An Investigation of a Computer Vision Controlled Robotic System

Evan Eliason*(1), Vince Jones*(1), Rachel Lauer*(1),
Bennet Outland*(1), Elijah Reitzel*(1), Logan Vojta*(1)

*All Authors contributed equally

South Dakota Mines
501 East St. Joseph Street
Rapid City, SD

evan.eliason@mines.sdsmt.edu, vince.jones@mines.sdsmt.edu, rachel.lauer@mines.sdsmt.edu,
bennet.outland@mines.sdsmt.edu, elijah.reitzel@mines.sdsmt.edu, logan.vojta@mines.sdsmt.edu

ABSTRACT

We investigate the performance of a robotic system that is solely controlled based upon manipulations to a video feed, otherwise known as a computer vision dependent system. We fulfilled a modified version of two 2022 Roboboat objectives while also considering conditions present and resources available. Through applications of robotic design, implementation and creation of computer vision systems, design of computer interfacing systems, and creation of parts through 3D printing, we were able to successfully complete the objectives. Ultimately, this project was instrumental in demonstrating our mastery of mechatronic systems.

Keywords

Mechatronics, Autonomous, Microcontroller, Microprocessor, Computer Vision, Robotics, Embedded Systems, Electromechanical Actuation System, Blob Detection, Arduino, Roboboat

1. INTRODUCTION

Our objective was to create an autonomous system that was able to traverse a set of modified courses developed for the 2022 RoboBoat Competition. The courses chosen were the "Navigation Channel" and the "Avoid the Crowds" challenges. However, the obstacle stanchions in the "Avoid the Crowds" challenge were removed for simplicity. The courses we used can be seen in Figures [1] and [2].

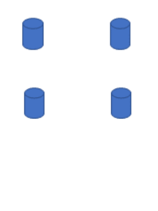


Figure 1: The Navigation Channel course layout. With the yellow line denoting the starting position.

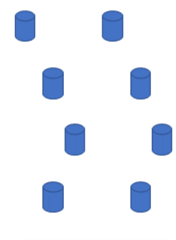


Figure 2: The Avoid the Crowds course layout.

The mechatronics project is aimed to allow students to compile subjects into a cumulative work. This idea is to be merged with Roboboat competition guidelines in order to challenge groups to achieve working models. The robots designed must traverse a team determined course in order to be a successful project.

Project requirements are kept in general terms, but some required components include the use of a microcontroller. For the vision processing requirements of the Roboboat competition, a single board computer is heavily encouraged. Other electronics components may be necessary for connecting other components such as DC motors for mechanical actuation. It should be noted that while the Roboboat competition is intended to be an autonomous surface vehicle, the main objectives of the competition can still be accomplished using a wheeled autonomous vehicle.

2. IMPACT

2.1 Ethical

This project meets the second Fundamental Cannon of the NSPE (National Society of Professional Engineers) Code of Ethics for Engineers: perform services only in areas of their competence. The goal of the project is to design and manufacture a robot using the knowledge gained throughout the course. Secondarily, course knowledge should be applied to the project for the purposes of component selection, wiring, and programming.

Different areas of knowledge are to be documented and referenced within the project to connect the robot to learning objectives met during the semester. The main objectives within focus of this project team were mechanical systems and digital communication.

The project also directly meets the other five NSPE canons to a lesser extent. Methods by which the other canons are met are outlined in the following two sections in greater detail.

2.2 Societal

Autonomous and semi-autonomous vehicles allow people to explore areas that would be dangerous for any life to enter. Firefighters can use semi-autonomous vehicles to search a burning building, divers can use an autonomous submarine to explore deeper into the ocean, and drones can be used by farmers to remotely check on crops to ensure they are growing safely, as a few examples. Creating machines that allow people to more safely perform their duties will lower risk and increase the opportunities to those who were unable to perform the job beforehand due to constraints. Creating devices that can help individuals complete tasks more efficiently, effectively, and safely reflects the fundamental ideals expressed in the first NSPE canon, which describes the importance of "holding paramount the safety, health, and welfare of the public."

2.3 Global

Through this project we make an impact globally through participating in the global marketplace and open sourcing the code used for the project under the GNU General Public License Version 3. As for engaging the global marketplace, our components and tools used in this project were manufactured in many different places including, but not limited to, the United States, United Kingdom, and China. This list does not include the many different countries where the raw materials were acquired. We are also graticious of the open source community and their drive to educate others through programming in addition to sharing their work to prop everyone up. In response to this, we have decided to publish our programs we created under the GNU General Public License Version 3. This is a permissive license to ensure that freedom in software continues through the use of our programs.

3. ROBOT DESIGN

3.1 Mission Statement

Fulfill Roboboat objectives while also considering conditions present and resources available. Teams must decide if it is better to go land or water based. This project will require a combination of CAD work, programming, 3D printing, and hardware implementation. In order to accomplish this, teams will divide members and complete subcategories in order to build the project in an effective and efficient manner.

3.2 Overall Specifications

The robot consists of an electromechanical system with two drive wheels and a 3D printed chassis. In order to better meet the needs of people with autonomous applications, a wheeled robot design was selected as being more widely applicable than a surface vehicle designed for water. The 3D printed chassis is printed out of polylactic acid, more commonly known as PLA, which is a common 3D printing filament that is also biodegradable plastic.

The robot consists of a series of subsystems that utilize computer vision in order to gather information from the environment. In order to do this, a Raspberry Pi 4 and an ESP-32 are used. The Raspberry Pi takes in input from the camera in order to determine what movement to make using computer vision. When a choice is determined, an output command is sent to the ESP-32 in order to send power to the wheels.

3.3 Design Considerations

The robot was designed to function on land due to logical and environmental considerations. A boat proved challenging since testing would be risky since the boat could sink or lose signal in the middle of a body of water. The boat would also affect the environment if hazardous waste such as plastics and battery acid were dumped into the water. By using basic methodologies of electronic measurements, the team was able to produce an acceptable model in order to accomplish objectives. The open concept body of the car accomplished with this project allowed for easy implementation of hardware, as many components were needed for implementation of software. This open-topped design also allowed for a reasonable size while making sure heat was dissipated. Components were chosen for cost effectiveness while having enough quality to function as intended. This was the methodology considered when designing this mechatronic system.

3.4 Raspberry Pi Functionality

The Raspberry Pi functions as the microcontroller for the data collection and analysis subsystems. A Nexigo N60 camera is connected to the Pi and used for visual input via a computer vision algorithm incorporating functions from the OpenCV library. Due to limitations of hardware the camera is not able to continuously check for input and instead checks at a variable time meant to be as small as possible. The program identifies stanchions and separates their general shape from the background using blob detection. The algorithm then analyzes the data to determine the robot's best course of action (i.e. left or right). It will then create a binary character based upon an analysis of the computer vision algorithm results and send the character to the ESP-32 via a serial interface.. Please see the following pseudocode for this process and reference the Appendix for the code flowchart.

```
==========================================
Inputs: Capture from camera and scaling factor
Outputs: Motor direction character
==========================================
Initialize the camera
while True
        Scale down the frame
        Convert to HSV color format from BGR
        Create a mask of blue colored objects
        Blob detection parameters for filtering
```

Blob detection

Determine two closest blobs

Get x coordinates of the centroids of the blobs

turn buffer = 0.15

if less than two blobs are detected

Move forward slowly

else

Determine center point between the centroids

if Middle of the Frame > (1 - turn buffer) * centroid center and Middle of the Frame < (1 + turn buffer) * centroid center

Move forward quickly

else if Middle of the Frame > (1 + turn buffer) * centroid center

Turn left

else if Middle of the Frame < (1 - turn buffer) * centroid center

Turn right

Else

Move forward slowly

Through the use of this computer vision algorithm, we were able to practice signal conditioning through filtering the results of the blob detection algorithm.

3.5 ESP-32 Functionality

In the case of our project, the ESP-32 is mainly used to control all of the subcomponents; it does not handle any higher processing for computer vision. In order to operate the smaller subcomponents, the ESP-32 will receive commands through binary characters denoting the different actions from the Raspberry Pi and use them to properly control the motors.

==

Input: Binary character through serial

Output: Motor direction

==

Listen on serial

if serial message equals ‘f’

send full power to both motors

else if serial message equals ‘r’

send 66% power to left motor

else if serial message equals ‘l’

send 66% power to right motor

else

send 66% power to both motors

3.6 Microcontroller Interfacing

The ESP-32 and Pi read different types of data and therefore a method to interface the two was required. Robot commands from the Pi were converted into a binary character and transferred via a serial connection. For this connection, a USB A connection on the Raspberry Pi was connected to a USB micro B connector on the ESP-32. A serial interface was used due to its reliability and simplicity. WIth both the Raspberry Pi and ESP-32 having support for Bluetooth communication, Bluetooth was an option for sending commands. However, for increased stability, a wired connection was preferred. As both microcontrollers are able to process binary commands, this method allows for successful interfacing. This demonstrates the course learning objective of computer interfacing.

3.7 Circuit Overview

The circuit used was created so that the Raspberry Pi would be able to interface with the ESP-32 via a serial connection using a USB A to micro B. The ESP-32 will then interface with the motors. See Figure [3] for further details.

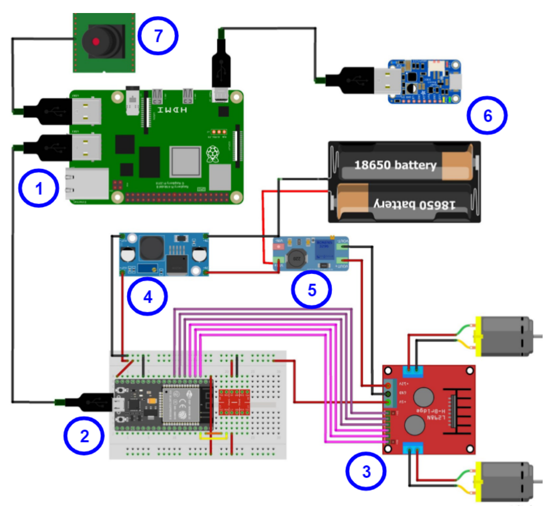


Figure 3: Here are the different parts of the system as indicated by their number:

(1) Raspberry Pi 4B, main computer processing unit

(2) ESP-32, smaller microcontroller to handle subcomponents

(3) LN298 Motor Driver and two DC motors for the drive wheels

(4) Step down converter to change voltage from 18650 battery pack to 5V for the ESP 32

(5) Step up converter - converts the voltage to 24V for motor control, where the higher voltage helps the motors run faster

(6) Battery packs. Upper one represents the Anker rechargeable battery bank used to power the Raspberry Pi. Lower one is the rechargeable battery pack using two 18650 lithium ion batteries

(7) USB camera for computer vision

3.8 Drive System

The drive system of the robot is designed such that there are three points of contact with the ground. This includes two powered wheels and a passive caster in the back center of the chassis for support. The two powered wheels are each individually driven by their own motors. Secondarily the caster supporting the back of the chassis is actually a ball transfer unit. The team decided that choosing an actuator setup as such would prove to be the most effective considering cost and maneuverability.

As for how the system moves, there are simple motion commands given to the ESP-32 by the Raspberry Pi depending on the vision input from the camera. If the camera and computer vision algorithm recognizes a stanchion, the algorithm will determine if the stanchion is centered, to the left, to the right, or unable to be seen. Each of these classifications then result in a simple motion command executing. If the robot sees a stanchion on the left or right, it will turn the opposite direction to adjust its course and avoid a collision. If the robot sees that it is centered between two stanchions, then the robot will move forward. If the robot is unable to see stanchions, then the robot will creep forward slowly. As a result, this ends up being a fairly accurate navigation algorithm.

4. RESULTS

4.1 Project Outcome

After the team was able to set up initial requirements, the final mechatronic system was seen. The team was able to create a robot capable of maneuvering a course, but not without traversing through a medley of problems. There were software bugs not allowing the robot to properly evaluate stanchions which would affect its movements. Physical problems were also present, which entailed the robot losing traction because of improper weight distribution. This also affected turning as commands could not be executed when the movements did not match what the software wanted. After these problems were fixed, a robot that could consistently function and mostly finish the course was able to be achieved. This robot is a Terrain Investigational and Navigational Automaton. Hence, it will be listed as T.I.N.A for short.

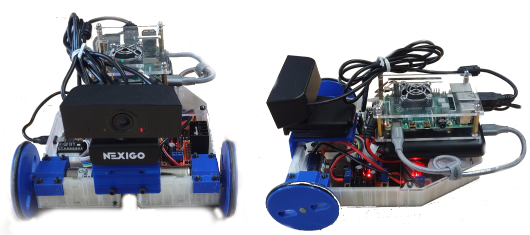


Figure 4: Photographs of the completed robot.

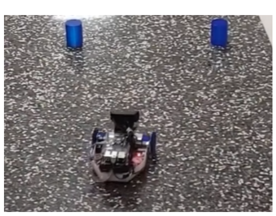


Figure 5: Photograph of T.I.N.A driving through the "Avoid the Crowds" course.

4.2 Issues and Challenges

The first issue encountered was the camera field of view. The initial camera used was the Logitech C270 with a FOV of 55 degrees. However, this field of view was too small and T.I.N.A had problems navigating from where she was on the track. Because of this, we changed our camera to a Nexigo N60 that has a FOV of 110 degrees.

Additionally, it was a bit difficult to interface with the motors through an ESP-32 controller, because most libraries for these are designed for ATMega microprocessors. Alternate code was written and applied to mitigate this issue. Additionally, during testing the wheels caught a lot of dust from the ground, which made the car "slip". Because TINA slipped, she couldn't steer, and was at times unable to navigate the course very well. The solution to this problem was wiping the wheels of the car in between tests, which increased traction.

Another issue discovered was that the Raspberry Pi CPU is not powerful enough in terms of video processing capabilities to get fast results while driving the car. Because the Raspberry Pi CPU had these limitations, this significantly hampered the performance. Finally, the last issue mechanically was there was too much weight on the rear of the car. The original design was not intended to support the weight of the Raspberry Pi and an oversized battery bank to power the Raspberry Pi. As such, the motors did not always work optimally, and some weight had to be re-distributed differently across the chassis. This allowed members amongst the group to deal with issues amongst the team and allow for a resolution to be found.

5. CONCLUSION

This project involved work with a wide array of subjects and various methods of problem solving in order for the team to be successful. Working on the Roboboat allowed a wide range of skills and disciplines to be acquired. By allowing teams to group together and work on a challenging project, members were better able to communicate and collaborate. This allowed for the course learning objective of teamwork, analyzing engineering problems and all student learning outcomes to be recognized and fulfilled. The robot proved to be problematic at times, but allowed for fulfillment by working through hurdles faced. This team was able to demonstrate fundamentals of an effective team to be able to accomplish a project successfully.

6. ACKNOWLEDGEMENTS

We would like to thank South Dakota Mines Rocker Robotics for letting us use the hardware that we lacked in the creation of this robot including a kit that was modified for this work as well as the Authors of OpenCV for creating a great computer vision library and documentation. Also our appreciation is sent to Dr. Fekrmandi for choosing this subject matter for the final project, as it was all started from such a decision.

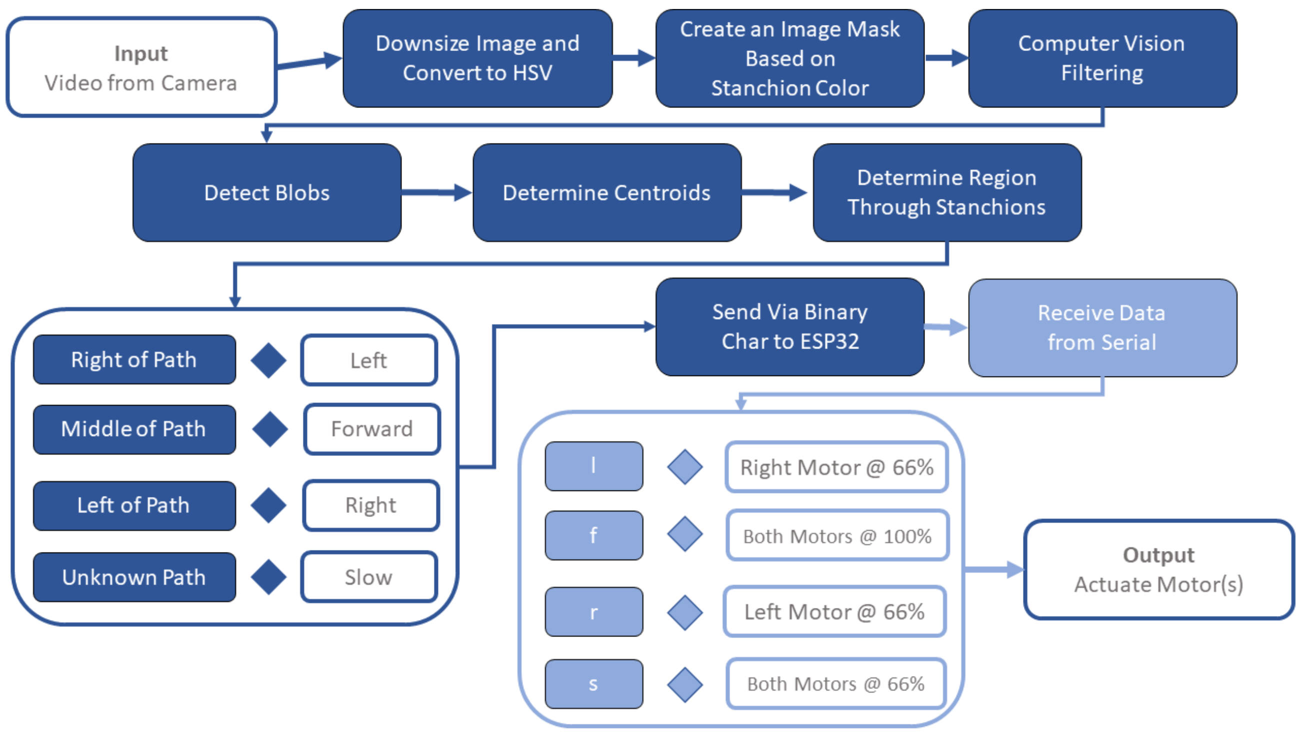
7. REFERENCES

Bolton, W. (2018). Mechatronics (7th Edition). Pearson International Content.

Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools.

Outland, B (2022) BuoyCV [Source code]. https://github.com/outlander1701/BuoyCV.

APPENDIX



Code flowchart for the project where the darker blue signifies operations completed on the Raspberry Pi 4B and the lighter blue signifies operations on the ESP-32.