The University of Akron

IdeaExchange@UAkron

Williams Honors College, Honors Research Projects

The Dr. Gary B. and Pamela S. Williams Honors College

Spring 2021

Ping Pong Ball Collecting Robot

Gina Lanese
The University of Akron, gml22@zips.uakron.edu

Jason Colonna
The University of Akron, jnc66@uakron.edu

Sarah Kuchcinski
The University of Akron, sek106@uakron.edu

JohnDavid Rogers
The University of Akron, jsr77@uakron.edu

Follow this and additional works at: https://ideaexchange.uakron.edu/honors_research_projects

Part of the Robotics Commons

Please take a moment to share how this work helps you through this survey. Your feedback will be important as we plan further development of our repository.

Recommended Citation

Lanese, Gina; Colonna, Jason; Kuchcinski, Sarah; and Rogers, JohnDavid, "Ping Pong Ball Collecting Robot" (2021). *Williams Honors College, Honors Research Projects*. 1370.
https://ideaexchange.uakron.edu/honors_research_projects/1370

This Dissertation/Thesis is brought to you for free and open access by The Dr. Gary B. and Pamela S. Williams Honors College at IdeaExchange@UAkron, the institutional repository of The University of Akron in Akron, Ohio, USA. It has been accepted for inclusion in Williams Honors College, Honors Research Projects by an authorized administrator of IdeaExchange@UAkron. For more information, please contact mjon@uakron.edu, uapress@uakron.edu.

Project Design Report

Design Project: Ping Pong Ball Collecting Robot

Design Team 02

Jason Colonna
Sarah Kuchcinski
Gina Lanese
JohnDavid Rogers

Dr. Jay L. Adams

11/25/2020

Table of contents 1

List of figures 2

List of tables 2

1.1 Need 3
1.2 Objective 3
1.3 Background 3
1.4 Marketing Requirements 8

2. Engineering Analysis 9

2.1 Circuits 9
2.2 Electronics 10
2.3 Image Processing 11
2.4 Communications 12
2.5 Electromechanics 13
2.6 Embedded System 17

3. Engineering Requirements Specification 17

4. Engineering Standards Specification 19

5. Accepted Technical Design 20

5.1 Hardware Design 20
Level 0 Block Diagram 20
Level 1 Block Diagram 20
Level 2 Block Diagram - Main Control Board 22
Level 2 Block Diagram - Ball Lifting Mechanism 25
Level 2 Block Diagram - Camera Vision 26
Level 2 Block Diagram - Power System 28
5.2 Software Design 29
Level 0 Software Flowchart 29
Level 1 - Initialization Flowchart 29
Level 2 - Scanning Algorithm Flowchart 32
Level 3 - Image Analysis Flowchart 34

6. Mechanical Sketch 35

7. Design Team Information 36

8. Project Schedules 37

9. Conclusion 38

10. References 39

List of figures

Figure 1: Ball stack approximation 13
Figure 2: Dimension approximation of the vertical spiral lift 15
Figure 3: X-Y Dimension approximation of the interior of the entire robot 16
Figure 4: Level 0 block diagram 20
Figure 5: Level 1 block diagram 21
Figure 6: Level 2 block diagram for the main control board 22
Figure 7: Level 2 block diagram for the ball collection mechanism 24
Figure 8: Level 2 block diagram for the ball lifting mechanism 25
Figure 9: Level 2 block diagram for camera vision 26
Figure 10: level 2 block diagram for the power system 28
Figure 11: Level 0 software flowchart 29
Figure 12: Level 1 initialization flowchart 29
Figure 13: Level 1 driving algorithm flowchart 30
Figure 14: Boustrophedon path and arena/table dimensions 32
Figure 15: Level 2 scanning algorithm flowchart 33
Figure 16: Level 3 Image Analysis Flowchart 34
Figure 17. Mechanical sketch of entire robot system 35
Figure 18: Gantt Chart for project 37

List of tables

Table 1: Current estimates for the main control board 9
Table 2: Current and power estimates for the robot 10
Table 3: Engineering Requirements Specification 18
Table 4: Engineering Standards Specification Table 19
Table 5: Functional Requirement Table for Pong Bot 20
Table 6: Functional Requirement Table for the Level 1 Block Diagram 22
Table 7: Functional Requirement Table for the Main Control Board 23
Table 8: Functional Requirement Table for the Vertical Spiral Lift 25
Table 9: Functional requirements table for the Vertical Spiral Lift 26
Table 10: Functional requirements table for Camera Vision 27
Table 11: Functional requirements table for the Power System 28

1. Problem Statement

1.1 Need

Authors: JC, SK, GL, JR

Where ping pong is played competitively, the problem of having to pick up many balls at once occurs frequently. At the end of a lesson or practice session, the players might find themselves with upwards of one or two hundred balls on the floor surrounding them. Currently, there is not a good system for picking up these balls. Many people and facilities use a fish net to pick up approximately a dozen balls at a time, or, if there are only a few dozen balls on the ground, they may pick them up by hand. This is a waste of time, and only further fatigues the player after an already tiresome practice or game session.

1.2 Objective

Authors: JC, SK, GL, JR

The objective of this project is to design, build, and test a robot that will collect stray ping pong balls. This robot will be small and lightweight. It will be wireless and operate autonomously to detect, pick up, and return stray ping pong balls.

1.3 Background

Authors: SK, JC

A critical part of any robotic system is the control algorithms programmed in its software that permit the robot to act autonomously. These algorithms determine how the robot travels through its environment, responds to sensor input from the environment, and acts upon its environment. An example of these control algorithms is seen in the patents for the Roomba autonomous vacuum cleaner (Jones & Philip, 2001). In addition, the system employed by researchers at the Northwestern Polytechnical University in their ball collecting robot will be explored. The Roomba is designed to have several methods to effectively cover the designated area to clean; these methods consist of a spot coverage mode, an obstacle following mode, and a bounce mode. In contrast, the researchers at Northwestern Polytechnical University used the A* algorithm.

In the Roomba spot coverage mode, the robot moves in an outward spiral pattern to cover a particular area. In the obstacle following mode, the robot again moves in a spiral pattern until it encounters an obstacle or travels a certain distance, in which case it travels straight until it encounters an obstacle. Once an obstacle is detected, it runs along the obstacle twice the length of the robot, then turns away from the obstacle and travels forward. In bounce mode, the robot travels forward until it encounters an obstacle, wherein it turns and travels in a random direction away from the obstacle.

Each of these modes could be used in the proposed system, depending on the desired behavior from the user and the location of ping pong balls needing to be cleaned up. One limitation of the Roomba algorithms, as applied to the proposed system, is that the Roomba is designed to travel over the entire designated area in order to clean it. The proposed system does not need to do this: it only needs to travel to the locations where there are ping pong balls to pick up. Therefore, the Roomba algorithms could be improved upon in the proposed system by performing detecting the location of the balls before moving.

After considering a series of path finding algorithms, the researchers at the Northwestern Polytechnical University employed the A* algorithm based on its computational efficiency and its ability to provide the most optimal path. Using the Dynamic Window approach (DWA), the robot's velocity and angular velocity are sampled, and the algorithm simulates multiple trajectories for the robot. Then, based on a number of factors such as obstacles, target location, and speed, the path with the most desired score is used (Nie et al., 2019). However, some of the main drawbacks to the A* algorithm is it is not the quickest algorithm and it requires a lot of storage since it saves all the possible paths. But based on the benefits listed, it is a viable option.

The proposed system will require a method of detecting obstacles. Several methods can be used to perform this function. The Roomba uses tactile bump sensors to determine when it has encountered an obstacle. A downside to this method is that the robot must physically encounter the obstacle in order to detect it. A better method would be to use ultrasonic sensors, as investigated in Jung et al. (2007). This method of detection consists of a pair of hardware circuits, one for transmitting and the other for receiving. The distance between an obstacle and the sensor is determined by measuring time-of-flight of the ultrasound that is emitted and received. If more sensors are used, the direction of the obstacle can be determined as well. These sensors can be easily and cheaply found as off-the-shelf components.

Author: GL

The robot must have a way to find ping pong balls in order to navigate to them and pick them up, and this is where computer vision and object detection will come into play. The ping pong ball collecting robot will need to accurately detect one type of object, a ping pong ball, but will only need to do this with relatively low precision. There are two general categories for object detection, region based methods (which have a higher runtime) and single-shot detectors (Szemenyei & Estivill-Castro, 2019). These methods use an “anchor box” to place a box around objects to signify that it has been found and identified. The ping pong ball collecting robot will only need to detect a general area so that it may navigate there and scoop up the ball, so a system which can detect a ball with decent accuracy but low precision is desirable, as less computational power can be directed towards precision.

There are many deep neural network architectures that already exist, with recent research favoring accuracy over size. Many of these are too large and require too much memory to run on an embedded system. SqueezeNet, a fairly new architecture, requires 50x fewer parameters but is just as accurate as AlexNet while being only 0.5 MB in size while being just as accurate as AlexNet (which requires a GPU and cannot run on an embedded system), making SqueezeNet an excellent architecture for an embedded system. The difference in size is obtained through replacing a majority of 3x3 filters with 1x1 filters (nine times fewer parameters), decreasing the number of input channels to the remaining 3x3 filters, and downsampling late in the network to maximize accuracy with fewer parameters (Iandola et al., 2016). Another promising architecture is TinyYOLO (You Only Look Once). TinyYOLO has a similar size when compared to SqueezeNet, so testing may have to be done to determine which architecture will be best for the ping pong ball collecting robot (Szemenyei & Estivill-Castro, 2019).

Authors: JR and JC

The premise of movement for a robot comes down to that of locomotion. There are several mechanical systems that can be used - simple or complicated, cheap or expensive, all dependent on the task at hand. To reiterate, the goal of this robot is to retrieve ping pong balls, maximizing speed (and thus efficiency) and ball quantity. For example, a quadruped or hexapod robot that achieves various forms of locomotion using a spiking neural network (SNN) may be possible, but the application is far beyond that of simply collecting balls (Espinal, et al., 2016).

Thus, a simpler means of locomotion is what may be required in order to be as efficient as possible. Simple wheels powered by motors that can be independently controlled would be far more efficient, especially when factoring in autonomy.

As for the locomotion of the robot, a system employed by researchers at the Northwestern Polytechnical University could be employed. This system consisted of a differential driving mechanism with two actuated wheels and one omni-directional wheel. The three wheeled design gives the robot flexibility in movement to be able to quickly change direction and orientation on the spot. In order to power this drivetrain, the researchers used two brushless DC motors to actuate the wheels (Nie et al., 2019). This technique could be applied as a means of locomotion as it offers simplicity to the design.

As for the ball pickup mechanism, there are many routes that can be taken. At Loras college, students made a similar robot that could traverse a specified terrain to collect ping pong balls. This system involved a vacuum system. Once the user sent the “pick up” command to the robot, a fan would activate creating a vacuum system. Essentially, the ball would be sucked up into the robot and stored in its holding container (Dirksen et al., n.d.).

Other ideas might involve some sort of spinning or rotating device - similar to how a vacuum cleaner works on a carpet. This system would involve bristles attached to a spinning rod constantly turning as long as the robot is powered on, or when commanded to do so by the user or the software. The mechanical function would be the same as a vacuum cleaner, but the bristles themselves would be made out of a different material to allow for the ball to pass through the mechanism, into the holding container. Once the robot detects a ping pong ball, it simply drives into it, and the spinning bristles on the rod force the ball into a collection bin.

A patent for a similar robot shows a mechanism similar to the bristle idea previously mentioned. However, instead of having bristles, it has three plates that rotate, similar to a rotating door but turned onto its side. It rotates as the robot drives forward, and the robot drives into what it recognizes to be a ping pong ball (Tsai, 2010).

A final idea for the pickup mechanism would be some kind of plunger device. This device would be essentially a tube that would position itself over the ball and retract in some sort of way that would 'suck' up the ball, one at a time. This pickup mechanism would maximize ball pickup speed, and it would depend on the locomotion of the robot to determine overall speed.

In summary, the best form of pickup would be the spinning bristles/plate idea, so long as the bristles/plate are designed to easily pick up and pass the ping pong ball through into the ball holding container. The ideal form of locomotion would be a simple wheel design. The front wheels of this design could turn, but also the robot could achieve turning in place by applying different torque/rotation direction to each wheel individually.

1.4 Marketing Requirements

Authors: JC, SK, GL, JR

1) System can navigate autonomously around the room and avoid obstacles
2) System will be able to identify and pick up only ping pong balls, differentiating between these and other objects.
3) System will be able to collect multiple balls at once, in a single run.
4) System will employ pathfinding strategies to scan the area efficiently.
5) System will be able to operate for 60 minutes without requiring recharge.

2. Engineering Analysis

2.1 Circuits

Authors: SK, GL

In order for the robot to function properly, all components must be supplied with the proper current and voltage. The Raspberry Pi 4 operates on 5V, and it is recommended to supply it with 2.5A. The motors for locomotion will run on 12V, and each will draw approximately 2.1A. The motors for ball collection and ball lifting will require 12V and each draw approximately 0.5A. The main control board will be designed to operate on 5V, and is estimated to draw 0.31A. Therefore, the batteries will need to operate at 12V at least, and supply 12.21A. The batteries need to power the robot for 1 hour, and it is best for battery health if they are never depleted past 20% of total charge. Therefore, the batteries will need to be rated for at least 15.26 amp-hours. The calculations to determine this are shown below.

Table 1: Current estimates for the main control board.

Component	**Voltage (V)**	**Quantity**	**Current (A)**
dsPIC33	3.3	1	0.15
LEDs	3.3	8	0.02
Lidar	3.3	1	0.000009
Gyro/compass	3.3	1	0.0003
Total current (A)			0.310309

Table 2: Current and power estimates for the robot.

	Voltage (V)	Quantity	Current (A)
Locomotion motors	12	4	2.1
Ball collection motor	12	1	0.5
Ball lifting motor	12	1	0.5
Pi	5	1	2.5
Cirucit boards	5	1	0.310309
Total current (A)			12.210309
Total power (W)			126.85155

Where power is calculated by

$$P = \Sigma(V * I) = 12 * (motor\ current\ draw) + 5 * (Pi\ and\ circuit\ board\ current\ draw)$$

Battery capacity at 1 hour and max discharge of 20%:

$$12.21A * 1hr/0.8 = \text{15.26A-hr}$$

2.2 Electronics

Author: SK

The robot will require a sensor for obstacle detection so that it does not run into any obstacles in the arena as it searches for balls. This sensor should be able to detect obstacles up to 1 meter away. This distance was chosen to be over twice the estimated robot dimensions so that the robot has plenty of space to detect and navigate around any obstacles it may encounter.

The system will need a sensor to ensure that the basket is not over-filled. It is not necessary for the function of the robot that it know the exact quantity of balls collected. The sensor need only detect when the collection basket is full.

The robot will need a method of determining its orientation, so that it can navigate the map accurately. This need can be met with a gyroscope and a magnetometer. The gyroscope will provide the heading that the robot is facing, with respect to the angle it was facing when it began measuring. The magnetometer will give the robot an absolute heading at all times during system operation.

2.3 Image Processing

Authors: SK

The system will need to process images of the environment to identify the location of any balls in the local area. In order to do this, the robot will use both the color of a ping pong ball (white) and the round shape of the ball to identify probable locations of balls in the images. The lower and upper limits of the color "white" in the HSV (hue, saturation, value) color space will be defined so as to detect the expected shade of white the ping pong ball will experience in the given lighting conditions. The lower limit will be 0, 0, 100 (pure white) in the HSV color space, and the upper limit will be determined experimentally.

To identify the round shape of a ball, the edges must be found. Detecting edges in an image is most easily performed by finding regions with strong intensity or color variation (Szeliski, 2020). This can be mathematically detected by taking the local gradient vector of a contour field, whose magnitude indicates the strength of the variation, while the orientation points perpendicular to the local contour. Since taking image derivatives accentuates high frequencies, thus amplifying noise, it is necessary to smooth an image with a low-pass filter before computing the gradient. Use of a circularly symmetric smoothing filter produces a response that is independent of orientation, so the Gaussian will be used. Once the continuous gradient of the image has been calculated, the maxima of the gradient magnitude in a direction perpendicular to the edge orientation will be identified to provide the edges detected in the image. These edges will then be linked to create contours that outline the objects detected in the image.

2.4 Communications

Author: JR

PongBot will not utilize any sort of wireless communication features. During the initial brainstorming of this project, a mobile app was considered which connected to the robot wirelessly. This app would be used to send the *start* signal to the robot, telling it to start searching for ping pong balls. However, this same implementation goal can be achieved by using a start button, physically placed on the robot, coupled with an emergency stop button (e-stop) to stop the robot in case of malfunction or other issues.

While the robot will not utilize Wi-Fi or any sort of other wireless communication protocols, it will primarily use a serial communication protocol(s) to communicate between the custom circuit boards that will be designed for the robot and the sensors that will be used on the robot.

2.5 Electromechanics

Authors: JR

One of the first calculations done was that of finding the minimum volume of the ball holding basket. This was necessary because, in order to find a rough estimation of the battery and motor requirements, the size and weight of the robot was needed, which all depended on the size of the holding basket.

A ping pong ball has a radius of 20mm (r = 20mm, d = 40mm).

Balls stack as seen below:

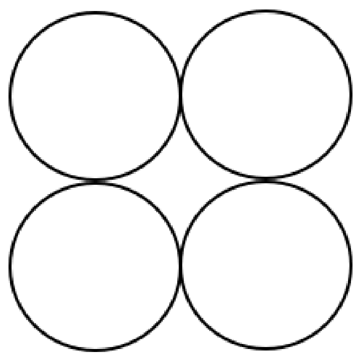

Figure 1: Ball stack approximation.

So, the volume of a ping pong ball can be considered to be the same as the volume of a cube (where w=h=l=diameter), assuming the ping pong balls stack like cubes.

So, 1 ping pong ball has the volume

$$d = w = h = l = 40\text{mm}$$

$$V_{ball} = d^3 = (40\text{mm})^3 = 64\text{ cm}^3$$

The robot will be designed to hold 100 ping pong balls, so

$$V_{ball,100} = 100 * 64 \text{ cm}^3 = 6400 \text{ cm}^3$$

The neat stacking of the ping pong balls cannot be guaranteed (left to right and top to bottom, all in line with its adjacent ping pong balls), so the design of the basket must be larger than 6400 cm^3 to accommodate for the lack of organization amongst the balls.

Choosing numbers that are easy to work with,

$$w = h = l = 20\text{cm}$$

$$V_{basket} = (20\text{cm})^3 = 8000 \text{ cm}^3$$

Estimated Robot Dimensions:

The robot's dimensions will be estimated as follows: the dimension of the basket is 20cm x 20cm x 20 cm. On the x-y axes, the length starting from the back of the robot. If on each side of the basket near a wall there is 1cm clearance, then the width of the robot becomes 22cm. The length from the back wall to the edge of the basket will be 21cm by this same logic.

From there, if the vertical spiral lift touches (or gets very close to) the basket, it will take an additional 9cm of length. This is because the center of the VSL will be a rod, of approximate diameter 1cm. If the ball needs to be held in the spiral, it must be at least the size of the ping pong ball, which is 4cm. So 4cm + 1cm + 4cm = 9cm. See figure below.

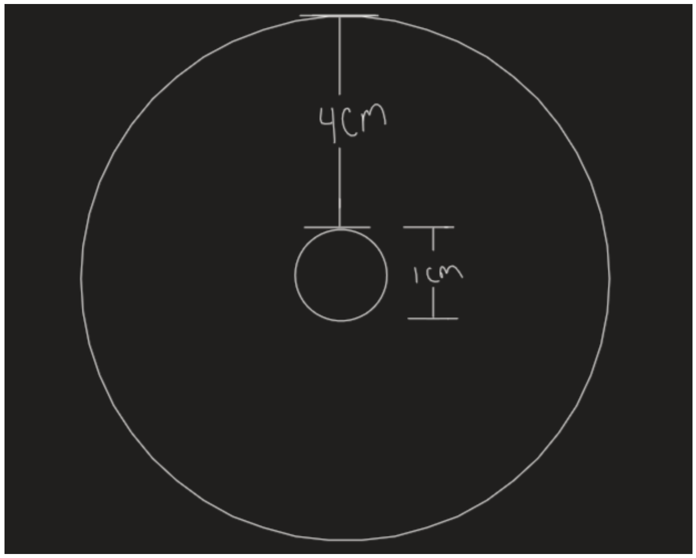

Figure 2: Dimension approximation of the vertical spiral lift.

The remaining area of the robot considers the ball buffer zone. The buffer will be designed to hold approximately 10 ping pong balls. Because the width of the robot has already been determined, it is known that approximately 5 ping pong balls can fit from side to side (because 22cm/4cm = 5 with a remainder of 2). Because of these calculations, 10 ping pong balls is an appropriate amount of balls to be temporarily held in the buffer, so the buffer needs to add an additional 9cm (8cm holds two rows of ping pong balls, plus 1cm of tolerance).
Finally, the addition of 5cm as an estimation for the length of the ball collection mechanism must be made. Thus, the final x-y dimensions of the robot are approximately 44cm x 22cm.

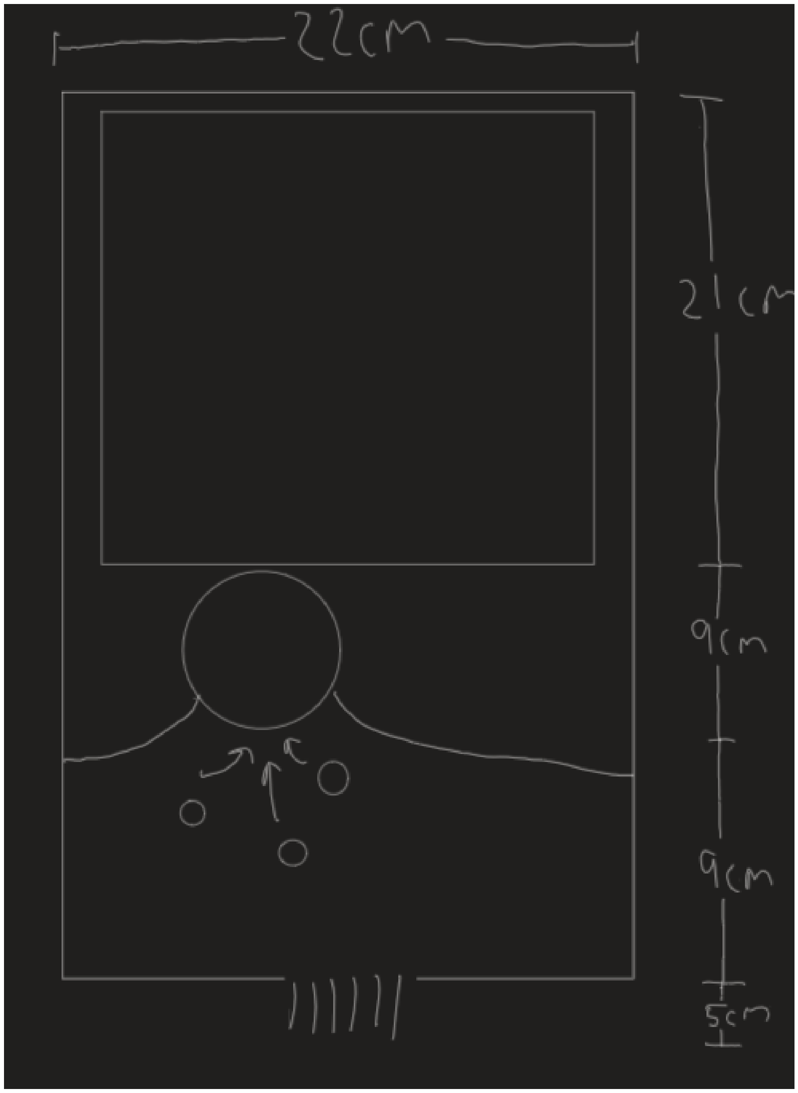

Figure 3: X-Y Dimension approximation of the interior of the entire robot.

Note that all calculations done so far have only considered the length/width from the inside wall to the inside wall. In other words, the length of the extruded aluminum that makes up the robot's frame has not been taken into account yet. Assuming the extruded aluminum is 2cm, then 2cm must be added to each wall of the robot, apart from the front wall (since measurement starts from the rear wall to the front of the ball collection mechanism, not the front wall of the robot, and the ball collection mechanism extends past the front of the frame). Thus the final x-y dimensions of the robot, including the frame, is 46cm x 26cm.

2.6 Embedded System

Authors: SK

The robot will require a main circuit board to process all sensor inputs and outputs, and run the software of the robot. This circuit board will receive the various sensor readings and output from the Raspberry Pi over a serial communication protocol. It will use the readings as inputs to the control software, and send outputs to control the robot. It will also have LEDs for the purpose of debugging and displaying status information. A dsPIC33 was chosen for the processor on this circuit board because of team member familiarity with it.

3. Engineering Requirements Specification

Authors: JC, SK, GL, JR

The following table details the engineering requirements and the marketing requirements they derive from, as well as a justification for each requirement.

Table 3: Engineering Requirement Specification

Marketing Requirement	Engineering Requirement	Justification
1, 4	Scan and navigate a user-specified area autonomously while collecting balls without any additional inputs from the user	Autonomous operation will eliminate amount of (human) time spent collecting balls and reduce fatigue of the players.
4	Use a path planning algorithm to avoid traversing the same area twice	The robot should not scan any area more than once to optimize efficiency.
3	Have the capacity to store 100 ping pong balls	It is estimated that approximately 100 balls will be used in a match or practice session.
3	Collect and store balls until the robot has traversed the area or the collection basket is full	It is most efficient if the robot can scan the area without stopping to empty the basket.
1, 4, 5	Traverse an entire room of size 12mx6m within 30 minutes	This is the standard size of a ping pong arena.
1	Avoid ping pong table and other furniture in the arena with an obstacle avoidance algorithm	The robot should not run into any obstacles.
2	Distinguish white ping pong balls from other objects in the environment	The robot should only pick up ping pong balls, not other objects.
1	Travel at a maximum velocity of 1.5 feet per second	This velocity should provide a good balance of power consumption and time.
5	Operate for 1 hour without depleting the battery charge below 20%	1 hour of operation will provide ample time for the robot to sweep an arena 1 time.

Marketing Requirements

1. System can navigate autonomously around the room and avoid obstacles
2. System will be able to identify and pick up only ping pong balls, differentiating between these and other objects.
3. System will be able to collect multiple balls at once, in a single run.
4. System will employ pathfinding strategies to scan the area efficiently.
5. System will be able to operate for 60 minutes without requiring recharge.

4. Engineering Standards Specification

Authors: JC, SK, GL, JR

In order for this design project to interface with pre-existing parts, it is essential that engineering standards are followed during the design process. The following table lists the standards that are under consideration for use in this project.

Table 4: Engineering Standards Specification Table

	Standard	**Use**
Communications	MIPI CSI-2 USB SPI, CAN, UART, I2C	-Camera to Raspberry Pi -Raspberry Pi to main control board -Main control board to other boards
Data Formats	JPEG	-images
Programming Languages	C Python	-Main control board software -Computer vision
Connector Standards	Ribbon connector	Camera to Raspberry Pi

5. Accepted Technical Design

5.1 Hardware Design

Authors: JC, SK, GL, JR

Level 0 Block Diagram

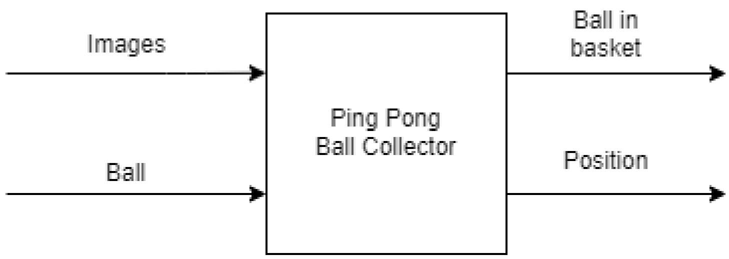

Figure 4: Level 0 Block Diagram

For the Level 0 Block Diagram, the Ping Pong Ball Collector takes in images from the environment and uses that information to drive to a certain location in the arena, in order to pick up the ball and put it in the ball collector.

Table 5: Functional Requirement Table for Pong Bot

Module	Pong Bot
Inputs	- Images of environments - Ball
Outputs	- Ball in collector - Location
Functionality	Uses the images from the environment to locate and pick up ping pong balls.

Level 1 Block Diagram

Authors: JC, SK, GL, JR

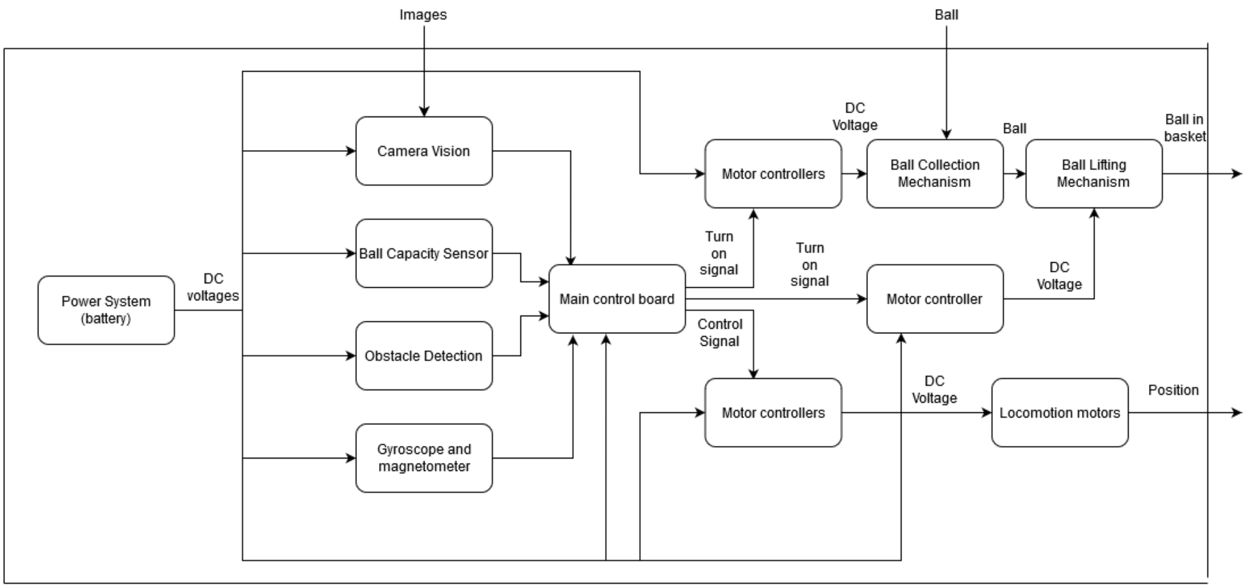

Figure 5: Level 1 Block Diagram

As shown in the Level 1 Block Diagram, power is supplied to different sections of the system, which includes the camera system, ball capacity sensor, obstacle detection, gyroscope and magnetometer, and the different motor controllers. From there, the main control board takes in the signals from the different sensors and outputs a control signal and turn-on signal to the motor controllers to tell the motors how much and how fast they should rotate. One of the inputs the system heavily relies on is the input from the environment which is gathered by the camera in the camera vision block. Once the robot detects a ball, the main control board will process the signals from the sensors and relay that information to the motor controllers to tell where locomotion motors where to go. The main control board also sends a turn-on signal to the motor controllers which control the ball collection mechanism and the ball lifting mechanism to pick up the balls from the environment and transfer them into the collection bin.

Table 6: Functional Requirement Table for the Level 1 Block Diagram

Module	Level 1 Block Diagram
Inputs	- Images of environments - Ball
Outputs	- Ball in collector - Location
Functionality	The robot uses the images from the environment to locate and collect ping pong balls.

Level 2 Block Diagram - Main Control Board

Author: GL

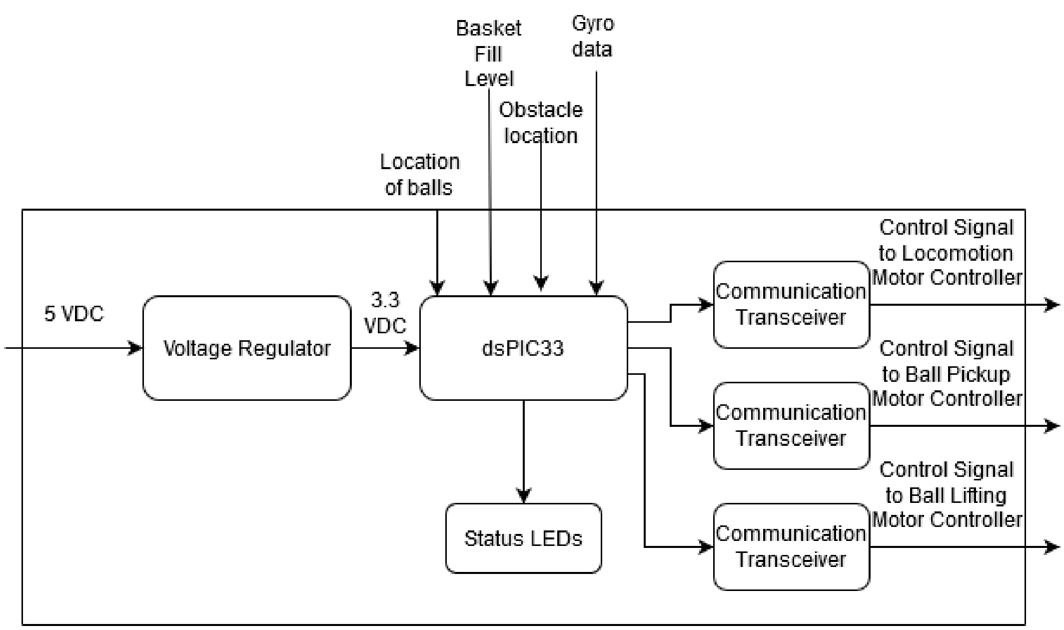

Figure 6: Level 2 Block Diagram for the Main Control Board

The main control board is a circuit board which will handle all signals and send the info to the appropriate motor controller. An input of 5VDC is sent into a voltage regulator on the board to step down the 5V to 3.3V for the dsPIC33, as the dsPIC33 operates on 3.3V. Sensor data that is sent to the main controller includes location of balls and location of obstacles from the computer vision component as well as any other obstacle sensing modules, the fill level of the basket so that the robot does not overfill its collection basket, and gyroscope data so the robot knows what orientation it is facing. Outputs include status LEDs for debugging purposes and any other status information, as well as three outputs to communication transceivers to send a control signal to the respective motor controllers. One of these control signals goes to the locomotion motor controller, which is the motor controller in charge of the driving, stopping, and turning of the robot. Another control signal outputs to the ball pickup motor controller, which will engage or disengage the ball pickup mechanism as needed based on proximity of the nearest ball(s). The final control signal output from this block is the control signal for the ball lifting motor controller, which will be engaged and disengaged based on the presence of a ping pong ball or ping pong balls.

Table 7: Functional Requirement Table for the Main Control Board

Module	Main Control Board
Inputs	- 5V DC - Location of balls - Basket fill level - Obstacle location - Gyro data
Outputs	- Control signal to locomotion motor controller - Control signal to ball pickup motor controller - Control signal to ball lifting motor controller
Functionality	Handles sensor data and sends either “on” or “off” to the relevant motor controller. Incoming data gives information about location of the robot, location of the next spot the robot must drive to, whether the collection basket has reached capacity, and whether it is time to turn on or off the ball pickup and ball lifting motors based on proximity of a ping pong ball.

Level 2 Block Diagram - Ball Collection Mechanism

Author: JR

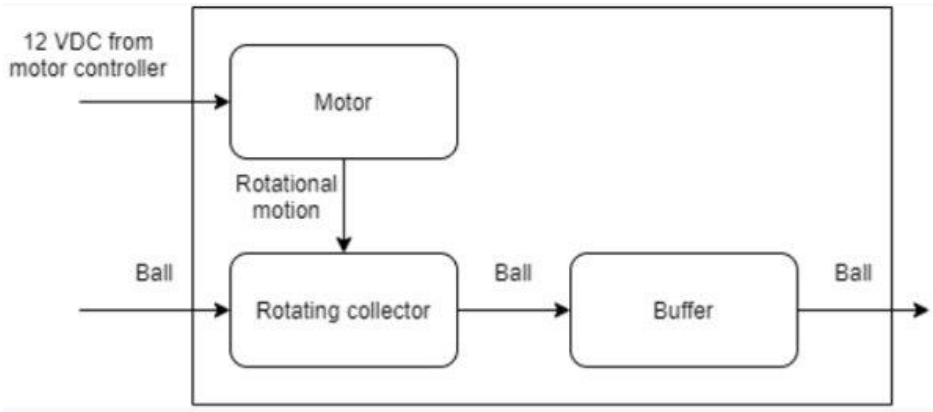

Figure 7: Level 2 block diagram for the ball collection mechanism

The ball collection mechanism is the electromechanical mechanism in which the ball is collected from the floor into the robot. This mechanism will have a rotating mechanism similar to that found on a vacuum cleaner: a horizontal rod rotates, and attached to it will be a soft, flexible material, similar to the bristles on a vacuum. The original plan was to use a hard plate instead of using a soft flexible "bristle"; the entire system could be imagined as a revolving door, but turned sideways. The problem with this design is that the hard, inflexible surface might catch the ball from the top, or any other unexpected angle. If this happened, either the ball or robot could be damaged, or more realistically, the ball would shoot off in an unexpected direction, and fail to be collected. Because of this, the use of a zip tie, or a similar flexible bristle-like material seems most suitable. From there, the ball will enter a small buffer that is on a slight incline, so that balls are forced by gravity to enter the vertical spiral lift, described in the next block diagram. This buffer allows for the robot to hold balls even while the vertical spiral lift is full, in the event that the robot collects many balls at once.

Table 8: Functional Requirement Table for the Vertical Spiral Lift.

Module	Vertical Spiral Lift
Inputs	- Power: 12 V DC - Ping pong ball
Outputs	- Ping pong ball
Functionality	Uses a horizontal bar that rotates on its longitudinal axis with a bristle-like surface fixture that directly comes into contact with the ball. The bristle-like fixture will be flexible enough such that if the ball comes in contact with any part of the bristle, it will not only refrain from damaging both the mechanism and the ball, but it also will successfully collect the ball.

Level 2 Block Diagram - Ball Lifting Mechanism

Author: JR

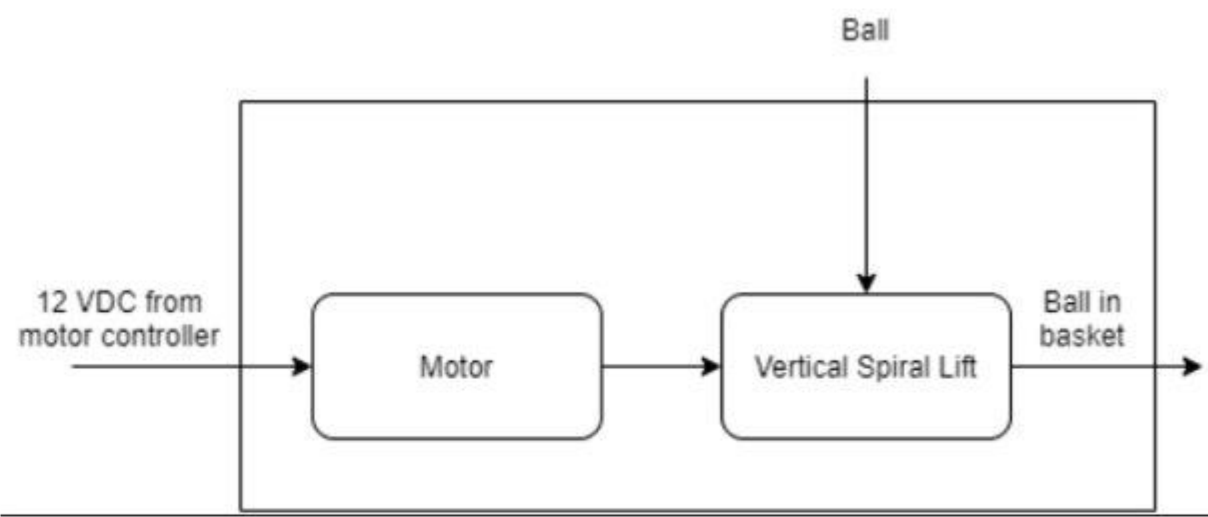

Figure 8: Level 2 block diagram for the ball lifting mechanism

The vertical spiral lift (VSL) uses a spiral design, inspired by an auger. The balls must be deposited into the collection basket from the top, because filling the balls from the bottom would mean as the basket fills up, the balls would have to be pushed upwards. This would lead to a more complicated design, and would also take more energy as more balls were collected.

The way the VSL will be designed will allow the ball(s) to be trapped against a wall (or more realistically, two poles placed to catch the ball as if it was in a corner), and as the spiral spins, the balls will slowly move upwards. Once the balls reach the top, they will hit a ramp and roll into the basket, from the top (see figure 14 in section 6 for a diagram).

Table 9: Functional requirements table for the Vertical Spiral Lift

Module	Vertical Spiral Lift
Inputs	- Power: 12 V DC - Ping pong ball
Outputs	- Ball in basket
Functionality	Uses an auger-like spiral design to lift the ball vertically, in order to deposit the ball into the collection basket from the top.

Level 2 Block Diagram - Camera Vision

Author: SK

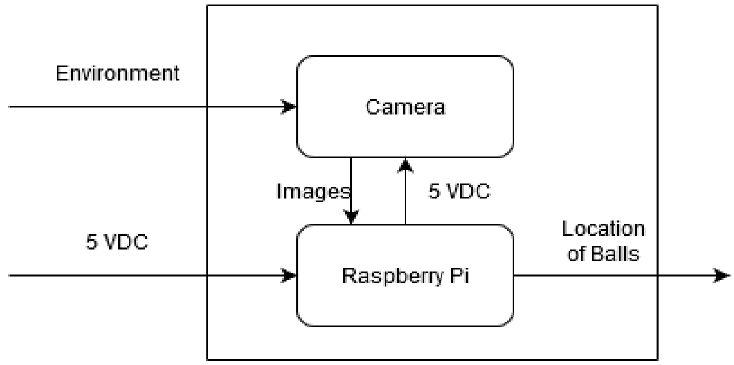

Figure 9: Level 2 block diagram for camera vision

For the robot to detect ping pong balls in its environment, it will use a camera to take pictures and analyze those pictures to identify a ping pong ball's color and shape. The camera will be connected to a Raspberry Pi to perform all the image processing. The Raspberry Pi runs on 5V, and it is recommended to be able to supply 2.5A to it. This will be supplied by the voltage regulators in the power system. It will connect over a ribbon cable to the camera, which will supply power to the camera. The camera itself will take pictures and send the images back to the Raspberry Pi over a ribbon cable. Finally, once the Raspberry Pi has finished the image processing, it will send the results to the main control board.

Table 10: Functional requirements table for Camera Vision

Module	Camera Vision
Inputs	- Power: 5 V DC - Environment: the area the robot is in, to scan for ping pong balls
Outputs	- Location of balls: where any balls detected in the scan are located in the environment
Functionality	Take a picture of the area the robot is in and analyze the image, looking for ping pong balls, to determine the location of the balls (if any) in the robot's environment. The image processing should analyze the image for both the color and the shape of ping pong balls. The location of the balls will be transmitted to the main control board over a serial data transmission protocol.

Level 2 Block Diagram - Power System

Author: JC

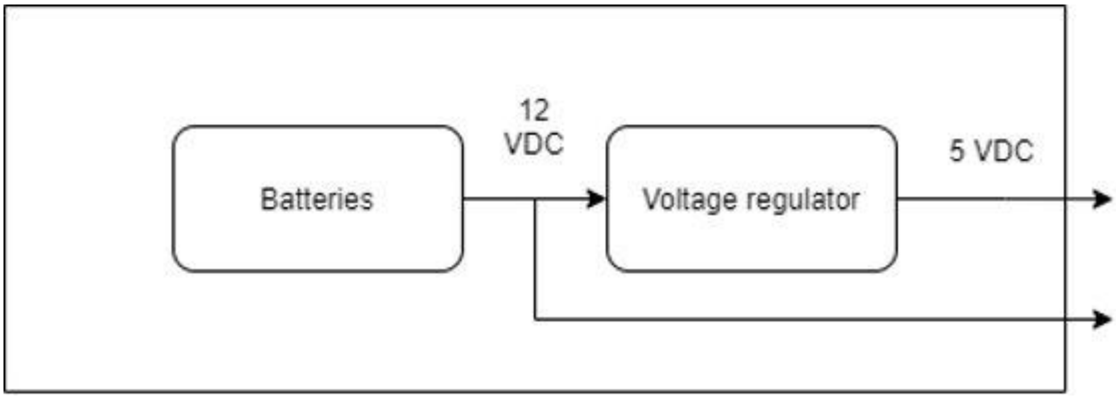

Figure 10: level 2 block diagram for the power system

The flow chart above shows the power system for the robot. The system is powered by a configuration of batteries that output approximately 12V, which goes to the motor controllers to power the motors. In order to supply power to the additional subsystems, a 5V voltage regulator is used to step down the voltage from 12V to a usable voltage 5V for the Raspberry Pi. Since the LiPo batteries have to operate with a certain capacity to preserve battery life, the LiPo batteries will only operate above 20 percent capacity.

Table 11: Functional requirements table for the Power System

Module	Power System
Inputs	N/A
Outputs	-Power: 12 V DC -Power: 5 V DC
Functionality	Takes the power from the battery and supplies it to the rest of the system.

5.2 Software Design

Level 0 Software Flowchart

Author: SK

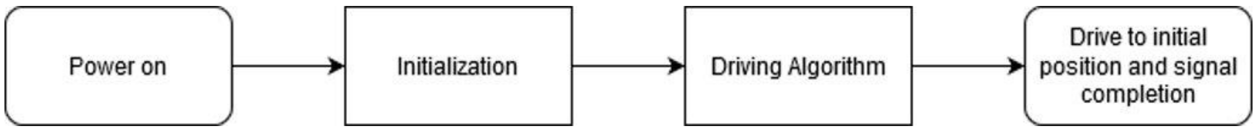

Figure 11: Level 0 software flowchart

The flowchart above shows the overall algorithm that the robot follows throughout its entire operation. When the power is turned on, the initialization code will run. Once initialization is complete, the driving algorithm will begin, and the robot will execute this algorithm until it has completed its task. Once this occurs, it will drive to the initial position and signal to the user that it has completed its task.

Level 1 - Initialization Flowchart

Author: SK

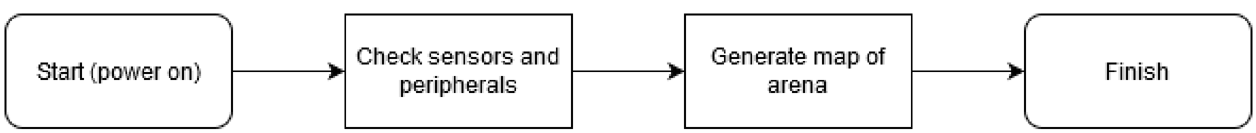

Figure 12: Level 1 initialization flowchart

When the robot is powered on, the initialization routine will run. First, it will check all sensors and peripherals to ensure they are communicating with the main control board. The robot will then use the known dimensions of the arena to generate a map of the arena, made up of individual nodes, and subdivide the arena into smaller areas made up of multiple nodes that are close to each other. Then the routine will finish, and the Driving Algorithm will begin.

Level 1 - Driving Algorithm Flowchart

Author: SK

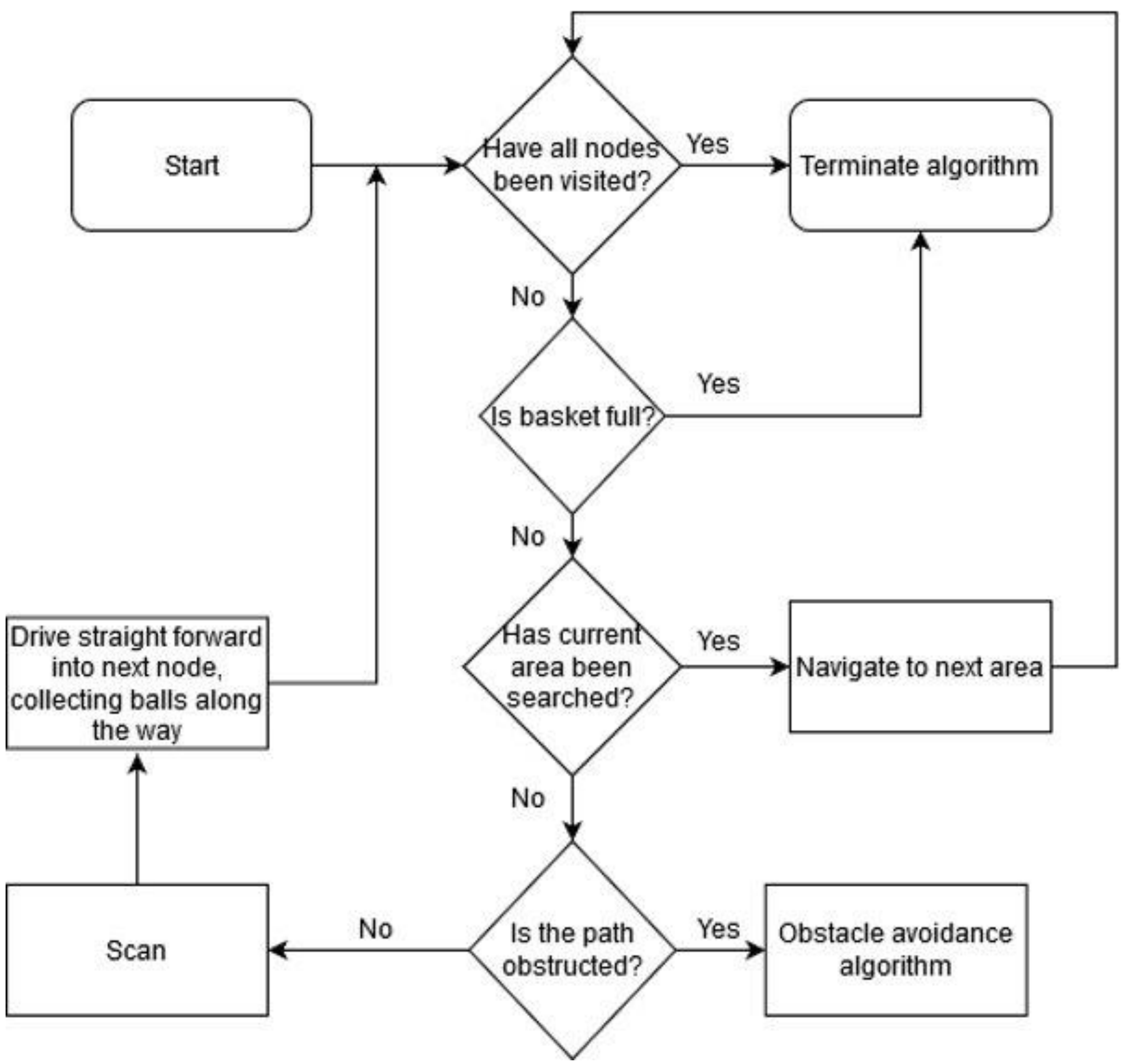

Figure 13: Level 1 driving algorithm flowchart

The driving algorithm begins once the robot has completed its initialization routine. The robot first checks three conditions: if all nodes in its map have been visited, if the basket is full, and if the battery is below 20% charge. If any of these conditions is true, it terminates the Driving Algorithm. If not, the robot then checks to see if it has visited all nodes in the current area it is searching. If so, the robot will navigate to the next closest area and continue searching for balls there. If not all nodes in the current area have been visited, the robot next checks to see if its current path is obstructed. If so, it begins an obstacle avoidance algorithm. The specific obstacle avoidance algorithm that will be used has not been determined yet. If not, it scans the area in front of it to determine if any balls are present. Depending on the results of this scan, it then continues on its path into a new node looking for more balls, or drives directly to a ball and picks it up. This algorithm then repeats until a termination condition is reached.

Driving Path Diagram

Authors: SK, JR

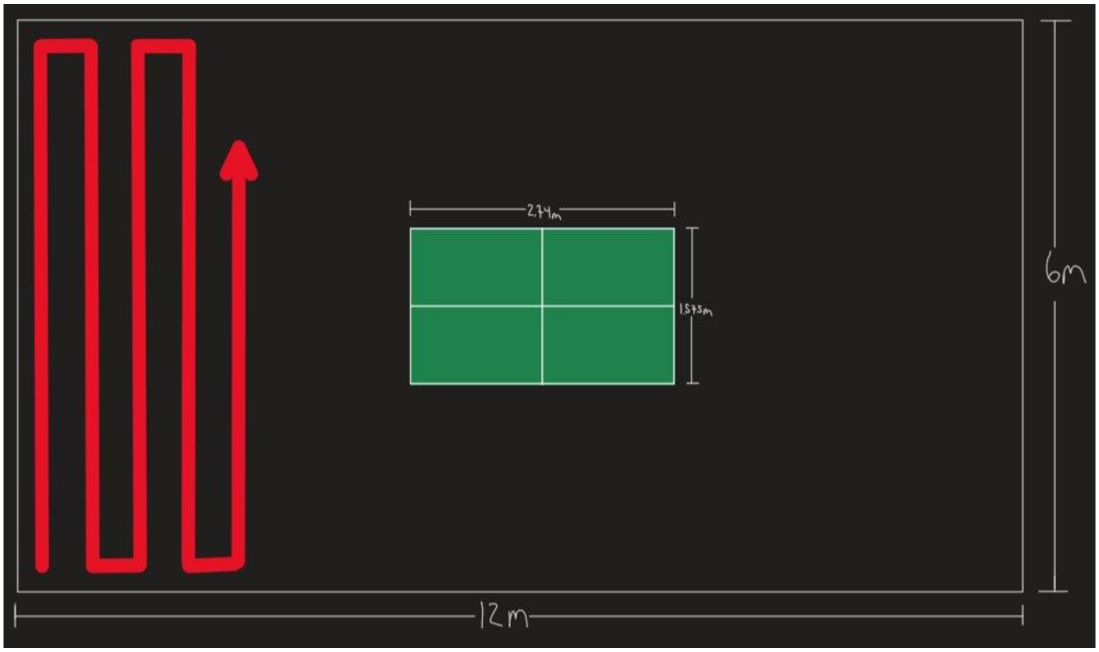

Figure 14: Boustrophedon path and arena/table dimensions.

The driving pattern the robot will follow is called a boustrophedon path (Ntawumenyikizaba, et al., 2018). This is shown in Figure 13 above. This path can be used to cover all accessible areas when the robot has no prior knowledge of its environment or the number of obstacles, in a finite and closed environment. It minimizes the traversal time and the number of overlapping paths, while being fairly simple. The robot will follow this path to optimize the time spent traversing the arena, while keeping track of the nodes it has already visited. If the robot must navigate away from the path to pick up a ball or avoid an obstacle, it will use the A* search algorithm to determine the best path to the next unvisited node in the current search area.

Level 2 - Scanning Algorithm Flowchart

Author: SK, GL

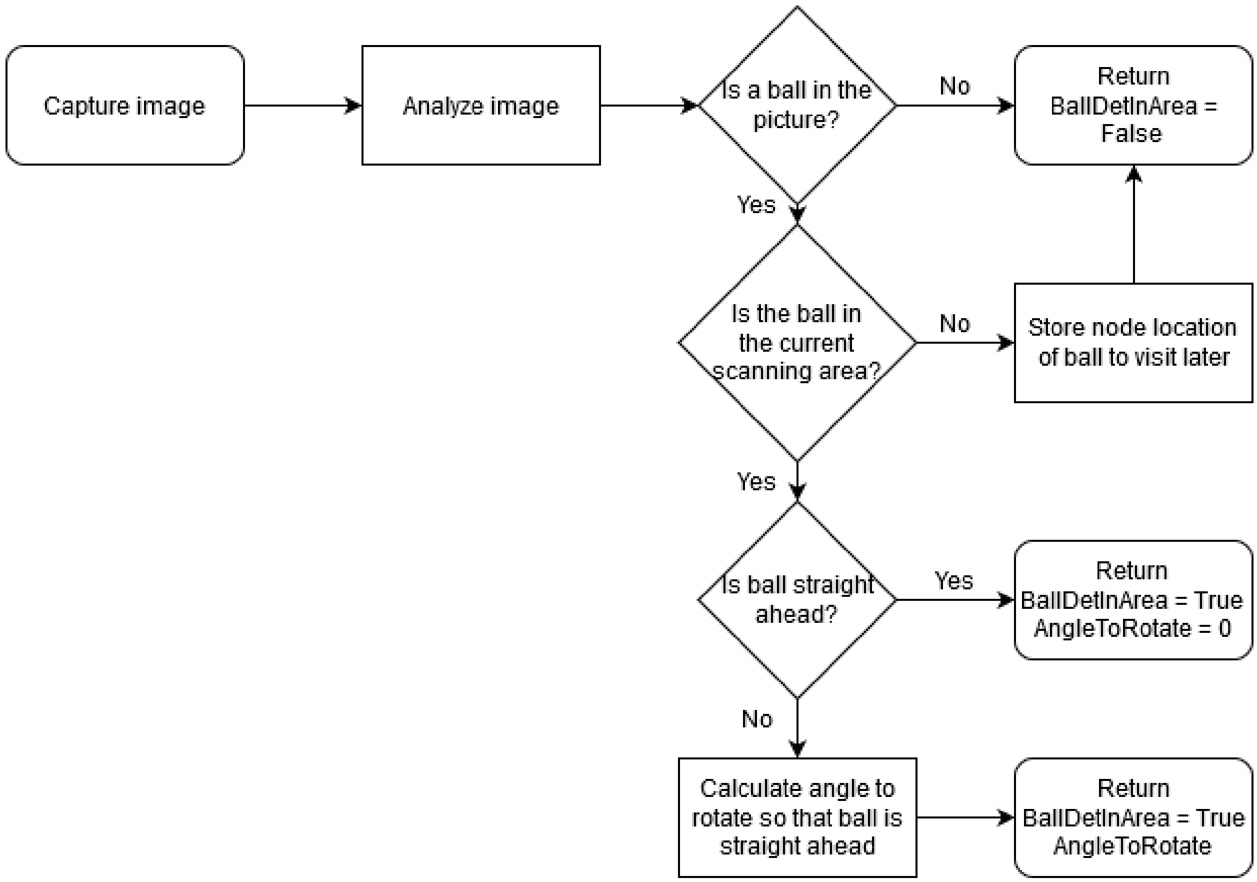

Figure 15: Level 2 Scanning Algorithm Flowchart

The scanning algorithm serves the purpose of scanning the area to determine where the robot will travel next. The first step is to take a picture of the area using the webcam attached to the Raspberry Pi, then analyze the picture, looking for a ping pong ball. If no ping pong ball is found, the robot will return that there is no ball in the area to the main control board. If a ping pong ball is found in the picture, a check is done to see if it is in the current scanning area. If the ball is outside of the scanning area but within the picture, the node location of the ball is stored so that it can be visited later. If the found ball is within the current scanning area, a check is done to see if the ball is straight ahead of the robot, at a small enough angle that the robot does not have to turn to collect it. If this is the case, the robot returns that a ball is present to the main control board and that the robot does not have to rotate to reach it. If the ball is within the current scanning region but is not straight ahead, the angle of rotation necessary to reach the ball and collect it is calculated. This angle to rotate as well as an indication that a ball is present in the area are returned to the main control board.

Level 3 - Image Analysis Flowchart

Author: SK

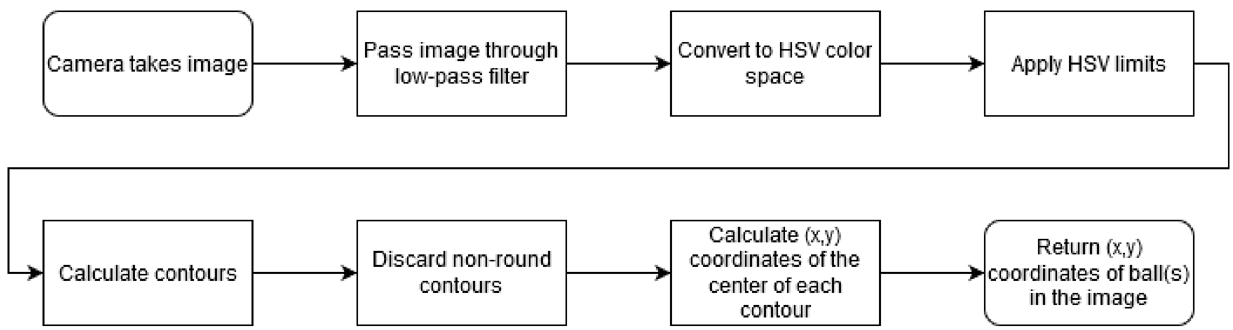

Figure 16: Level 3 Image Analysis Flowchart

The way the ball detection process will work is as follows. An image taken by the camera will be smoothed with a low-pass filter to reduce high frequency noise, then converted to the HSV color space. The lower and upper HSV color limits will then be applied to the image, producing a binary mask showing the location of any white objects in the image. Next, the contours of all white objects will be calculated, and only round contours will be kept. Then, for each round object detected, the (x,y) coordinates of the center will be calculated. These coordinates will then be transmitted to the main control board.

6. Mechanical Sketch

Authors: JC, JR

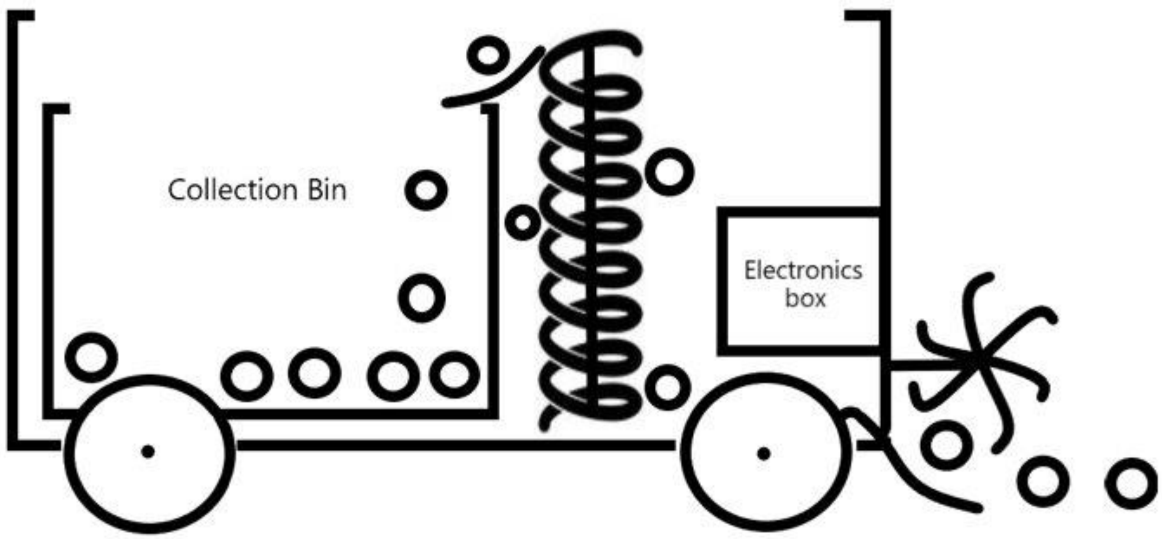

Figure 17. Mechanical sketch of entire robot system

This simple mechanical sketch of the robot encapsulates its main functions. Following the path of the ball, start with the ball collection mechanism. The ball starts on the floor and is detected using the computer vision algorithm. The ball collection mechanism will turn on when a ball has been detected, and remain spinning for a specified timeout period. The collection mechanism will use rotational motion, similar to the bristles of a vacuum. The plan is also to use a flexible material to pick up the ball, so as to not damage the ball or accidentally come in bad contact with the ball, sending it flying unexpectedly. Once the ball has been collected, it enters a small buffer zone, which has a slight decline. This allows for the robot to hold a small amount of balls while the vertical spiral lift (VSL) is full. From the buffer, the ball enters the VSL: a simple design inspired by an auger. Once the ball travels up the VSL, it deposits the ball into the collection basket, from the top.

7. Design Team Information

Jason Colonna, Hardware Lead, Electrical Engineering

Sarah Kuchcinski, Software Lead, Electrical and Computer Engineering

Gina Lanese, Project Manager, Computer Engineering and Music

JohnDavid Rogers, Archivist, Computer Engineering

8. Project Schedules

Authors: JC, SK, GL, JR

Task Name	Duration	Start	Finish	Resource Names
SDP1 Fall 2020 Gantt	**100 days?**	**Mon 8/24/20**	**Wed 12/2/20**	
	1 day?	Mon 8/24/20	Tue 8/25/20	
SDP I 2020		Mon 8/24/20		
Project Design	**98 days**	**Wed 8/26/20**	**Wed 12/2/20**	
Midterm Report	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Jason,JohnDavid,Sarah
Cover page	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Jason,JohnDavid,Sarah
T of C, L of T, L of F	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Jason,JohnDavid,Sarah
Problem Statement	**40 days**	**Wed 8/26/20**	**Mon 10/5/20**	**Gina,Jason,JohnDavid,Sarah**
Need	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Jason,JohnDavid,Sarah
Objective	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Jason,JohnDavid,Sarah
Background	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Jason,JohnDavid,Sarah
Marketing Requirements	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Jason,JohnDavid,Sarah
Engineering Requirements Specification	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Jason,JohnDavid,Sarah
Engineering Analysis	**40 days**	**Wed 8/26/20**	**Mon 10/5/20**	**Gina,Jason,JohnDavid,Sarah**
Circuits (DC, AC, Power, ...)	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Sarah
Electronics (analog and digital)	40 days	Wed 8/26/20	Mon 10/5/20	Jason,Sarah
Image Processing	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Sarah
Communications (analog and digital)	40 days	Wed 8/26/20	Mon 10/5/20	JohnDavid
Electromechanics	40 days	Wed 8/26/20	Mon 10/5/20	JohnDavid
Embedded Systems	40 days	Wed 8/26/20	Mon 10/5/20	Jason,Sarah
Accepted Technical Design	**40 days**	**Wed 8/26/20**	**Mon 10/5/20**	**Jason,Gina,JohnDavid,Sarah**
Hardware Design: Phase 1	**40 days**	**Wed 8/26/20**	**Mon 10/5/20**	Jason,Gina,JohnDavid,Sarah
Hardware Block Diagrams Levels 0 thru N (w/ FR table	38.38 days	Wed 8/26/20	Mon 10/5/20	Jason,Gina,JohnDavid,Sarah
Software Design: Phase 1	**40 days**	**Wed 8/26/20**	**Mon 10/5/20**	**Gina,Sarah**
Software Behavior Models Levels 0 thru N (w/FR table	40 days	Wed 8/26/20	Mon 10/5/20	Gina,Sarah
Mechanical Sketch	40 days	Wed 8/26/20	Mon 10/5/20	Jason
Team information	40 days	Wed 8/26/20	Mon 10/5/20	Jason
Project Schedules	**40 days**	**Wed 8/26/20**	**Mon 10/5/20**	**Gina**
Midterm Design Gantt Chart	40 days	Wed 8/26/20	Mon 10/5/20	Gina
References	40 days	Wed 8/26/20	Mon 10/5/20	Jason,Gina,JohnDavid,Sarah

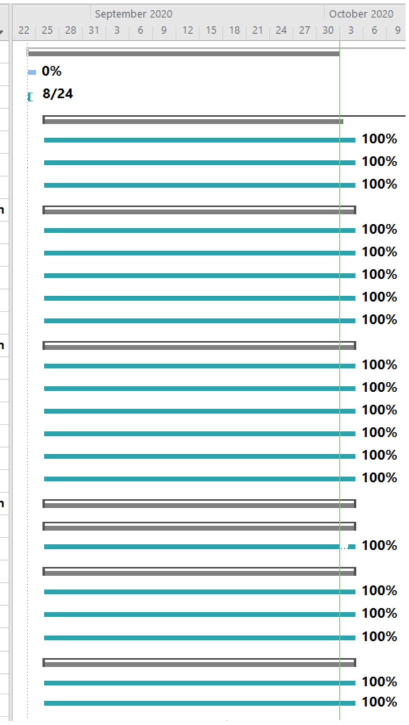

Figure 18: Gantt Chart for project

9. Conclusion

Author: JR

In conclusion, the premise of this robot has the possibility to revolutionize how ping pong lessons are conducted, and in general change upper level table tennis practice, by simplifying ball collection following any practice session. This robot might be slow in this first iteration, but the fundamentals of the systems engineering put into it allow a basis to be formed for future iterations that may be manufactured. The design decisions that have been made so far should cement a path to a successful robot. However, as this is only the design phase, it is inevitable that the robot will change once prototyping begins. Regardless, the engineering efforts put into the research and design thus far will inevitably pave the way for the success for this revolutionary robot.

10. References

Dirksen, B. M., Gotto, T. J., Halbur, S. G., & Torres, A. M. (n.d.). Wireless, web-controlled, ball-collecting robot. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.4.6795&rep=rep1&type=pdf

Espinal, A., Rostro-Gonzalez, H., Carpio, M., Guerra-Hernandez, E. I., Ornelas-Rodriguez, M., Puga-Soberanes, H. J., Sotelo-Figueroa, M.A., & Melin, P. (2016). Quadrupedal robot locomotion: a biologically inspired approach and its hardware implementation. Computational Intelligence and Neuroscience, 2016, 1–13. doi: 10.1155/2016/5615618

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size*.

Jones, J.L., & Mass, P.R.; *US Patent No. 6.809,490*; Washington, D.C. US Patent and Trademark Office.

Jung, D., Lim, Z., Kim, B., & Kim, N. (2007) Multi-channel ultrasonic sensor system for obstacle detection of the mobile robot. 2007 International Conference on Control, Automation and Systems, Seoul, pp. 2347-2351.

Nie, H., Cai, G., Liu, B., Hu, X., Yang, F., Ni, J., & Hou, X. (2019). On Development of An Autonomous Ball Collecting Wheeled Mobile Robot. *2019 3rd Conference on Vehicle Control and Intelligence (CVCI), Vehicle Control and Intelligence (CVCI), 2019 3rd Conference On*, 1–5. https://doi-org.ezproxy.uakron.edu:2443/10.1109/CVCI47823.2019.8951531

Ntawumenyikizaba, A, Viet, H. H., & Chung, T. (2012). An online complete coverage algorithm for cleaning robots based on boustrophedon motions and A* search. *2012 8th International Conference on Information Science and Digital Content Technology (ICIDT2012)*, Jeju, 2012, pp. 401-405.

Raspberry Pi Foundation. (n.d). *Raspberry Pi 4 model B specifications.* https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/

Szeliski, R. (2010). *Computer Vision: Algorithms and Applications* (1st). Springer.

Szemenyei, M., & Estivill-Castro, V. (2019). *ROBO: robust, fully neural object detection for robot soccer.*

Tsai, K.; Mar. 9 2010; *US Patent No. 7,674,196*; Washington, D.C. US Patent and Trademark Office.