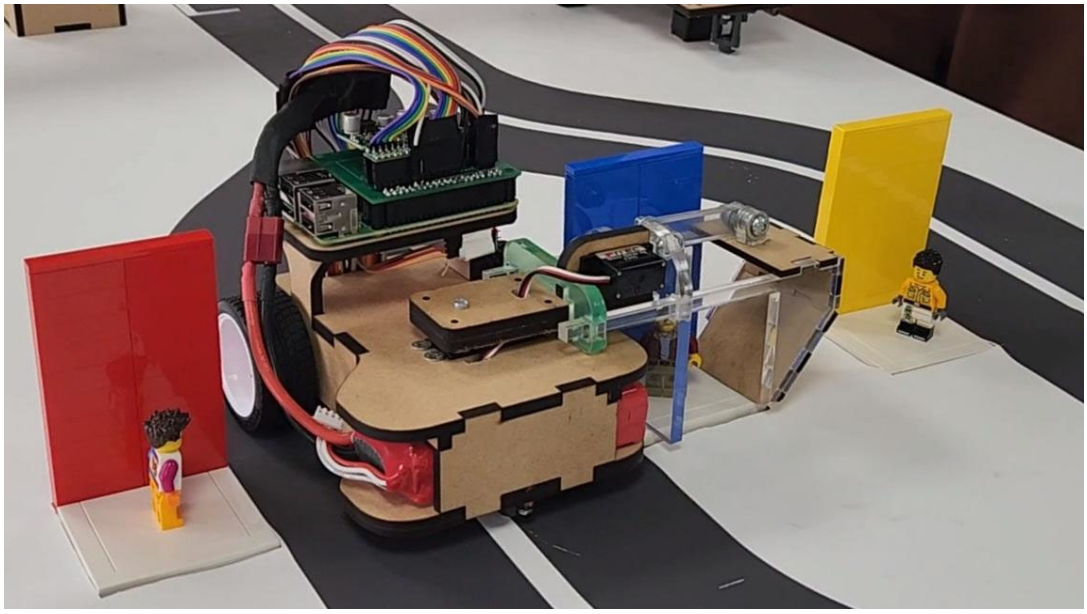
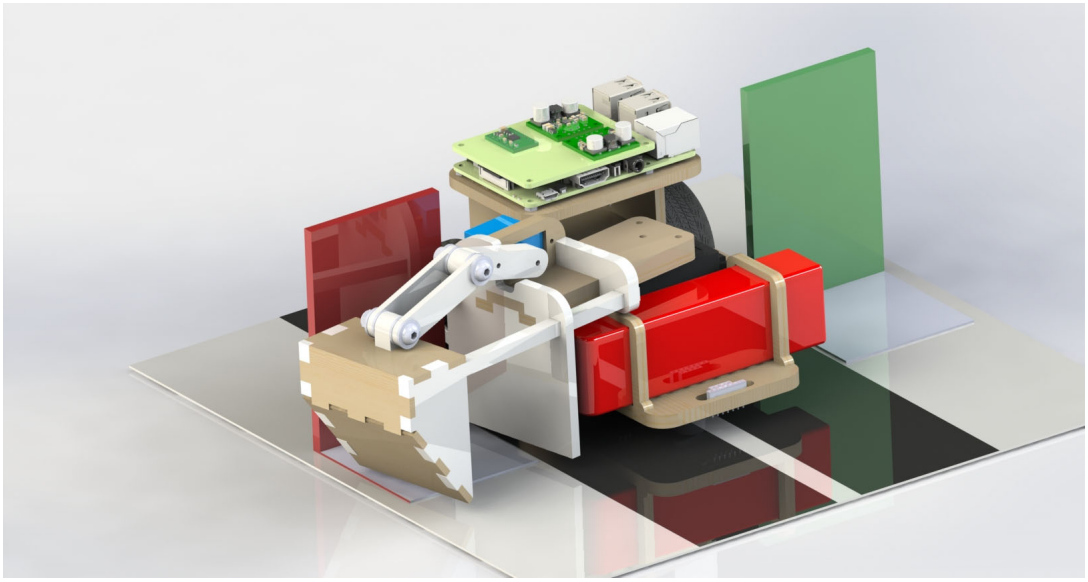
Team 8

ENGMT280 - SAV Report

Group 8





THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

Team 8

Introduction

Added safety, greater convenience, and increased eco-friendliness compared to regular vehicular transport - with those advantages surely Smart Autonomous Vehicles (SAVs) are the way of the future! Or are they? This report details how our team developed and made an SAV to navigate a small-scale track designed to imitate a real-world “smart city”.

We learned first-hand how to devise innovative solutions to problems that mimic real-world issues. The SAV needs to be able to start from a home position, navigate the track, identify junctions, and take the appropriate directions to pick up the correct customer (a LEGO Minifigure) and drop them off at their desired destination. Designing a suitable chassis, picking the right sensors and their locations on the SAV, and developing a reliable pick-up mechanism were some of the challenges that we faced and ultimately overcame in the process of completing this project.

Through the task we gained a better understanding of some of the challenges engineers face when designing SAVs and other technologies for the real world, helping equip us to solve challenging problems we may face in the not-too-distant future.

Problem Statement

The transport of the future will be safe, fast and autonomous. Creating a smart autonomous taxi to operate on a network of cameras in a smart city presents many problems and requires a complex solution.

The main problems include:

Autonomous navigation: Creating a robust autonomous navigation system that is safe, efficient and reliable in avoiding obstacles whilst navigating complex urban environments.

Camera integration: Developing seamless integration with a smart cities surveillance network for safety and navigation.

Safety and reliability: Implement safety systems/protocols to minimise potential accidents and improve the reliability of the autonomous taxi.

Scalability: Create a system that can evolve and grow as the smart cities' infrastructure requirements develop over time.

And many more complex issues.

Solving these issues is essential to providing autonomous transport that can seamlessly integrate with a network of surveillance cameras to provide a smart city with safe, reliable and convenient transportation.

SAV Mechanical Design

Challenges during the Design of V1

During the CAD/Mechanical design phase of the project, we had fairly limited knowledge surrounding both the task at hand and the general movement and functionality of materials/resources we had access to. This made the initial design phase somewhat challenging. This meant that our V1 had a lot of educated guesses in the dimensions of component sizes, which could intern impact our performance.

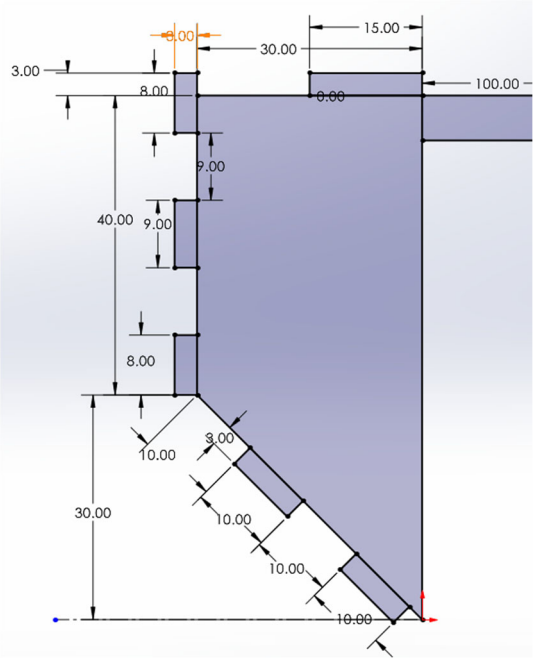


Figure 1: Dimensions of the ScoopSide V1

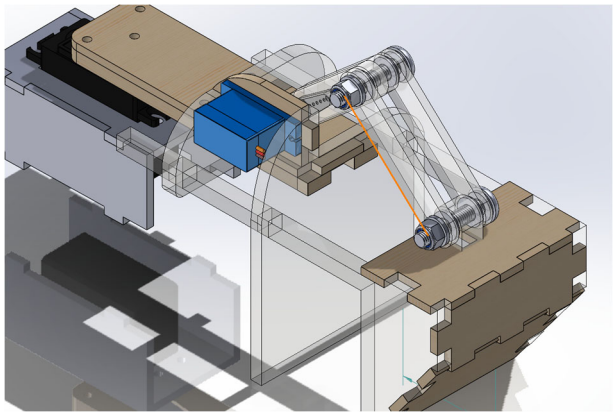


Figure 2: Render of Scoop V1

An example of this guesswork would be the angle of the base plate of the scoop part of the mechanism. The initial design for the sizes of the Volumes within the scoop was largely guessed at. While this didn't cause issues later on since the angle was great enough that the Lego Minifigure would always slide down, we probably could have gotten away with a shallower angle and a generally smaller space. Which as explained below would've helped us during the earlier design phases when trying to meet the SAV overall dimension requirements

During the initial design phase, the last piece of the puzzle was designing a location for the RPi and PCB, as shown on the left, it was too tall to go under the current design of the arm. And as shown below it was too wide to fit in sideways. Our solution to this was to build its own platform and raise it above the arm. This proved to be a wise decision for the assembly and testing phases later on when we need quick easy access to it. However, it would've been aesthetically pleasing to have it completely encased. We designed some potential solutions to this which are detailed below.

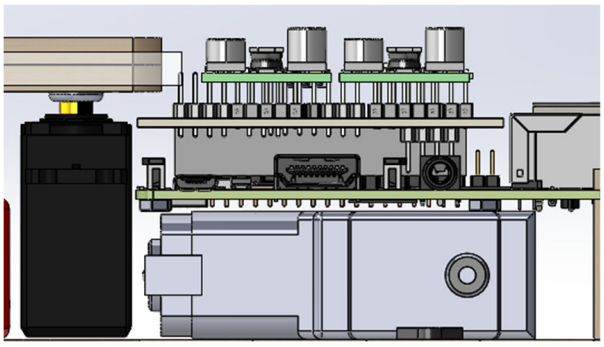


Figure 3: Side view of RPi and Gearbox

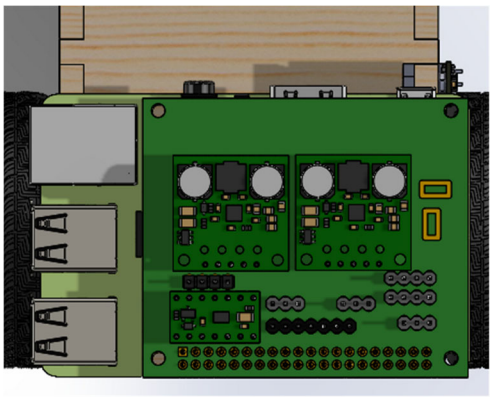


Figure 5: Top View of P short Axis Mounted PCB

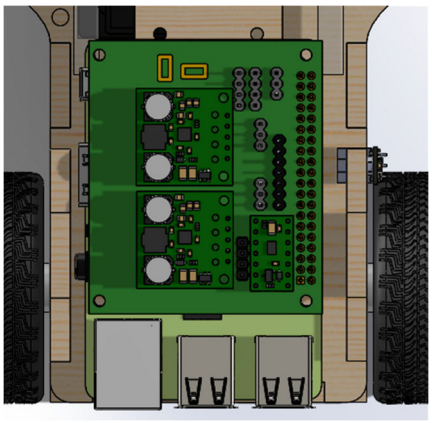


Figure 4: Top of View of Long axis mounted PCB

Our design avoided the need for rotation over the top of the SAV since wouldn't need to orientate the scoop differently for each side. However, this made getting within the 210mm Length requirements challenging.

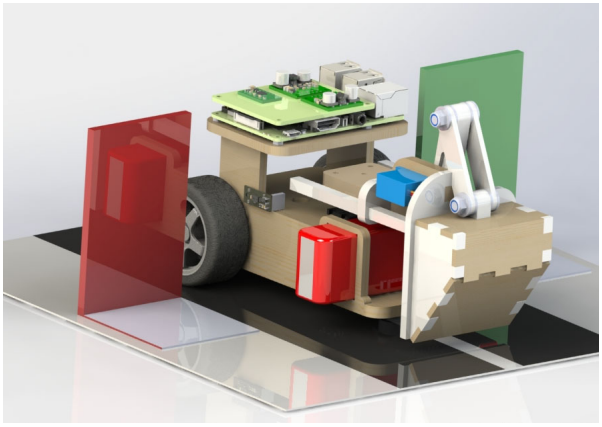


Figure 6: Render of V1 on Mockup Track

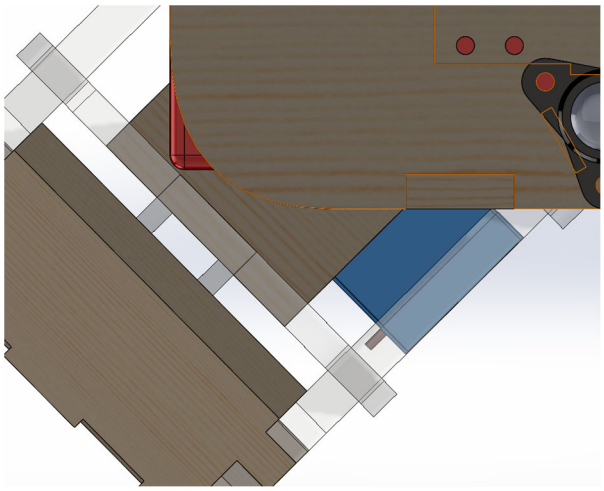


Figure 7: Battery and Scoop Collision

The distance from the centre of rotation of the scoop arm had to be consistent on both sides. This meant that the length of the arms was largely determined by the track dimensions. As seen in the image on the right the gap between the battery and the scoop plate, was incredibly tight. To achieve this some incredibly tight chassis planning was required to fit within the regulations and still clear the battery.

While it was possible to calculate the distance required between the centre of the scoop and the distance sensor to detect a pickup location. It was beneficial to place the car on a CAD mock-up of the track to gain an understanding of the size of the car and if there were any issues we hadn't foreseen. One issue that was also solved here was making sure scoop height was optimised to only just clear the pickup platform. Any extra height would cause issues in a pickup.

Problems and Solutions of V1

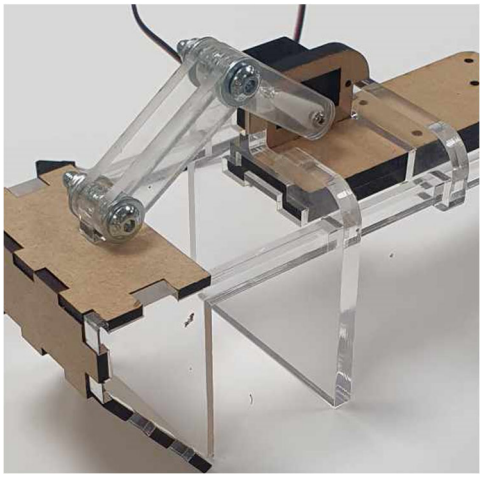


Figure 8: Physical Model of Scoop V1

One of the biggest problems with our initial design was that the tolerances for the sliders in the tracks were not great enough to create a smooth slide that the servo was capable of opening and closing. This was due to our limited knowledge of component design because this decision was made very early on in the process. Another thing that made this tricky was that the line of action of the Servo and its mount on the Scoop lid wasn't parallel with the sliders let alone inline, which caused increased torque in the system which worked against the motion. Once again this was due to our limited experience in mechanical design.

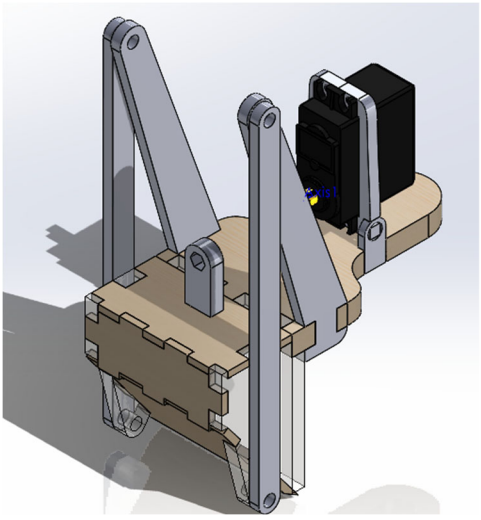


Figure 9: Concept Render of Scoop V2

A potential concept solution to the servo not being powerful enough to overcome the slider's friction was to remove them altogether and have the scoop pivoted off a high swing arm and brought in by a lead screw. While mechanically this system was likely better. The height and thus the mass of the arms required to make the arc length of travel at the bottom of the scoop approximately a straight line would've taken us outside of our regulations.

Our solution to this problem was to just file back large amounts of material in the slider mounting holes to reduce friction. This worked far better than we thought it would. However, this introduced a lot of play into this system which caused some issues when trying to stop the scoop from getting caught on the platforms.

The best solution would've been to combine the two systems together to have sliders that are lead screw driven. Ideally, we could've used some sort of kitchen draw-style runner. The telescopic approach would've helped with the clearance of other parts on the SAV however the difficulty of introducing these would've been too complex.

An issue that cropped up very quickly during our initial testing and assembly was how difficult the wires were to keep concealed so that any moving parts wouldn't get caught or held up.

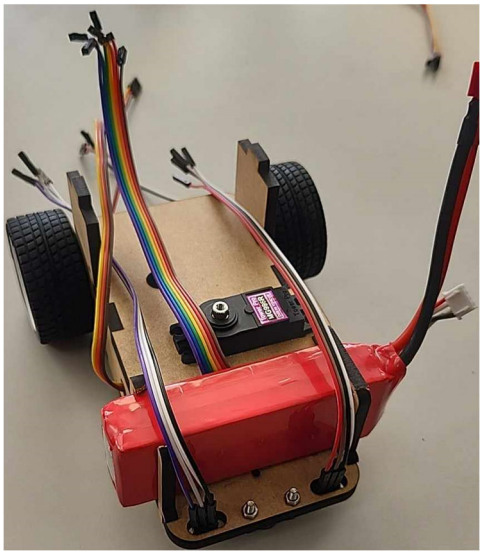


Figure 10: Cable Routing V1

The Battery mounts, left, broke quite quickly due to their thin nature. This was due to a last-minute decision in the design before laser cutting, that saw these holders be included. Before this cable ties were going to be used, however, we wanted to be able to remove the battery quickly if needed for H&S reasons.

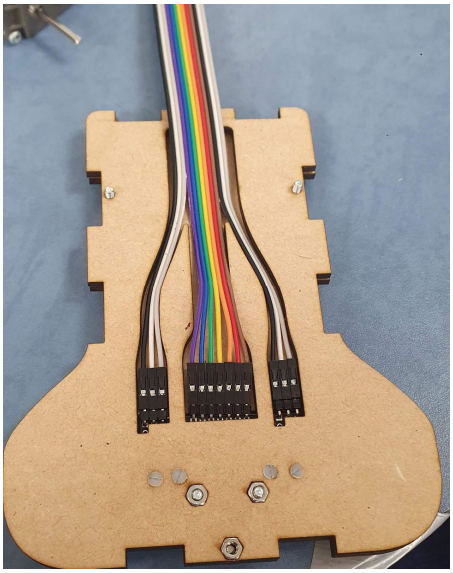


Figure 11: Cable Routing in Chassis Floor V2

To tackle the cable routing issue we decided to split the base for the chassis into 2 and make the lower one have holes for the sensors to embed into and the upper one has cable routing tracks laid out so that they wouldn't interfere with the Gearbox and servo. Slits in the top deck and PCB base were also added to encourage the wires to stay neat for as long as possible. During the assembly of these components, we intentionally used brand-new wires that were still attached to each other so that keeping them in line was easy. A hole was also added in the middle of the arm to allow for the scoop servo cable to travel into the chassis. During the the Chassis Floor redesign, the battery was laid flat which meant it would fit within the height of the main body. This allowed for a more secure hold

of the battery. The new Chassis also included wings that were slightly greater than the battery so that if the SAV were to bump into anything it wouldn't affect the power supply.

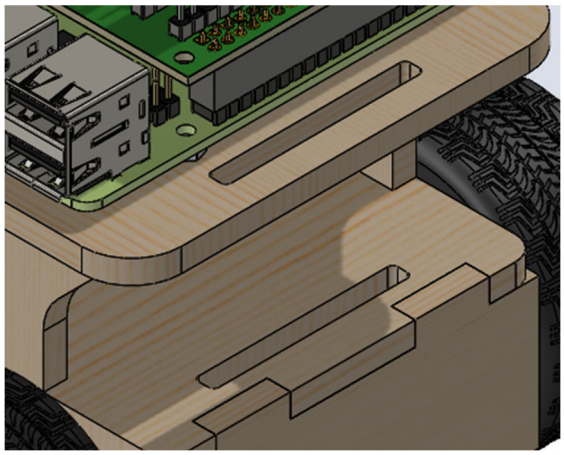


Figure 13: Render of Cable Guide slits

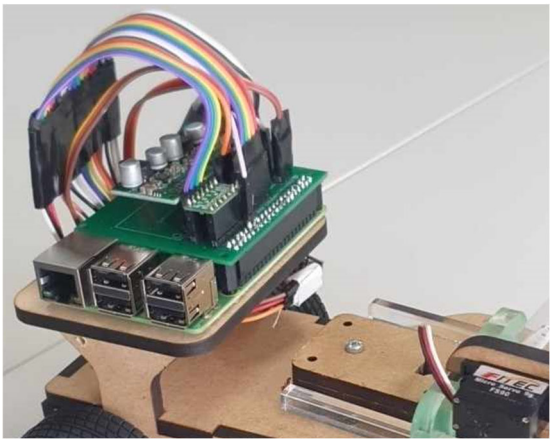


Figure 12: Heatshrinked and Improved Cable management

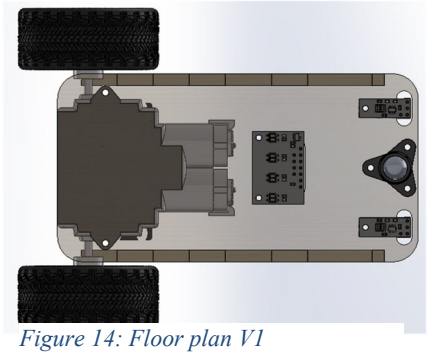


Figure 14: Floor plan V1

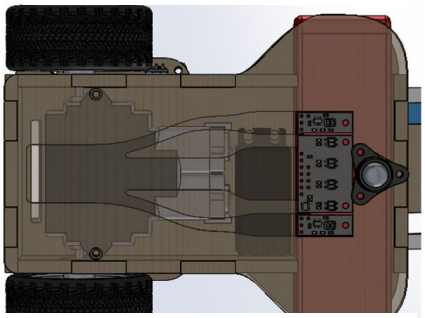


Figure 15: Floor Plan V2

Once again the lack of knowledge of how the project would pan out caused the line following sensors to be placed in such a way that would perfectly line up to see all white when detecting a fork. This proved problematic because this could also be seen as an off-track error. Stopping the car altogether. By the time we had redesigned the chassis, we had a far greater understanding of what we were trying to achieve. By lining 2 single reflectance sensors next to the 4 sensor array, we're able to create a 6 sensor array that would dramatically increase our line-following ability and general system logic.

Problems and Solutions of V2

While the placement of the distance sensor was very deliberate, as we started to increase the speed of our SAV dramatically in the latter stages of testing, we began to overshoot the pickup locations very easily. This could've been combated by making the detection earlier if the sensor had been further ahead.

While the improvements to the cable management were phenomenal, the battery cable still remained an issue and due to its bulky and stiff nature, it is strong enough to push the pickup locations out of the way causing alignment issues during pick-up and drop-off when travelling at speed.

There was also an issue where if a Minifigure was facing away from the track its feet would likely get caught under the moving edge of the scoop. This would cause the feet to then get caught outside the platform during drop off which would likely result in a failed drop off due to it not making it onto the platform. A Redesigned Scoop that had ramping curves which helped for smoother transitions which lift the scoop onto the platform would help with this. So would increase the sharpness of the moving scoop edge, which would encourage a successful pickup.

Version 3 will be after this report is complete, this will see the moving sides of the scoop get a redesign that will encourage the Minifigure to fall towards the middle after each pickup to improve the reliability of the Dropoff.

SAV Electrical Design

The electrical design for the SAV presented a challenge in that it was the first time we had to create a custom PCB. This first required learning the basic principles of PCB design. We went through numerous iterations of PCB designs before selecting the one we would progress forward with for the rest of the project. There were multiple aspects dictating what makes a good pcb, current load is a major consideration when selecting trace widths, and another major hurdle had to be overcome when routing the individual traces for each component. However, the first challenge to overcome was placing the headers for each component in an efficient place to have the shortest trace lengths possible.

Unit	Current (Amps)	Width (Mil)
Voltage regulator	4	121.89
DC motor driver	1.2	23.16
Gearbox + motors	2.20	53.44
MiniServos	0.80	13.24
Servos	1.10	20.54
4x Line followers	0.06	0.39
Line followers	0.01	0.01
Range sensors	0.03	0.14

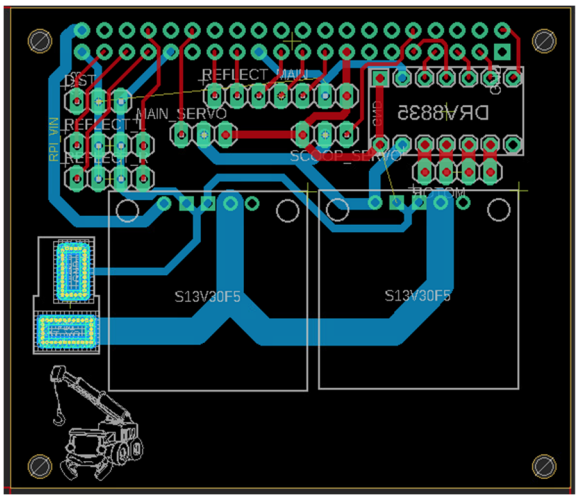


Figure 16:Final PCB Design

A source of discussion amongst PCB designers is the use of ground traces vs. ground planes. After research online it was found that each solution has benefits and drawbacks for specific use cases; ground plains are extremely useful in isolating 'noisy' signals so as not to see interference in sensitive signals however they also act as a large heatsink making soldering of components more difficult. Ground traces on the other hand are more commonly used as designers can route their grounds more precisely. For our project, it was

found that there would be no real benefit to either solution, as such we decided to go with ground traces on our final design.

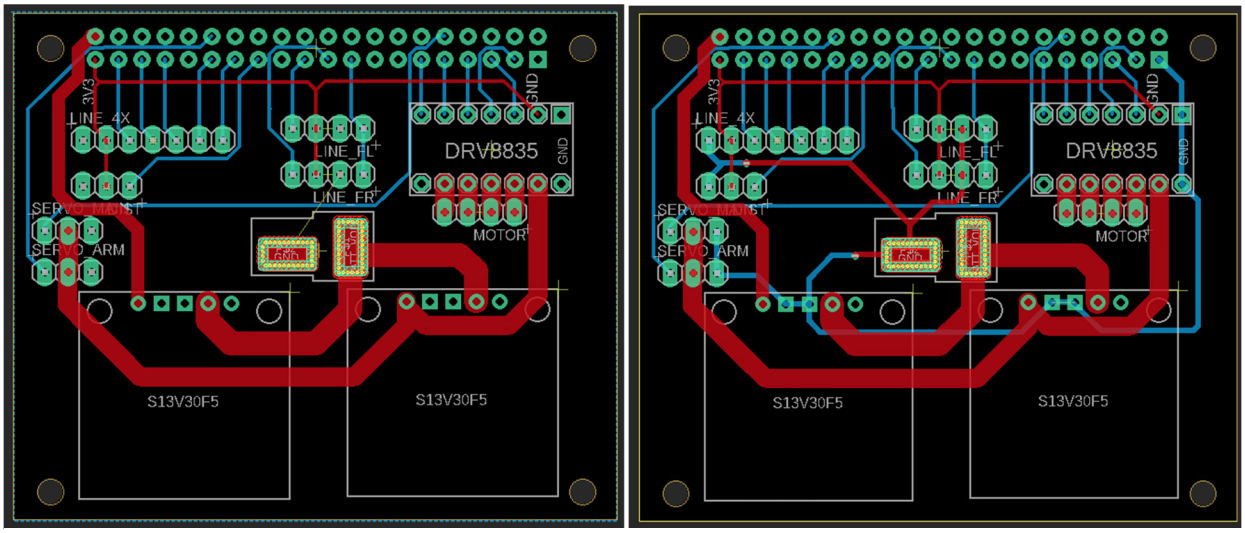


Figure 17:Two identical pcb designs one with a ground plane (left) and one with ground traces (right)

A drawback experienced with using custom PCB is the PCB's inherent limited use, that is the board was designed to be used with specific hardware in a specific configuration and is not easily adaptable when the specified hardware is no longer available due to failures and limited supplies.

A second source of challenges to our electronic design came in the form of our Raspberry Pi 3b+. An unexpected issue was encountered when it was discovered that some of the pins on the Pi were naturally high. After research failed to provide a solution it was decided that the problematic pins be replaced with new ones being soldered in place. During the design process, it became apparent that the limited number, 2, of PWM pins on the Pi was a roadblock that needed navigating.

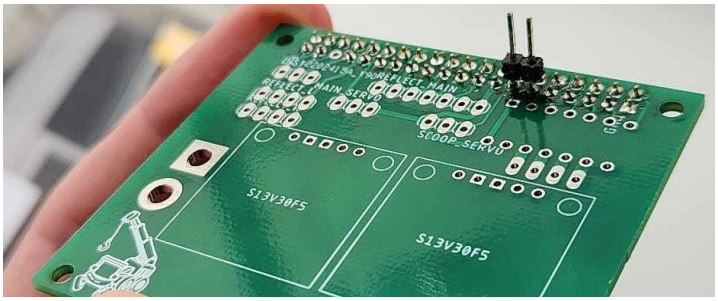


Figure 18: PCB Pinout Solution

The design of our SAV calls for 2 servos to actuate the collection mechanism in addition to the required 2 motors thus requiring 4 PWM pins to realise the design. This shortcoming of the hardware was overcome by a software solution through the utilisation of the library 'PIGPIO'. This library allows any GPIO pin to be used as PWM.

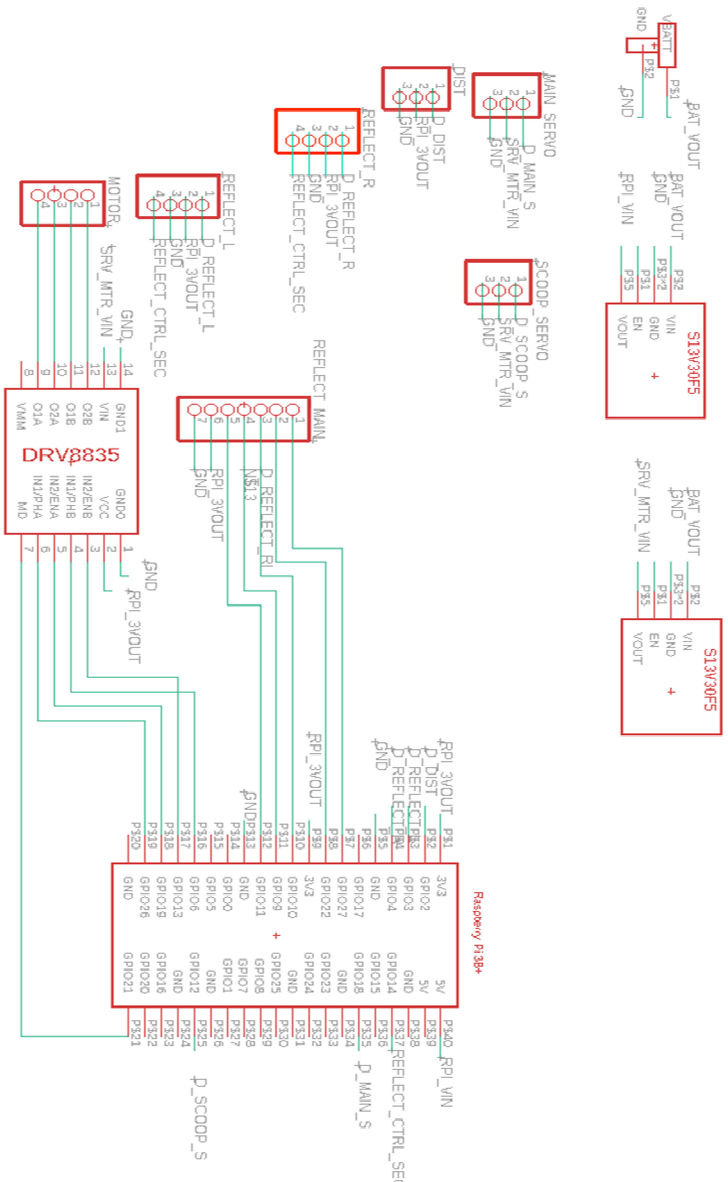


Figure 19: PCB schematic

SAV Control Logic Design

A Simple Problem

A line-follow robot is one of the simplest robotics projects anyone can make, and the control logic of it is no different. The very first step to a proper line-follow robot is just that, following the line. Once the robot follows a line well, more logic can be introduced. The simplest approach to line following is called "bang-bang" control. When the robot is too far to the left as in Figure 1, the right sensor sees white, and the robot therefore turns to the right and drives forward. The robot continues to drive until the left sensor sees white, and the robot turns to the left and drives forward, and so on.

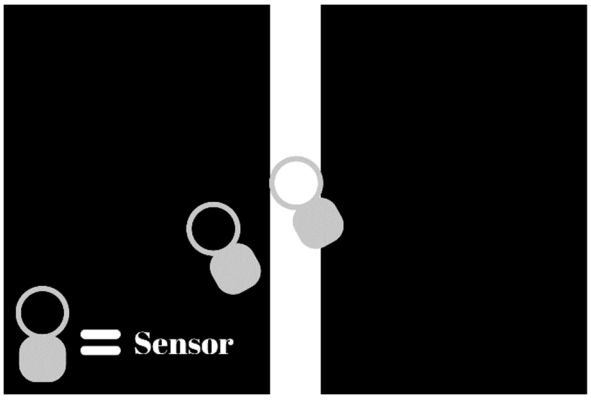


Figure 20

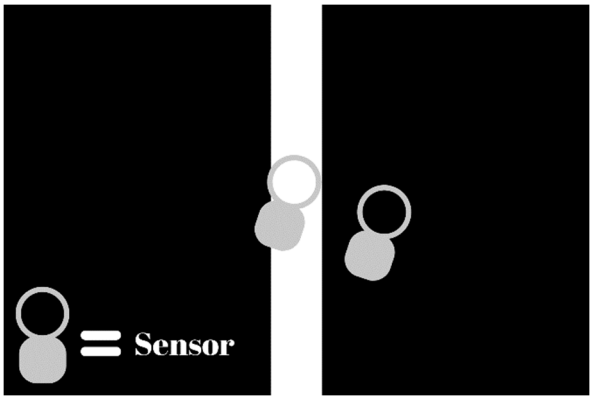


Figure 21

However, there are several problems with this control method. There can be a lot of oscillation in the robot, which affects how accurate it can be when trying to detect forks in the path. Obviously, the robot is taking a much longer route than necessary to the destination, as it oscillates back and forth.

A (Less) Simple Solution

Instead of using this “bang-bang” control method, we decided to use PID control for the line following. PID stands for Proportional, Integral, Derivative - for each of the parts that contribute to the control method. The proportional part is simple when following a line, it is simply related to how far off the goal (in this case, directly centred over the line) the robot is. The further from the goal, also called the setpoint, the greater the robot's reaction is. If the robot is far to the left of the line, it will try to the centre of the line with greater speed than if it was only slightly left of the line. This means that the robot spends less time travelling perpendicular to the line, and more time actually following the line.

The derivative part of the control method essentially dampens the proportional part. The robot compares the previous error (how far off the centre of the line it is) with the current error it has just calculated and changes the robot’s reaction accordingly. So if the previous error is less than the current error (error is increasing), the robot increases the derivative output, which in turn increases the overall output to the motors to be slightly more than it otherwise would have been. And if the error is decreasing, then the derivative output is decreased, which reduces the overall output to the motors. These work together such that if the robot is constantly overshooting because of the proportional control, the derivative control will dampen it and so keep the robot following the line closer.

The final part of the control method is the integral. The purpose of the integral is to keep track of the cumulative errors and adjust the overall output to the motor to compensate for that error. So if the robot is consistently slightly to the left of the line throughout the whole line-follow, the integral part takes that into account and adjusts the overall output to steer the robot slightly more to the right.

Combining all these together gives us the equation: Output Motor Speed = Kp * Current Error + Ki * Cumulative Error + Kd * (Error - Previous Error), where Kp, Ki, and Kd are some constant multipliers that are found by tuning the whole system. This Output Motor Speed is then applied to the motors as follows: Right Motor Speed = Base Motor Speed + Output Motor Speed, and

Left Motor Speed = Base Motor Speed - Output Motor Speed, where Base Motor Speed is some speed chosen by the user that will determine the overall speed of the robot as it follows the line.

Logic and Directions

With the line following implemented as described above, the next step is to enable the robot, or SAV (Semi-Autonomous Vehicle) to detect intersections (specifically in this case, forks) and take the appropriate directions at those forks. Detecting the forks is accomplished by checking if the line sensor output matches any of the 3 possible fork detection scenarios - centre detection (figure 21), slightly left (figure 22), and slightly right (figure 23). Detecting if we are off the track is accomplished by sensing if the SAV is completely off the track, i.e. all sensors returning white. Determining if the SAV is at a pick-up or drop-off zone is achieved by a simple combination of reading the range sensor, and flags and timers in the code that reduces false detections. A similar strategy is used for detecting the finish zone.

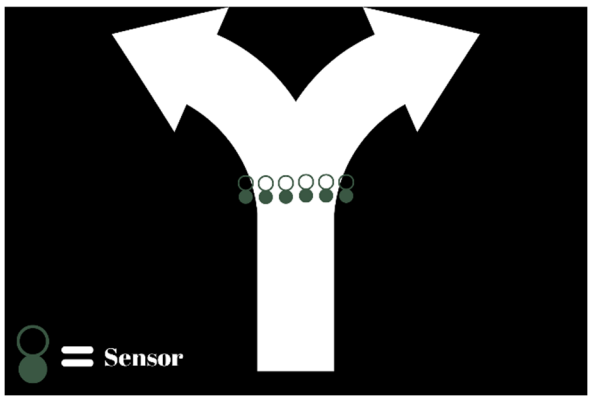


Figure 21

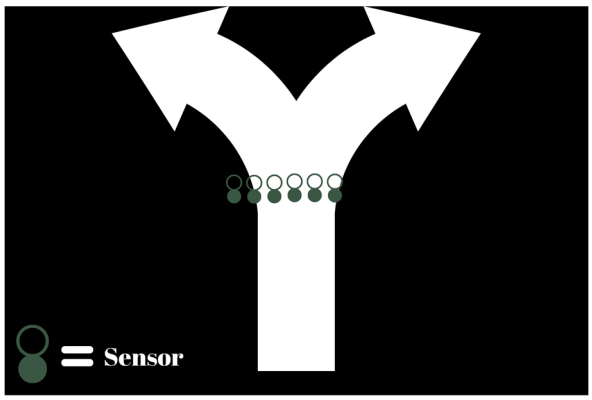


Figure 22

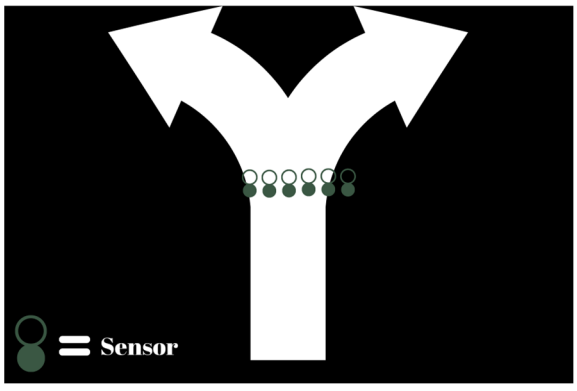


Figure 22

In order for the SAV to take the correct directions at the fork, the camera tower must communicate the directions needed before the SAV gets to the fork. The initial directions, left or right at the first fork and pick up left or right, are communicated before the SAV starts its run. The final set of directions, right or left at the second fork and drop off right or left, are communicated when the range sensor triggers as the SAV passes the camera tower. So, combining all these together gives us the following diagram that summarises the SAV control logic.

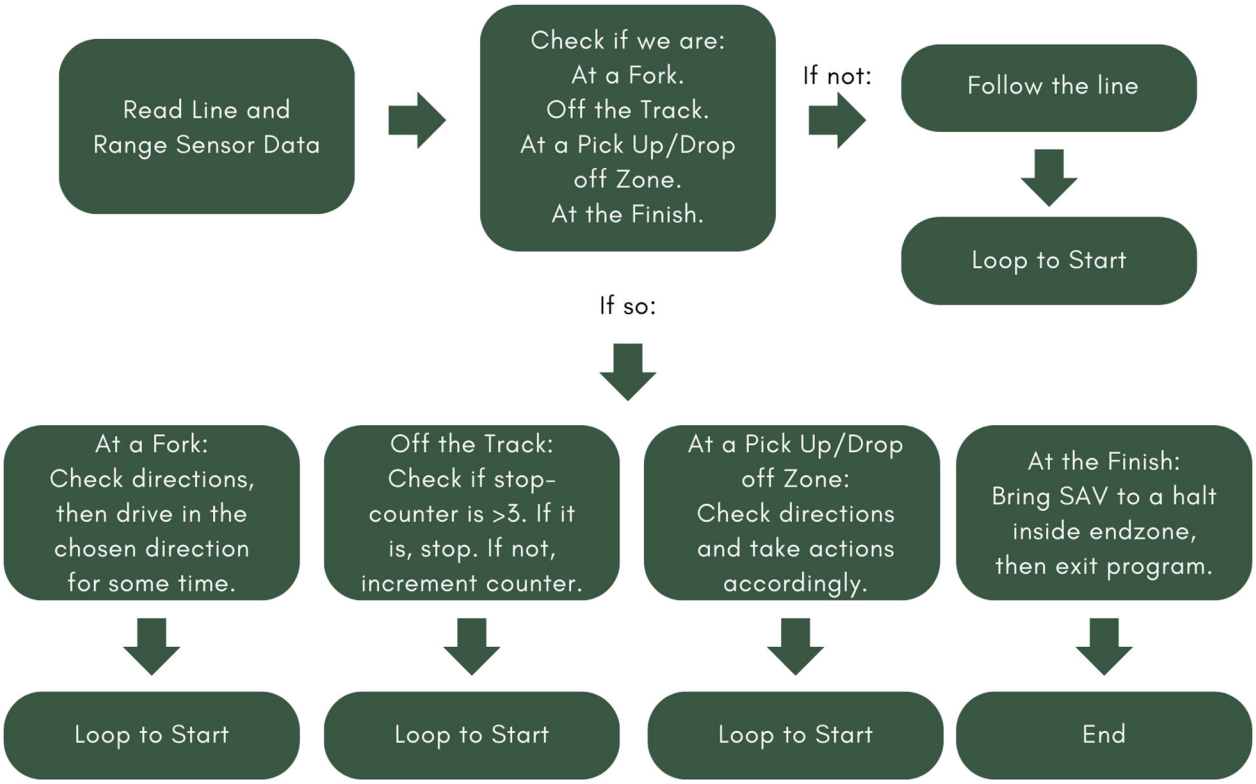


Figure 24

Camera Tower System Design

The Goal

The purpose of the camera tower is to provide the SAV with the directions necessary to complete the desired route. The system needs to be able to cope with dynamically updating the directions as the drop-off zones get put out after the SAV has already started. This means the tower needs to send the correct pick-up information to the SAV while there are only 4 zones on the map, and then update the SAV with the drop-off information after the next 4 zones are put out. This must all be done autonomously, with the user only adjusting mask values before the race starts, and inputting the colour of the pick up and drop off zones.

The Hardware

All teams get to use the same camera tower hardware, a Raspberry Pi 3B+ with a Pi Camera mounted on a raised MDF tower, looking down over the track, as shown in Figure 7.

The Software

The main library the camera tower software uses is OpenCV for Python. The program itself is very simple. First, it reads a frame from the Pi camera and processes it into the HSV colour space, because HSV allows for masking for colours a lot easier than the RGB colour space. Now that the image is in the HSV colour space, the program auto-crops the image down to just include the track and zones. This speeds up image processing and also eliminates more possible false

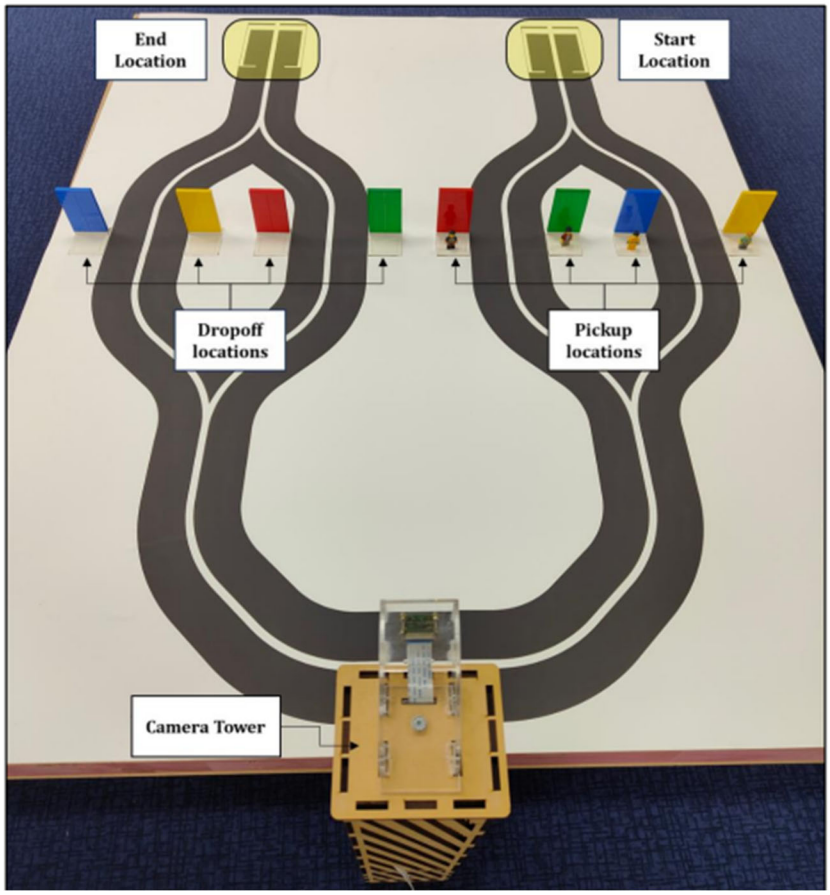


Figure 25

Positive identifications of zones from background noise. Then the cropped image is masked for each of the approximate hue, saturation, and values of the four colours of zones. These masks are first bitwise OR-ed together to make the “zoneMask” as seen in Figure 26. Using "erode” and “dilate” this mask then has the noise reduced, producing the “cleanZonesMask”. This mask is then used to determine the X coordinates of all the zones.

The mask of the chosen colour (mask) also has this noise reduction applied to it producing the “cleanMask”. From this and the X coordinates found previously, the location and therefore

directions can be determined. Figure 25 shows the UI with all 8 zones out, however, everything still works in much the same way when there are only the 4 pickup zones out.

The final part of the camera tower system is the sliders seen on the right in Figure 25. These are used to dynamically update the mid-points of the hue, saturation, and value of all the masks. By adjusting these values after the program is already started it can be tuned for the specific lighting conditions present. After determining the directions, they are sent to the SAV by using an MQTT broker, server, and client.

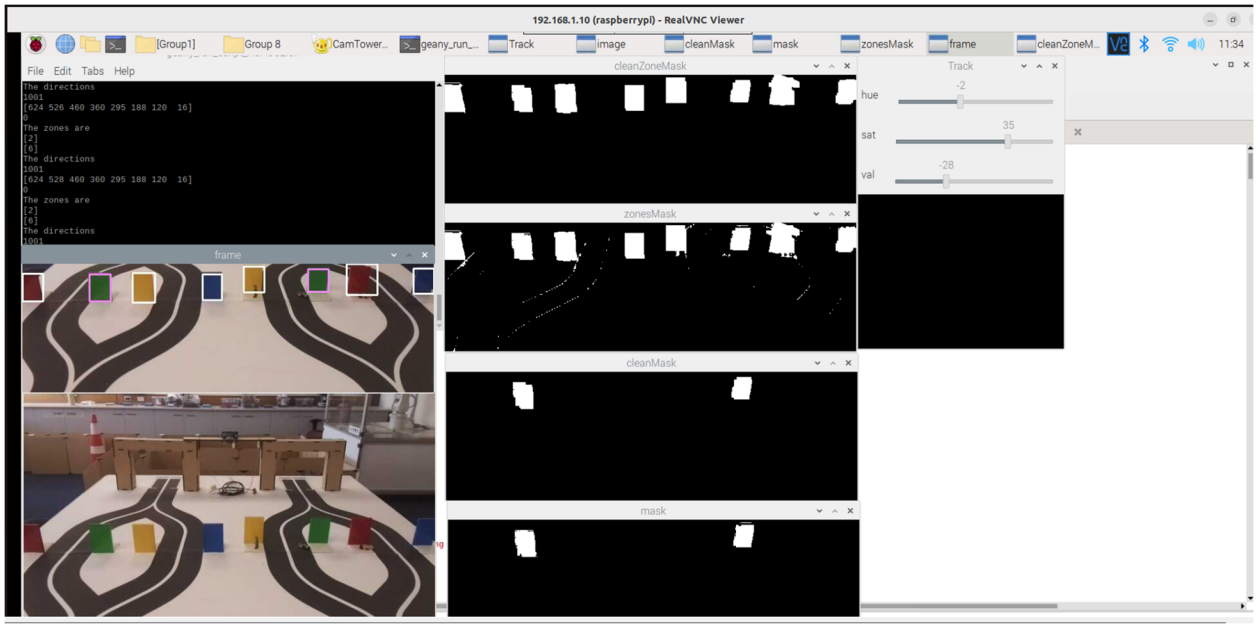


Figure 26

Vision Mātauranga

Mātauranga Māori translated literally, means ‘Māori Knowledge’. Through our education as culturally competent engineers, we know that a more accurate description of its meaning includes traditions, world views, philosophies, values and understandings that are uniquely Māori. This is a concept that is as applicable to the cutting-edge engineering of an SAV as ethical farming practices. Vision Mātauranga states that we as professionals must consider the Māori perspective and values in all that we do and create.

This means aligning with and supporting the Māori view point and Te Pai Tawhiti. This project aligns well with the hoe of Wakatū incorporated’s Hoea Ki Te Pae Tawhiti.

Ngāhau Hihiko states that we bring strength, energy and courage to all our endeavours as we strive for a better tomorrow.

Where Pūtea and Taiao also state that we must employ environmentally friendly practices and uphold the highest standards of safety as to protect the nurturing life sustaining forces of our land, water and sea whilst protecting and enabling whānau to achieve their aspirations.

By upholding this vision we respect the Mana of ourselves the University and the Iwi that owns the land on which we live and learn.

Results and Conclusions

The SAV performed exceptionally well at the given task. After a couple of design revisions following real-world testing, the SAV can consistently complete the course and tasks in under 20 seconds. The SAV logic code runs at over 100Hz. Reading the line sensors, calculating the position of the SAV over the line, checking for special conditions (e.g. intersection or drop-off zone), and then taking the appropriate action all takes less than 10ms. Achieving this speed is mainly due to the way we parallelized our line sensor reading code, taking our average code cycle down from around 80ms to 10ms.

The scoop design proved to be very reliable in collecting the minifigures from the pick-up zones no matter their position in the zone. One of the scoop's best attributes was its limited use of small delicate laser-cut components and thus the SAV was fairly robust. This made assembly and rigorous testing incredibly easy and definitely attributed to our success in completing the task early.

In conclusion, we have developed a small-scale model of the smart city system and created a smart, safe and efficient autonomous vehicle that can reliably collect its passengers and navigate the approximation of the urban environment to deliver them to their desired location in under 20 seconds.

Future works: Moving forward we aim to continue to refine our PID line following capabilities for more accurate line following in an effort to increase our travel speed. To achieve these greater speeds we may face stability issues due to limitations of the three-wheel design. To overcome this we will investigate; a ground-effect aerodynamics design or, more likely due to design constraints, the addition of active ground-effect stimulating technologies. This may include aerodynamic channelling on the ground plate of the vehicle with the support of an aerodynamic skirt in addition to some motor/fan to actively generate a suction force. These additions will enable us to attain the highest possible transport speeds for our autonomous taxi hence increasing its efficiency and desirability.