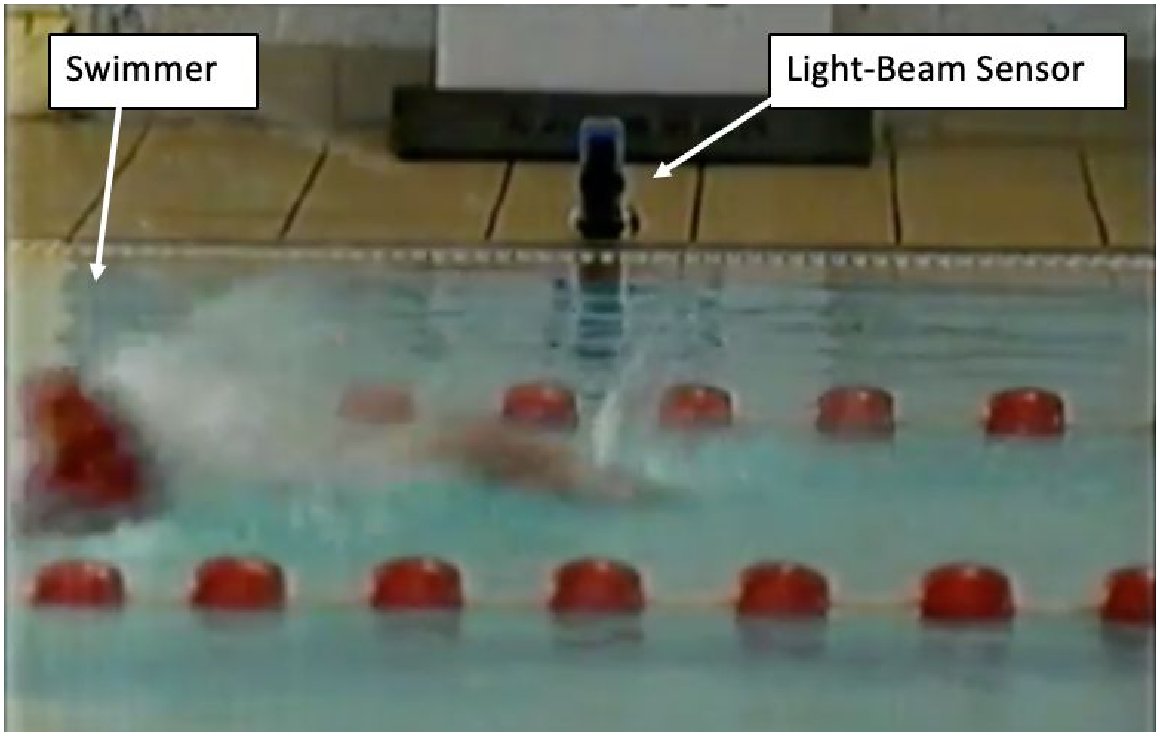

Figure 2: AquaEye device in use

In all the research we have done, we have only found one product that has a similar concept as ours. In 2009 a group created a device called AquaEye. The device has the same idea of tracking a vision impaired swimmer and then sending them a signal via headphones so that they know where they are located in the pool. The way that they track their swimmer is with a light beam, and when it gets broken the digital signal is sent back to the processor. In our product we are trying to get the specific distance of the swimmer away from the wall. With the light beam, it limits which lane the swimmer can use because it has to be placed on the outside of the pool lane.

If their product is used as shown above then no one would be able to swim in the lane closest to the sensor otherwise it would alter the performance of the device.

3.2 Technology Considerations

The strength of our device is that it will be able to function no matter what lane the user is in. With the AquaEye they can only swim in the lane near the edge. This advantage given to the user, makes tracking the swimmer more difficult. We need to be able to detect the swimmer no matter where they are in the lane, but we can't get any interference from the neighboring lanes, otherwise it will alter the performance of our device. There always seems to be negative side effects to improve the functionality of a pre existing device. Other options we have thought about include using a break sensor on the side of the pool, but the problem for this would be that the sensor would detect swimmers in other lanes so it would only be able to allow one swimmer to use it at a time.

3.3 Task Decomposition

For this project we have split it into tasks which include: find sensor and find communication. Within the find sensor we want to research, test, and conclude if the sensor will work and continue this process until we find a sensor that would work for the project. Then for finding communication we want to do the same thing but need to find a transmitter that will work underwater and can be connected to a pair of headphones that would allow us to tell the swimmer when they need to turn.

3.4 Possible Risks And Risk Management

One concern that we have is with the waterproofing so that the electronics do not shock anyone swimming, since the sensor might have to go into the water. It is also possible that the control boxes have water accidentally splashed on them, so they will need to be water resistant to ensure the device's durability.

Another concern we have is in finding a reliable sensor that will be able to detect the user without interference from the water or other swimmers. Depending on the sensor that we end up choosing, we will have to match the design of the control box to the positioning requirements of the sensor.

During prototyping, we have also had some issues transmitting a clear signal via FM to the user's earbuds in the pool environment.

3.5 Project Proposed Milestones and Evaluation Criteria

Key Milestones

- Finding consistent sensors for detecting the location of swimmer in the pool
- Choosing a transmitter and receiver for the signal to the swimmer for when to turn around
- Interfacing all of the components to work together
- Creating a final product that will be easy to set up and consistently aids vision impaired users

Tests

- To confirm we have them working, test components out of water
- Get components working in the water reliably
- Reach out to a blind swimmer to test functionality
- Confirm that product works with all four competitive strokes

3.6 Project Tracking Procedure

We have been tracking our progress this far with our weekly reports and presentations given to the instructor and our client. These reports have in-depth explanations that each of our team members have accomplished along the way. To make sure we are on track, we are using a timeline found in section 4.1.

3.7 Expected Results and Validation

The desired outcome for our project is to create a device that allows a swimmer with visual impairment to be able to detect the end of the pool before running into the wall using sensors and wireless communication devices. We will confirm the successful implementation of our device by first testing our device with a team member either wearing a blindfold or keeping their eyes closed while swimming. After we believe the device works reliably, we will have visually impaired swimmers try the device themselves.

4. Project Timeline, Estimated Resources, and Challenges

4.1 Project Timeline

Our original project timeline is below. We ended up having problems getting a sensor that worked well and Covid-19, so we weren't able to get our prototype together.

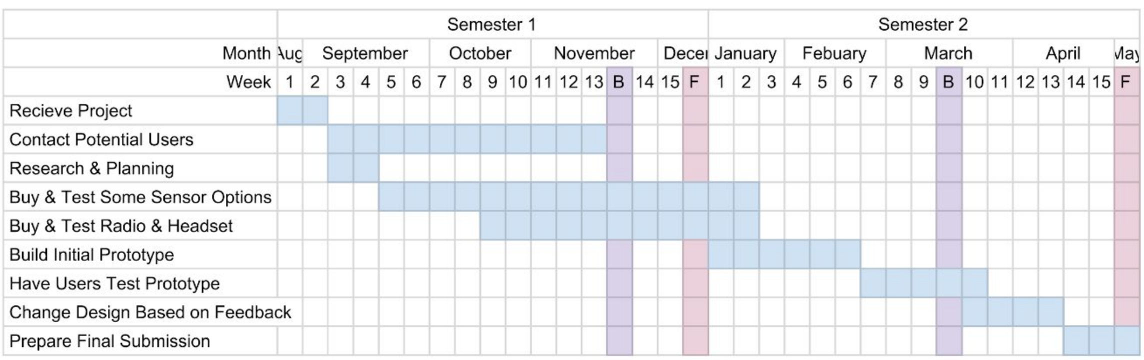

Figure 3: Project timeline

4.2 Feasibility Assessment

At the end of the project, we plan to have a working device that can help a vision impaired person to know when they reach the end of the pool. We think that this is a feasible goal. There are some potential challenges we may face in reaching this goal. One challenge that we faced was getting the sensors to work how we wanted with the water. For example the IR sensor would work as a break sensor for a lane, but we would have trouble if the swimmer was completely under water when the swimmer crossed that point because it does not detect objects under the water. Then we also had troubles waterproofing the sonar sensor to put it under water. We started with the sensor out and were getting good readings which we found out was due to the ground wire coming loose. After we fixed that we got it working again and put it back in the PVC elbow we are using to water proof, but did not get it in all the way and think water got on the wires or sensors or sensor so we are getting different result before and are working to figure out how to fix that now. We then started testing with Computer Vision with old testing videos and found that with it we could track the swimmer in one lane and be able to detect when the swimmer was close to the edge of the pool.

4.3 Personnel Effort Requirements

Table 1: List of personnel effort requirements

Task	Description	Estimated Hours
Contacting Potential Visually Impaired Users	Reaching out to visually impaired community members, ideally who are also swimmers, to learn more about the challenges they face and specifications for our project that we may not have known/thought of.	30 Hours
Research and General Planning	Brainstorming design ideas and evaluating the potential implementation of those ideas.	40 Hours
Sensor Testing	Testing swimmer detection sensors in a pool to determine what sensors will be ideal for our application.	80 Hours
Communications Testing	Testing devices to communicate with the swimmer when they are near the end of the pool.	20 Hours
Prototype Assembly and Internal-Testing	Integrating our chosen sensor and our communication devices into one system. Testing our system with team-members simulating vision impairment.	40 Hours
Design Revision and User-Testing	Getting feedback from our targeted users and making revisions based off of their feedback.	30 Hours

4.4 Other Resource Requirements

The sensors used to identify the swimmer's location that we have tested so far are the XL-MaxSonar sonar sensor and the Sharp GP2D12 IR sensor. We plan on testing various scenarios with each sensor to decide which gives us the most accurate data. Committing to the sonar sensor will require 2 units for each end of the pool while committing to the IR sensor may require 4 units, 2 for each end of the pool. We are using the Arduino system to interpret sensor data and send a signal to a communication device. The communications device we are using is a waterproof FM radio bud headset (UWaterK7) that will receive FM radio signal from a transmitter (Adafruit Stereo FM Transmitter) connected to the Arduino device. The sonar device is housed in a PVC pipe to keep out water and connect to the Arduino device. For the PVC casing we will screw the sonar into an elbow that is connected to a 2 ft PVC pipe that goes up to the pool's edge and to another elbow that will be connected to another 2 ft piece going to the control box. We also will use Beyer Halls lap swim pool to test and State Gyms pool when Beyer Hall is closed.We have now found that Computer Vision works best for this project and can be used with any camera, for example we are using videos recorded from our phones now since pools are closed due to the epidemic.

4.5 Financial Requirements

The FM radio headset (UWaterK7) costs $40. The XL-MaxSonar sensor costs $100. The Sharp GP2D12 IR sensor costs $20. BlueRobotics Sonar costs $250. The Adafruit Stereo FM Transmitter costs $20. The PVC piping costs $5. The Arduino costs $30. We will need a microcontroller for the device which will cost around $10. Depending on what sensor we commit to, the pricing will vary for the final product. Each will require the Arduino device, a waterproof box to house the Arduino, the FM transmitter, the microcontroller, and the receiver device. Assuming we need a box on each side of the pool, the total cost of the universally necessary components will be approximately $160. If we commit to the XL-MaxSonar, then the price of the sensor component will be $205 making the total device cost $365. If we commit to the Sharp GP2D12 IR sensor, then the price of the sensor components will be $80 making the total device cost $240. If we commit to the BlueRobotics Sonar, then the price of the sensor component will be $505 making the total device cost $665.

5. Testing and Implementation

1. We had to start by unit testing different types of sensors including sonar and IR sensors. We tested how they reacted to the water and if they could detect a swimmer in the water. For communicating with the swimmer, we plan to test an FM Transmitter and FM headphones to broadcast our audio.

2. We needed to test if the IR and sonar sensors would be able to track a swimmer's distance accurately. For communication we need to test the sound quality of the FM headphones and how far the FM transmitter would transmit.

3. To begin we wanted to test how the sensors reacted to water so we took them to the pool and had someone swimming to see what results we would get. We tried getting data, but we realized that we had to take a different approach in software to obtain usable data. Once we knew a little more about the sensors, we calibrated them to read correctly and then went to the pool to record data as someone swam. For the IR we found it most useful to line up the timestamp of the data we were receiving and then calibrate it with a video to determine how it functioned. For the MaxBotix sonar we set up a test for the swimmer to go back and forth to track what the sensor was reading. Also, with both sensors we tested the width of detection

.

4. For the IR we cannot implement it in the water so it will have to have the correct angle to perform properly. Our thought is that the sonar sensor would work better in the water and be able to obtain the swimmer's actual distance more accurately.

5. The IR sensor detects the swimmer, but the distance it detects the swimmer depends on the angle of the sensor. We couldn't obtain accurate distance measurements, but we did determine that we could implement this sensor in a digital application and use it as a breakpoint. The sonar didn't perform like we hoped. The voltage in and serial in ports

both have a filter on them, this filter requires that 3 consecutive range readings are within 1cm of each other to be considered a valid range reading. If the range readings are outside 1cm, the sensor discards the range reading set and reports the last valid range reading. The analog envelope port is the only one that doesn't have this buffer.

6. We made the changes in code to try and capture this waveform but ran into many difficulties. First, we discovered one of our wires had broken when we screwed into the PVC mount. We fixed that and were able to obtain readings above water on an oscilloscope. Put together some code and went to the pool. When we went, we weren't getting good results and starting messing with the sensor. Then we discovered that we had some water damage and that started to affect our results. After we broke the first wires, we put some Teflon tape and didn't screw it in all the way with a pliers. This allowed some water to leak in and damage the sensor.

7. Overall, the MaxBotix did not give us good results so we ordered a new sonar device that is more implemented for water. The MaxBotix 7072 is only rated for IP67 water conditions, so we had to make it more waterproof to submerge it in the pool. We will not have to do any waterproofing to the new sensor.

8. We also started looking into Computer Vision with our old testing videos to detect where the swimmer is and when they are approaching the edge. This has been working well and we have come a long way from using the IR and sonar sensors.

5.1 Interface Specifications

We planned on using either an IR or sonar sensor to detect the swimmer, but have now decided to use Computer Vision to do so. The sensor will send a signal to an Arduino, that will then send a signal through an FM transmitter to the FM headphones on the swimmer. This audio signal will tell them when they need to turn.

5.2 Hardware

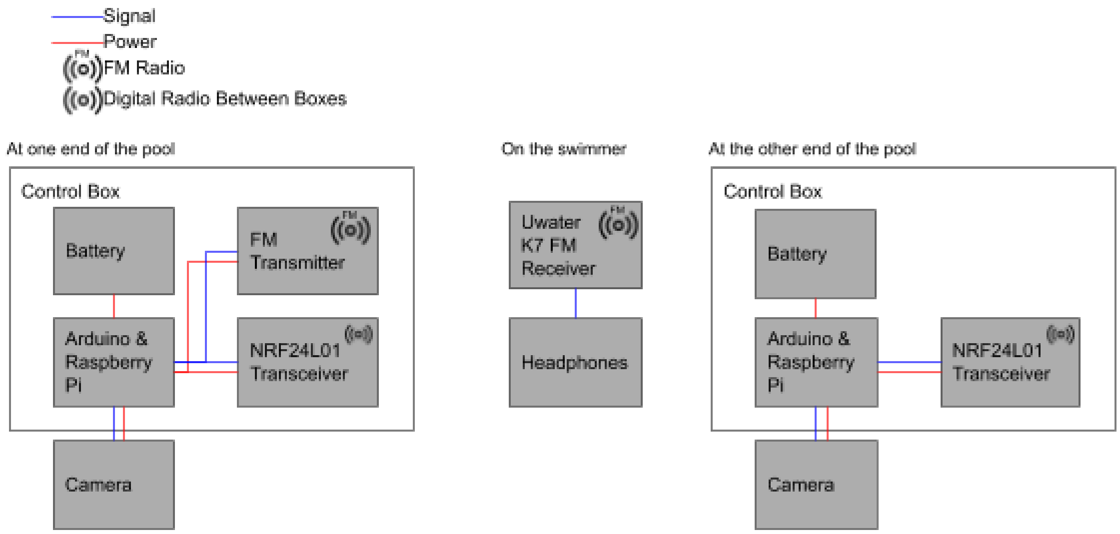

Figure 4: Diagram of the hardware components and their connections

At each end of the pool we will have a sensor to detect the swimmer. This is connected to an Arduino which processes the data from the transmitter and generates sound to be transmitted over FM. The swimmer has an FM receiver that connects to headphones, so they get the warning when they are near the edge of the pool.

5.3 Software

We are using a Raspberry Pi to compute the swimmer's distance from the edge using computer vision using C++ and OpenCV. The Raspberry Pi sends the calculated distance to an Arduino programmed with Arduino C/C++ libraries. When the swimmer is close to the edge, the Arudino generates a sound that is transmitted with the FM transmitter.

5.4 Functional Testing

For the function test we used the two sensors and the computer vision that we decided on from the non-functional performance test and took them to the pool. With both sensors we set them up and took measurements with a person in the water. We had the swimmer move back and forth to detect how close the swimmer would need to be. We also had the swimmer move side to side to see how far from the center of the sensor the swimmer would need to be. Then finally we had the swimmer swim towards the swimmer and away from the sensor to see if it would detect a moving object.

5.5 Non-Functional Testing

For non-functional tests we had to do performance testing at the beginning to decide on what sensor would work for the application. We started with IR and sonar sensors that we had used in

CPRE 288, but they did not have the range that we thought they did so we ordered new sensors with better range. To test this, we took the sensors and set them up at the pool and had the swimmer in the water swimming to see if the sensor could detect them. Then for computer vision we used the old videos and used it to detect when the swimmer got close to the edge of the pool.

5.6 Process

Before we began, we got a sensor and an Arduino Uno from the ETG that we used in CPRE 288 and set up the sensor and code to get the correct distance from the sensor.

When we started testing different sensors in the pool we originally took sensors we knew from CPRE 288 and tested them in the pool to see if the sensors would be able to detect the swimmer in the water when the swimmer was at the “T” in the pool that tells the swimmer to turn.

The test above also doubled to allow us to see what values we would get with these sensors when we used them in the pool because we were unsure how the sensors would react to the water.

Then we have found two sensors we believe will work so starting to order FM headphones, FM transmitter, and an Arduino with more storage space. We have got one pair of FM headphones and the FM transmitter and tried testing it out of the water, but the headphones we ordered did not have an FM function like the website said so we had to order another pair of FM headphones. We did find that the FM transmitter worked by turning it on and plugging our phone into it to play a song on the FM station and our car radio picked up the song when it was on that FM station.

Now we have found that using computer vision and videos from old tests we can track where the swimmer is in any lane and with some math calculation in the code we can find how far the swimmer is from the edge of the pool.

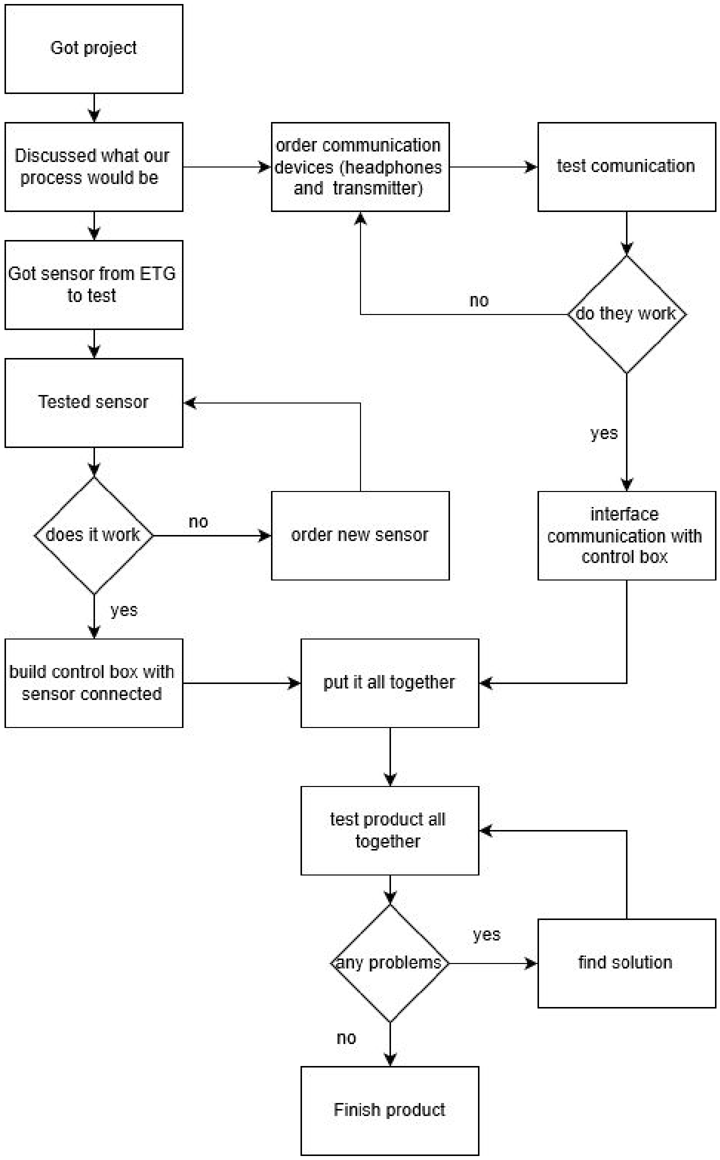

Figure 5: Flow diagram of progress

5.7 Results

IR Test 1

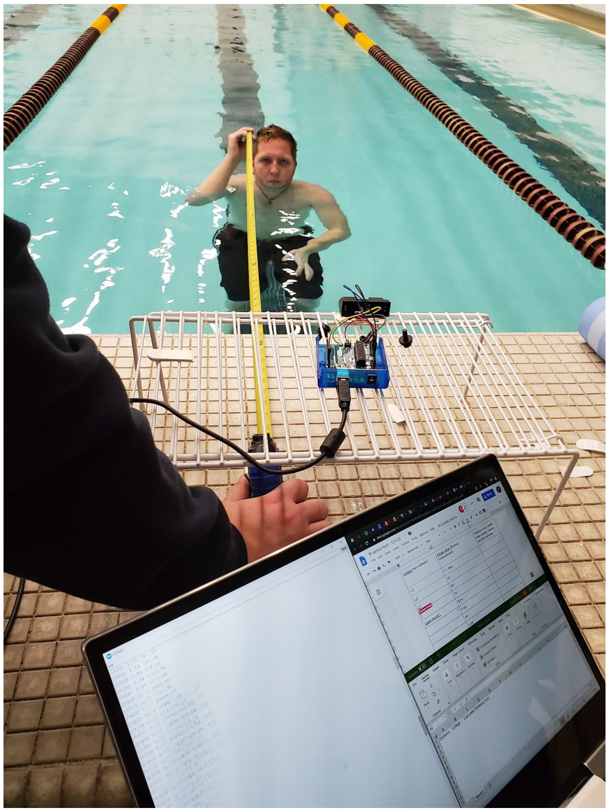

Figure 6: Image testing the IR sensor

We tested the sensor to see if it would work in the pool. The sensor was placed 7 inches from the edge of the pool. We measured the voltage from the sensor with the swimmer at different distances from the edge. We tried this several times with the angle of the sensor adjusted each time. The data is in the appendix.

We were able to show through this test that the sensor could detect the presence of the swimmer, when the swimmer was standing. However, we were not able to find any correlation with the voltage from the sensor and the distance of the swimmer from the sensor.

IR Test 2

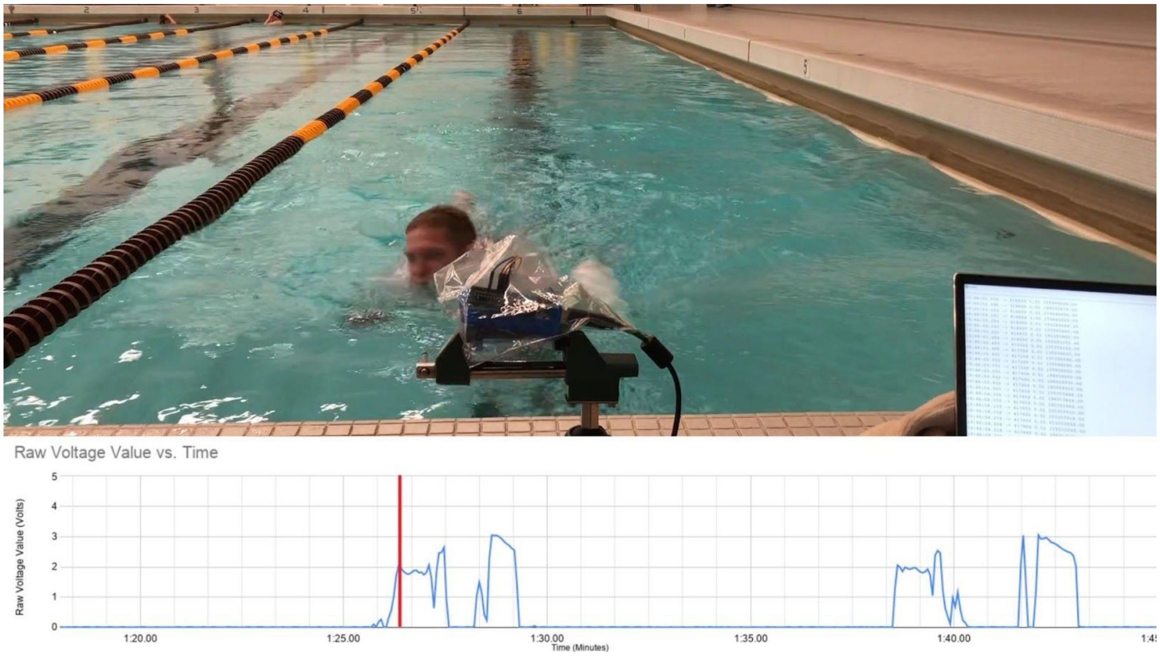

Figure 7: A frame from the video of the IR test with the sensor data overlaid

We took the IR sensor to the pool for a second test. This time we took a video and recorded the data from the sensor over several minutes. Afterwards we overlaid the data over the video so we could see how the sensors responded along with a video of the swimming. The video is linked in the appendix.

We found that the sensor was good at detecting when the swimmer was near the edge, but the sensor could not find the distance of the swimmer from the edge.

Sonar Test

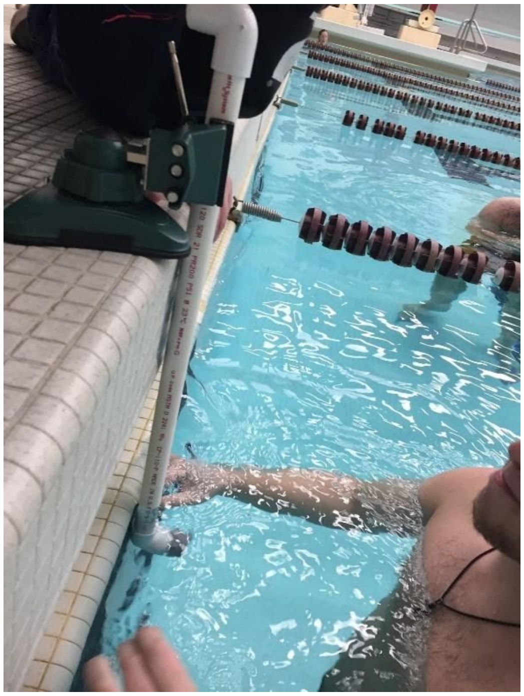

Figure 8: MaxBotix sensor in the water for testing

The next sensor we tested was the MaxBotix MB7072 XL-MaxSonar-WRA1. This is an ultrasonic range finder that was designed to be used above water, but we read on the internet that some people had success with this sensor in water (see Coconut Pi in references). We got the sensor working above water and were able to get it to work, but when we put it in water, we were not able to detect anything. See the appendix for graphs.

Radio Test

Inside the Beyer pool (Pool Length: 75 ft)

Signal Quality

0 - can't make out any music
1 - can hear static with soft music
2 - half static and half music
3 - low static and decent music
4 - decent music with no static
5 - good music with no static

Channel 1: 94.1

Table 2: FM Transmitter results from channel 1 in the pool

Distance from transmitter	Signal Quality (0-5)
0 in	5 until 2 inches under water
6 ft (T)	5 until 2 inches under

20 ft	5 until 2 inch under
60 ft	3 until 2 inch under

Channel 2: 89.3

Table 3: FM Transmitter results from channel 2 in the pool

Distance from transmitter	Signal Quality (0-5)
0 in	5 until 15 inch under water
6 ft (T)	5 until 10 inches under
20 ft	5 until 5 inch
60 ft	5 until 2 inch

Channel 3: 99.6

Table 4: FM Transmitter results from channel 3 in the pool

Distance from transmitter	Signal Quality (0-5)
0 in	5 until 5 inch under water
6 ft (T)	5 until 5 inches under
20 ft	4 until 5 inch under
60 ft	4 until 2 inch under

Computer Vision Test

Even though we could not go to the pool we were still able to test the Computer Vision by using old videos since it would use a regular camera. We have a three step approach. First, detect the lane boundaries. Next, find the swimmer's location. Finally, using the swimmers position in the frame relative to the lane, compute the swimmer's distance from the edge.

Detecting Lane Boundaries

The program first finds all of the blue areas in the image. It takes the largest of these areas and assumes that it is the lane with the swimmer. It finds the convex hull of this to fill in the rough edges. Finally, the program finds the boundaries of this shape.

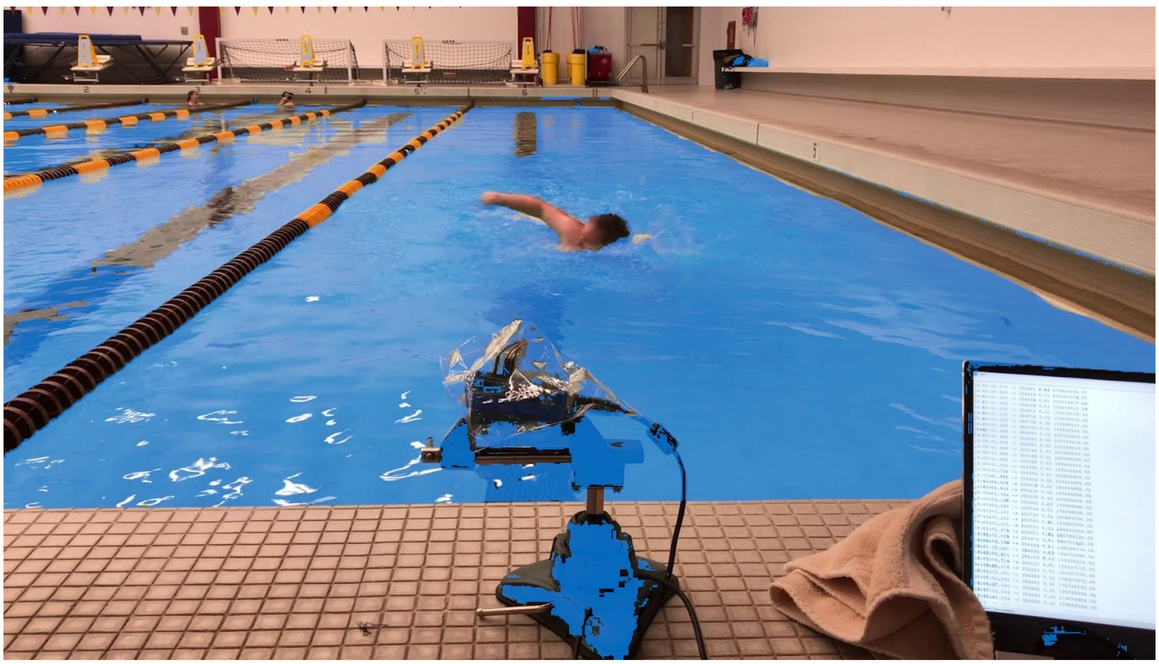

Figure 9: Frame with Blue Areas Highlighted

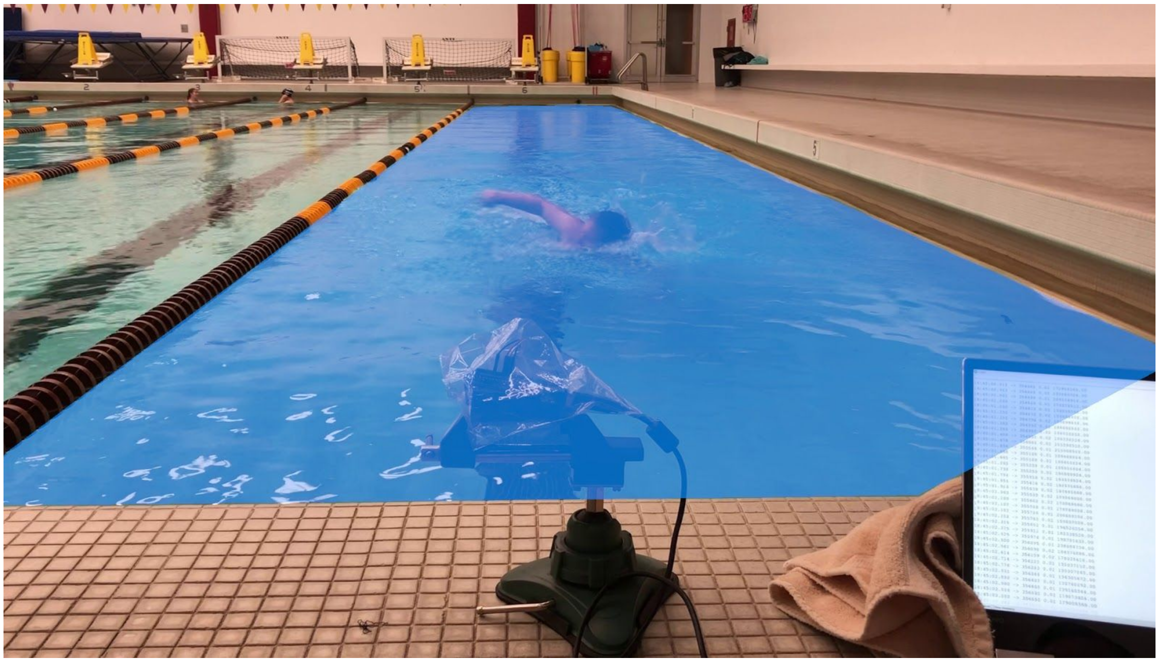

Figure 10: Frame with Convex Hull of Largest Blue Area

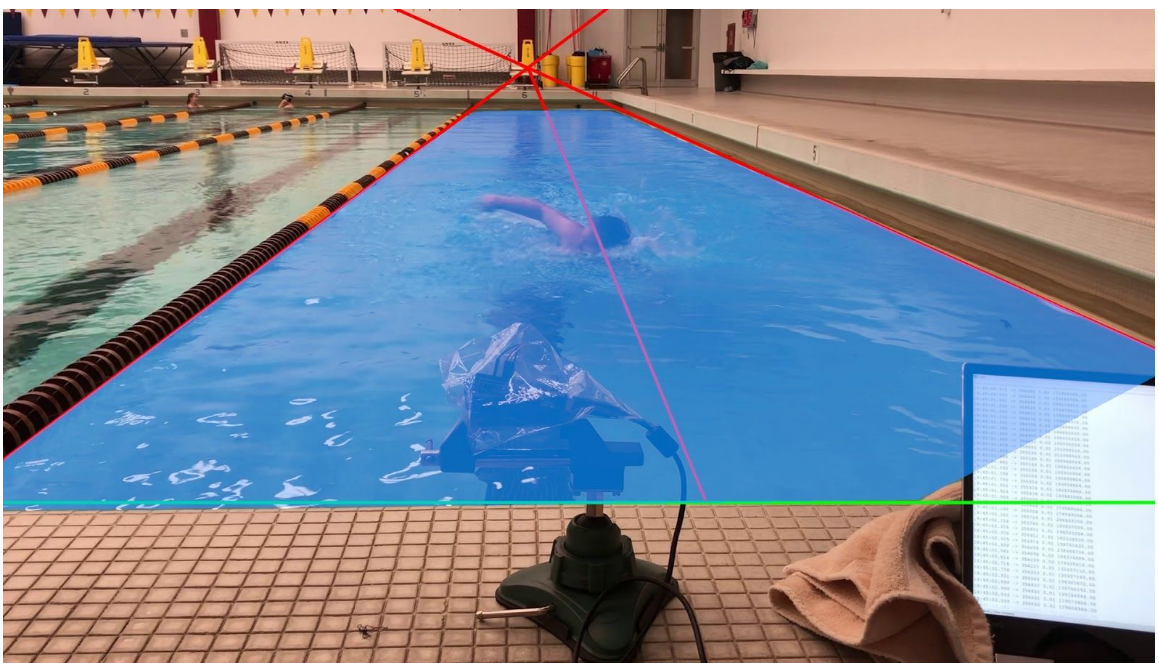

Figure 11: Frame with Lane Boundaries Overlaid

Detecting Swimmer’s Position

The program keeps a moving average of all the frames. It then takes the difference between the average and the current frame to highlight the places that have motion. Since the waves create reflections in the image it masks out all of the bright spots. It also masks out the frame outside of the lane. The program takes the brightest spot in the difference image as the position of the swimmer.

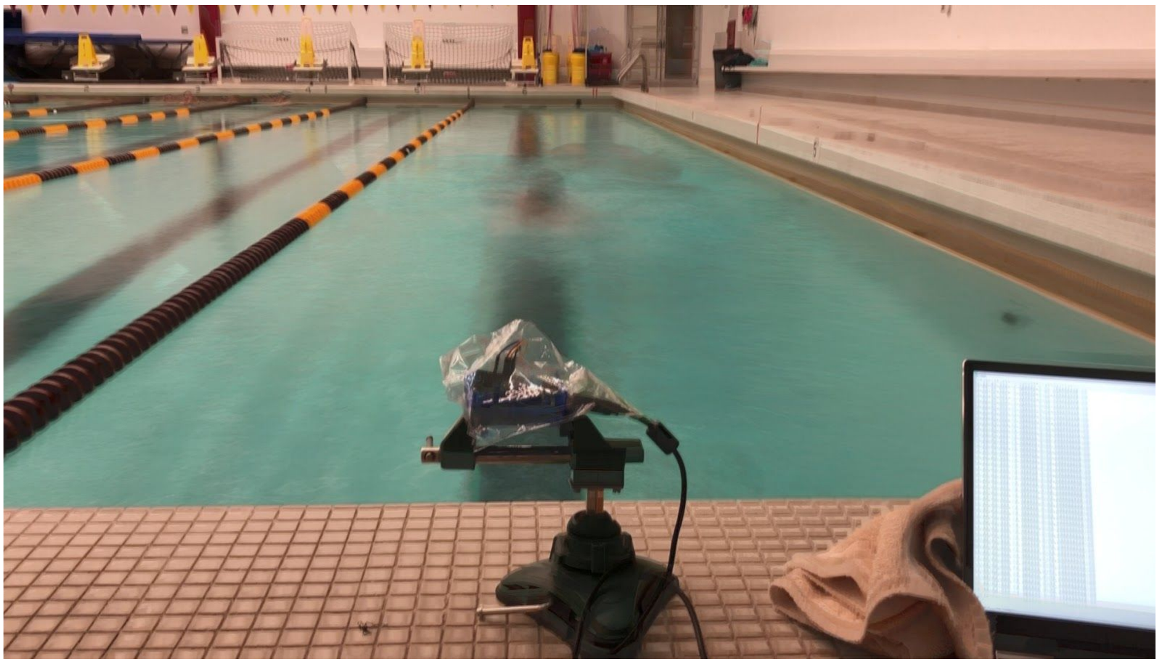

Figure 12: Moving Average of Previous Frames

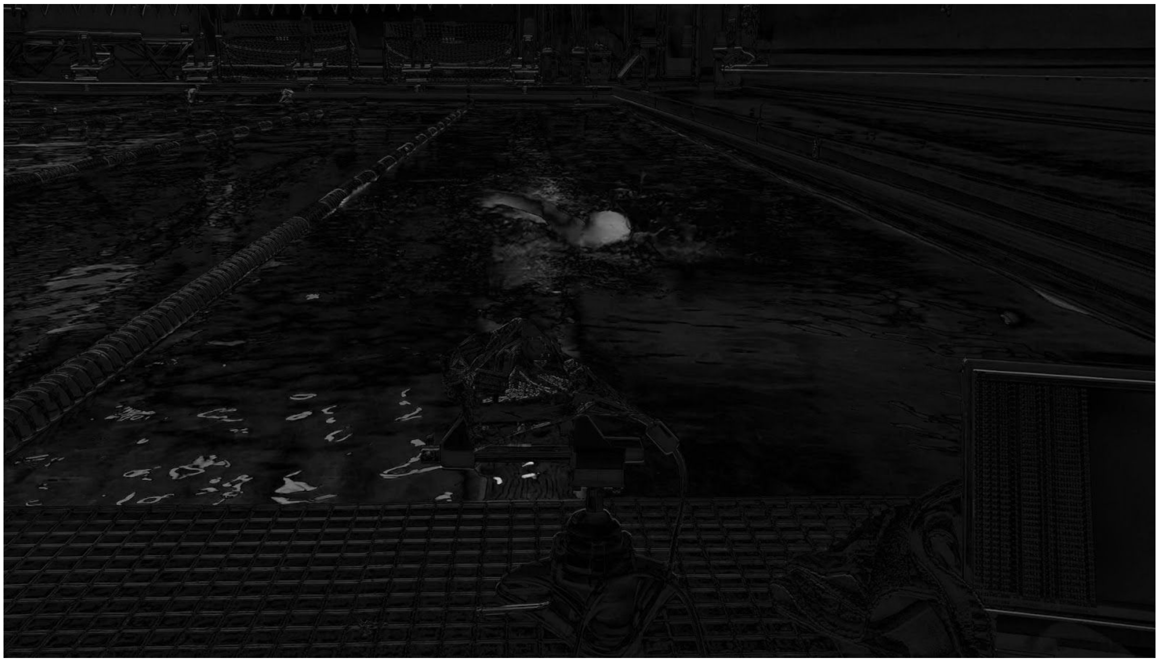
Figure 13: Difference Between Average and Current Frame

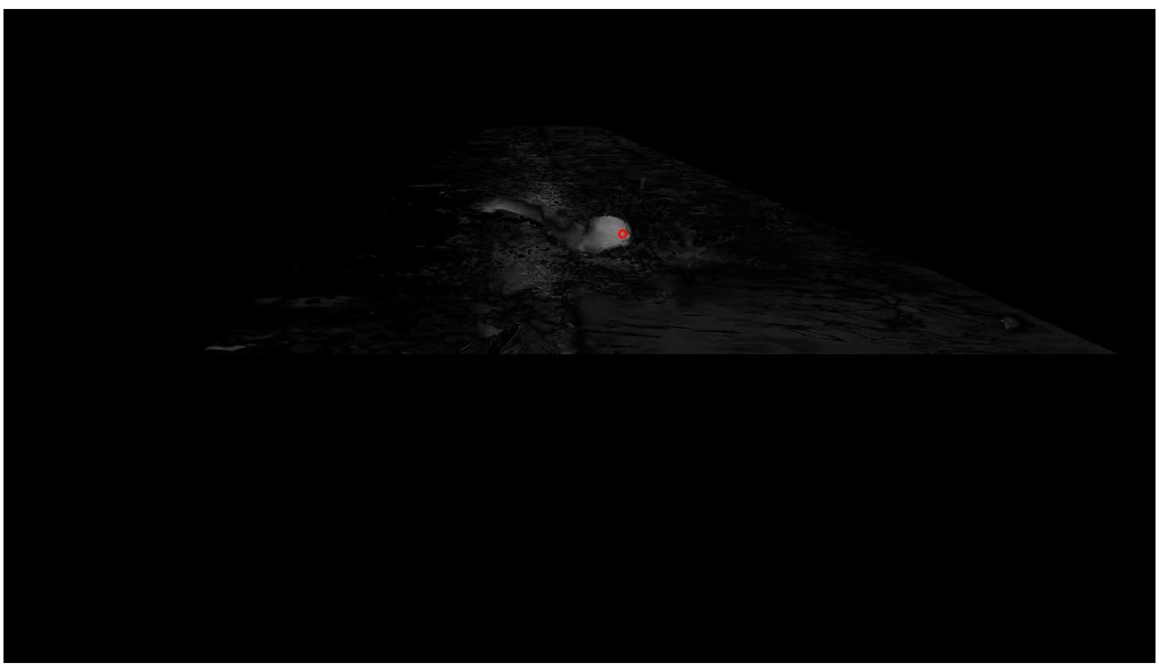
Figure 14: Masked Difference with Brightest Spot Marked

Computing the Swimmer's Distance

The program looks at the swimmers location in the frame relative to the frame boundaries. It treats the lane boundaries as a one point perspective projection and computes an estimated location in the pool. A video of the program run over our test video can be found at: https://drive.google.com/file/d/10i-ms0vg7NnMJNqkf9CHvek37B9R1CSn/view?usp=sharing

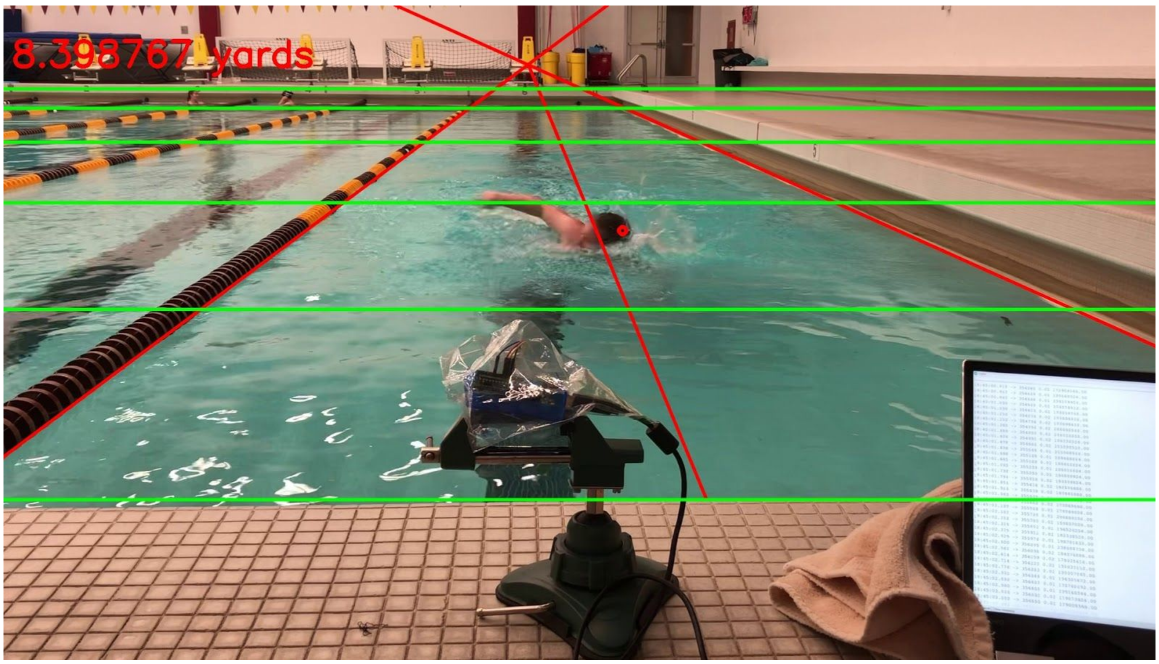

Figure 15: Frame with Yard Lines and Distance Overlaid

Conclusion

The program is able to correctly find the location of the swimmer and find a reasonable distance at least half the time. We were unable to verify the distance since we can't get to the pool due to Covid-19, but the distances outputted by the program make sense. The program shows that computer vision may be useful for this application, but it definitely needs to be improved in reliability before a human could rely on it.

6. Closing Material

6.1 Conclusion

We tested on multiple sensors to find which would be most reliable to detect the swimmer. In our testing we found that Sonar based sensors are not able to detect a swimmer on the surface. The IR sensor is able to detect when the user is near the edge, but can not determine its distance. Computer vision is able to detect the swimmer's distance from video footage at the end of the lane, though our implementation is not reliable enough for a human to use it for lane turning. More work on the computer vision and use of other computer vision techniques could improve its reliability.

We also found that the FM transmitter and FM headphones we tested worked in the lap swimming building and outdoors and found a way to interface the transmitter to the computer vision using both a Raspberry Pi and an Arduino.

6.2 References

Kieranjoc. "Visually Impaired Swimming Aid - AquaEye," YouTube, 2009[Video file]. Available: https://www.youtube.com/watch?v=qKPErSn6SGs . [Accessed: Oct. 15 2019].

C. Keller, "Overview of Common Swimming Strokes / Styles," *Enjoy Swimming*, 02-Dec-2019. [Online]. Available: https://www.enjoy-swimming.com/swimming-strokes.html. [Accessed: 08-Dec-2019].

E. Wei, "Individual Components testing and development," *Coconut Pi*, 23-Nov-2012. [Online]. Available: https://coconutpi.wordpress.com/2012/11/23/individual-components-testing-and-development/. [Accessed: 08-Dec-2019].

6.3 Appendices

IR Test Data

Table 5: IR testing results from 20 degrees

Actual value (physical measurement)	Voltage (V) (from software)
46 in	1.2 V
50 in	1.8 V
54 in	1.7 V
58 in	2.0 V
62 in	1.8 V
66 in	1.9 V
70 in	1.5 V

Table 6: IR testing results from 25 degrees

Actual value (physical measurement)	Voltage (V) (from software)
46 in	1.2 V
50 in	1.8 V
54 in	1.7 V
58 in	2.0 V
62 in	1.8 V
66 in	1.9 V
70 in	1.5 V

Table 7: IR testing results from 30 degrees

Actual value (physical measurement)	Voltage (V) (from software)
46 in	1.2 V
50 in	1.8 V

54 in	1.7 V
58 in	2.0 V
62 in	1.8 V
	1.9 V
	1.5 V

Video from the second test:
https://drive.google.com/file/d/1BZhlLMk9uyB8V4Svc9BUzt_XRbIsGeoN/view?usp=sharing

Sonar Test Data

These are the graphs of the acoustic waveform from the sensor. The initial spike is generated by the sensor. Subsequent spikes are sound reflected back to the sensor.

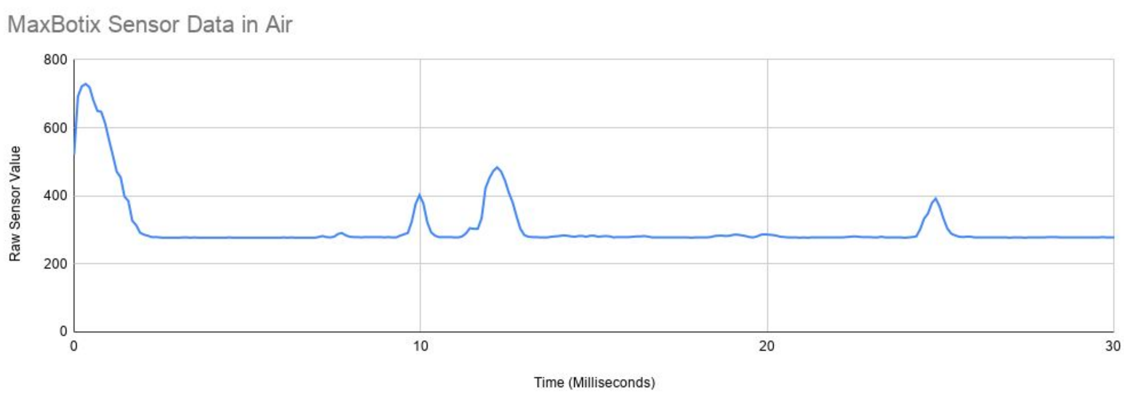

Figure 16: Sonar test data above water

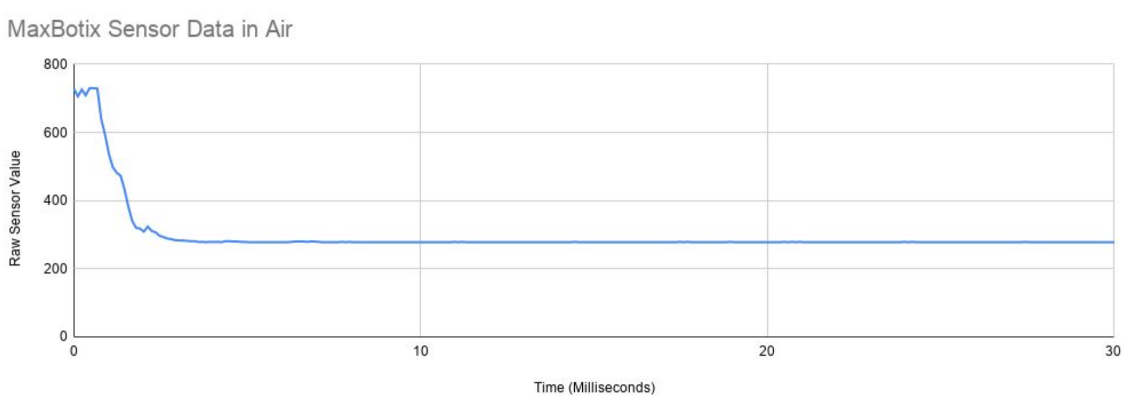

Figure 17: Sonar test data below water