Critical Design Review (CDR) report

Pseudo GPS for Romi
Critical Design Review

May 2, 2024

For Charlie Refvem

Team
Emmanuel Baez – embaez@calpoly.edu
Gabriel Coria – gacoria@calpoly.edu
Owen Guinane – oguinane@calpoly.edu
Conor Schott – cjschott@calpoly.edu

College of Engineering
California Polytechnic State University, San Luis Obispo
2024

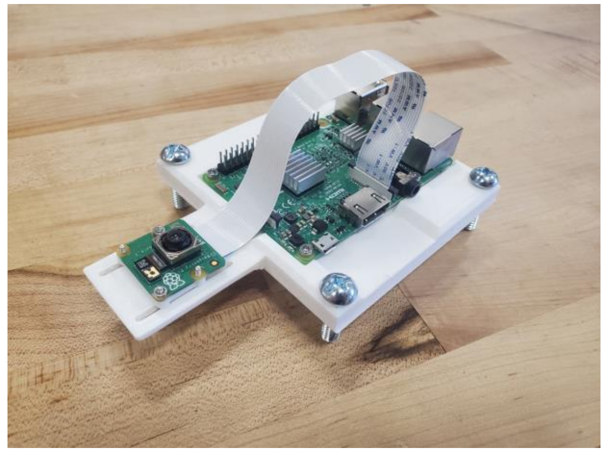
Table of Contents

Statement of Disclaimer

Since this project is a result of a class assignment, it has been graded and accepted as a fulfillment of the course requirements. Acceptance does not imply technical accuracy or reliability. Any use of information in this report is at risk to the user. These risks may include catastrophic failure of the device or infringement of patent or copyright laws. California Polytechnic State University at San Luis Obispo and its staff cannot be held liable for any use or misuse of the project.

Four Panel Chart

Table 1. Four-Panel Chart

Project Overview	Concept Description
• Project Objective: Currently, the Romi robots used in the Mechatronics lab have no way of obtaining absolute orientation or location from onboard sensors. The Cal Poly ME 405 instructor, Charlie Refvem, and Mechatronics students need a way to deliver precise and real-time location data for multiple Romi robots for the development of better algorithms. Our aim is not just to improve tracking but also to enhance communication standards among multiple Romis by ensuring geolocation accuracy, precision, range, latency, and update rates. • Documentation and Impact: The project involves comprehensive documentation, including detailed electrical schematics and code explanations.	 The idea with our structural prototype is to prove that our system will be able to detect the Romi bots as well as that it will properly mount to a ceiling tile. In the picture above is our Raspberry Pi and Camera Module wired together ready to be used.
Concept Justification	**What's Next**
The decision to integrate camera technology into the project is based on its accuracy, providing precise locational data within ±5mm. The Raspberry Pi camera's real-time data capabilities and fast update rate enable monitoring of multiple Romi robots. Additionally, the camera's cost-effectiveness and plentiful documentation online further supports its selection for enhancing tracking capabilities.	This quarter we will have to keep working on our software to get the detection to work, as well as implementing a grid system to help with the position of the Romi bots. We will perform multiple pass/fail tests on the detection and position and safety tests with our system mounted. The safety test for mounting the system will consist of FEA on our ceiling tile, and a shake test to see if the system will fall.

Overview

We have finalized the concept selection and are currently expanding further iterations on our concept prototype. This report aims to describe our chosen concept, our reasoning behind it, and a table of the alternate solutions considered. In addition, this report also includes a preliminary analysis, a weighted design matrix, an updated Gantt chart, a design hazard matrix, and our references. We hope to get feedback on our reasoning for the sensor choice and on the housing system.

Concept Description and Justification

For our system, we decided to use a Raspberry Pi with a camera to obtain absolute orientation and positional data. The basic layout is depicted in Figure 1 below.

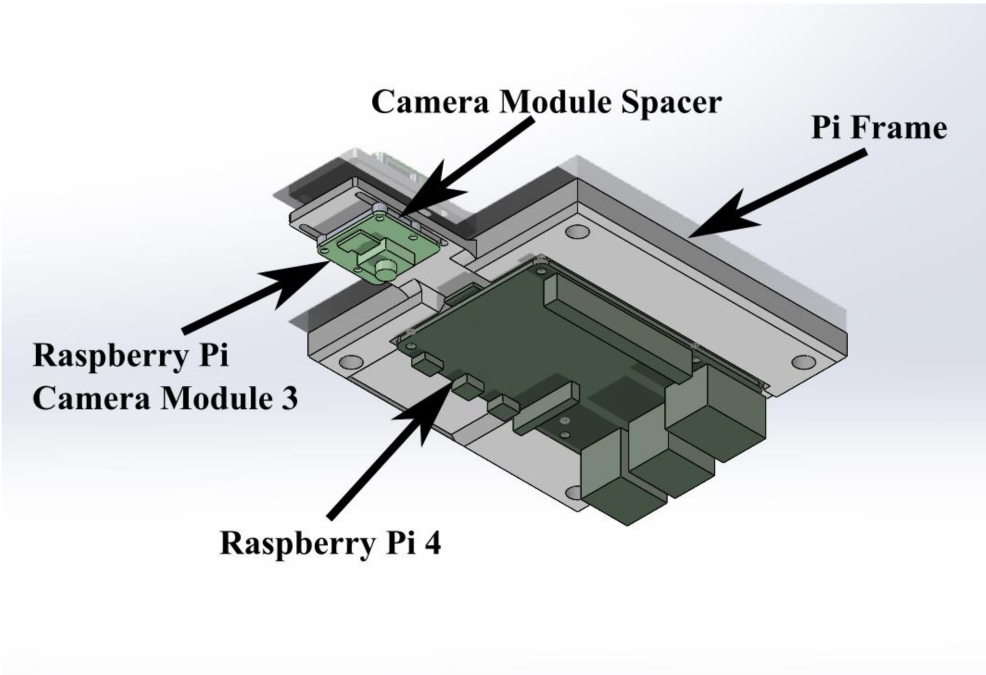


Figure 1. General layout of system

Mounted overhead, the camera captures images of the lab table, which are processed to generate a grid overlay in 2-dimensional space, facilitating precise localization and differentiation of objects. The device then creates an origin point on the table, which it will use to measure Romi bots and obstacles relative to it. It will constantly monitor Romis and obstacles within its field of view. Data, such as grid representation and object positioning will be handled by our microcontroller for processing. Through various algorithms, the microcontroller categorizes objects based on predefined characteristics and determines their relative locations with a precision of ±5mm, utilizing image processing techniques like object recognition and spatial mapping to track and locate Romis and obstacles in real-time, ensuring current and reliable tracking data.

The software integrates computer vision and machine learning techniques to detect Romis, obstacles, and the table. Initially, essential components such as the COCO dataset's class names and the SSD MobileNet V3 model configuration are imported. This setup enables the system to recognize a wide range of objects with high precision. Upon initialization, the detection model is configured with specific parameters for optimal performance. During operation, the software continuously captures frames from the overhead camera feed and undergoes object detection using the pre-trained SSD MobileNet V3 model. Detected objects are identified based on their confidence scores and class labels, and bounding boxes are drawn around them for visualization. Simultaneously, the software establishes a grid overlay on the detected table and defines an origin point, facilitating precise localization and measurement of objects within the environment. Coordinates of Romis and obstacles relative to the table's origin are then transmitted to the microcontroller for further processing and monitoring. Through the integration of object detection, grid generation, and coordinate calculation functionalities, the software ensures accurate detection and monitoring of Romis and obstacles.

Safety

To ensure the safe operation of our system, several main concerns need to be addressed. The largest safety concern with our system comes with mounting. Because the system is roof-mounted, we need to design the system to be mounted securely and minimize the chance of falling out, due either to structural failure of mounting parts or by not fitting properly within a drop-down roof tile frame. To ensure this does not occur, we plan on performing finite-element analysis on the mounting hardware to verify that it can hold the entire weight of the system with a safety factor of 5. Minor safety considerations include adding fillets and chamfers to 3-D printed parts to eliminate sharp corners and edges which can cut the user when handling.

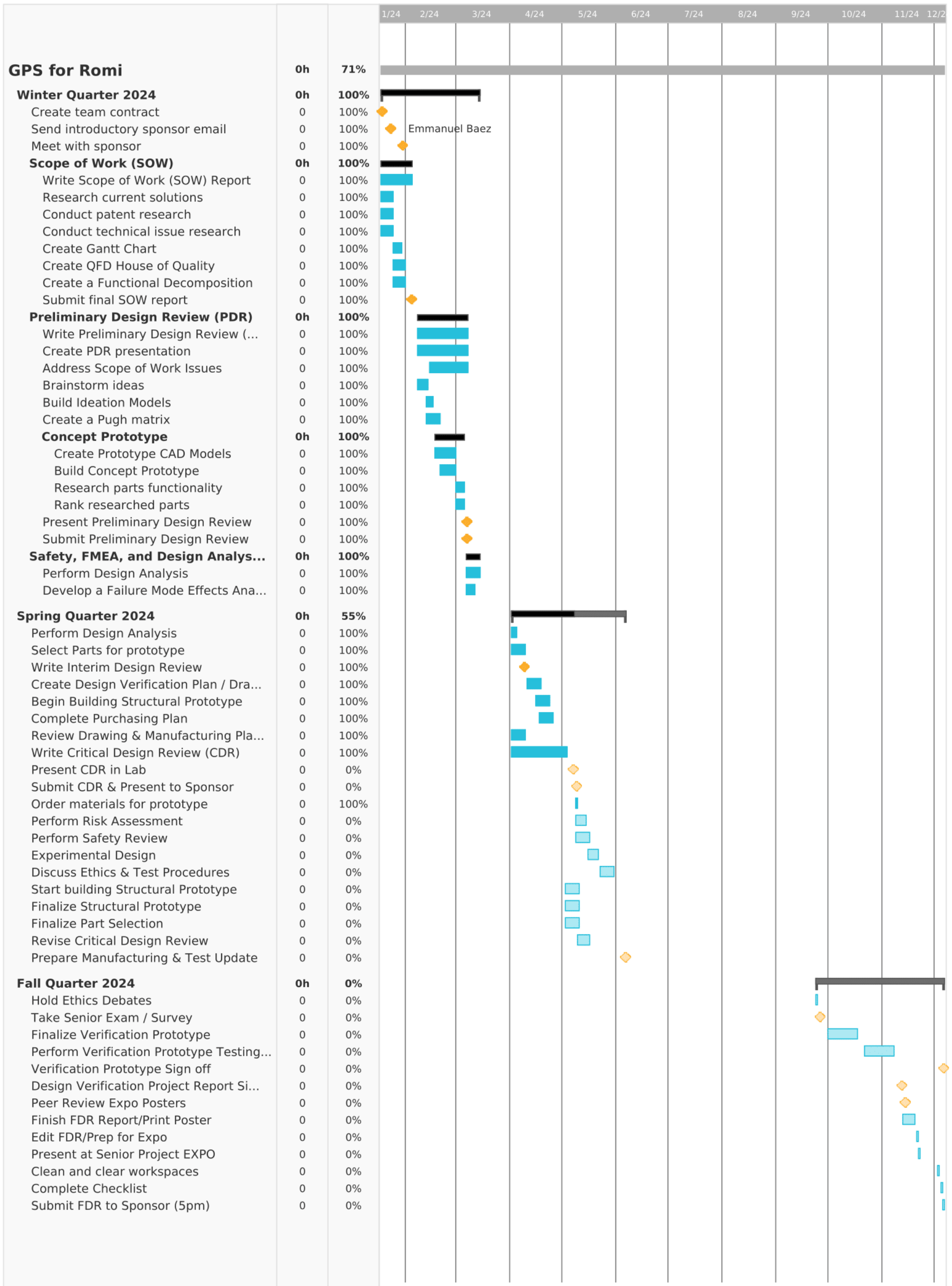
Justification Table

Table 2. Justification Table

Spec #	Specification Title	Specification Description	Target (units)	Justification
1	Total System Length	The system would prefer to be able to fit onto a Romi.	Romi Size	Currently, we plan to attach an ESP32 module to the Romi bots for communication. Our sponsor has approved.
2	Total System Weight	The Romi will also have other sensors attached to it, so we would not want to encumber it further.	2 lbs.	Currently, we have plans to attach an ESP32 module to the Romi bots for Communication, and our sponsor said it's okay.
3	Field of View	The system should track a Romi placed on the classroom table.	96x48 in.	The Raspberry Pi Camera 3 modules have a wide enough field of view to capture the table in its entirety, depending on the mounting distance.
5	Data Transfer	The system should be able to transfer data at a high rate.	6 Hz	The Raspberry Pi 4 that we purchased has the specs to transfer data at 6 Hz.
6	Number of Romis being detected	The system should be able to track multiple Romis	6 bots	Our tracking camera algorithm combined with the camera positioning will be able to track 6 Romis.
7	Tracking Resolution	The system should be able to track the position of Romi.	+/- 5 mm	The Raspberry Pi Camera 3 modules will have enough resolution to track Romi position.
8	Production Cost	The budget is $500. (could be higher)	$500	We have an Excel that we put in every purchase to make sure we are within budget and notify our sponsor when we need to make a purchase.

Appendix

Updated Gantt Chart



Design Hazard Checklist

Table 4. Design Hazard Checklist Table

Y	N	
	N	1. Will any part of the design create hazardous revolving, reciprocating, running, shearing, punching, pressing, squeezing, drawing, cutting, rolling, mixing or similar action, including pinch points and sheer points?
	N	2. Can any part of the design undergo high acceleration/deceleration?
	N	3. Will the system have any large moving masses or large forces?
	N	4. Will the system produce a projectile?
Y		5. Would it be possible for the system to fall under gravity creating injury?
Y		6. Will a user be exposed to overhanging weights as part of the design?
	N	7. Will the system have any sharp edges?
	N	8. Will any part of the electrical systems not be grounded?
	N	9. Will there be any large batteries or electrical voltage in the system above 40 V?
	N	10. Will there be any stored energy in the system such as batteries, flywheels, hanging weights or pressurized fluids?
	N	11. Will there be any explosive or flammable liquids, gases, or dust fuel as part of the system?
	N	12. Will the user of the design be required to exert any abnormal effort or physical posture during the use of the design?
	N	13. Will there be any materials known to be hazardous to humans involved in either the design or the manufacturing of the design?
	N	14. Can the system generate high levels of noise?
	N	15. Will the device/system be exposed to extreme environmental conditions such as fog, humidity, cold, high temperatures, etc?
	N	16. Is it possible for the system to be used in an unsafe manner?
	N	17. Will there be any other potential hazards not listed above? If yes, please explain on reverse.

Manufacturing Plan

Table 5. Manufacturing Plan table

Sub system	Component	Purchase (P) Modify (M) Build (B)	Raw Materials Needed to make/modify the part (only M & B)	Where/how procured?	Equipment and Operations anticipate using to make the component	Key limitations of this operation places on any parts made from it
Ceiling Mount	Ceiling Tile	M	Ceiling Tile	Provided by Wood Workshop at Cal Poly for use in Lab 192-118	11/16" Drill Bit, Cordless Drill to make holes that the threaded inserts will screw into.	n/a
	1/4"-20 Zinc Plated Insert Nut	P	n/a	Purchased at Home Depot	n/a	n/a
Housing	Main Housing	B	PLA Plastic	3D printed	3D Printer to create the housing, M2.5 x .45, 1/4"-20 tap drill bit and a tap s to create internal threads on the large outer holes and small inner holes. Deburring may be necessary after tapping.	Small size may prevent M2.5 holes from printing properly. Additional drilling may be required
	1/4"-20 1" Machine Screws	P	n/a	Purchased at Home Depot	n/a	n/a
	M2.5 10mm Screws	P	n/a	Purchased from Amazon	n/a	n.a
	M2 10mm screws	P	n/a	Purchased from Amazon	n/a	n/a
Microcontroller	Raspberry Pi 4 Model B	P	n/a	part already provided by sponsor (Adafruit)	n/a	n/a
	Micro HDMI Cable	P	n/a	part already provided by sponsor (Adafruit)	n/a	n/a
	SD card	P	n/a	part already provided by sponsor (Adafruit)	n/a	n/a
	Heat Sink	P	n/a	part already provided by sponsor (Adafruit)	n/a	n/a
	Power Supply	P	n/a	part already provided by sponsor (Adafruit)	n/a	n/a
Camera	Raspberry Pi Camera Module 3	P	n/a	part already provided by sponsor	n/a	n/a
	Camera Spacer	B	PLA Plastic	3D printed	3D Printer to create the part	Small Size may prevent M2 holes from printing to size. Screws may have to be threaded through.

Manufacturing Steps

Ceiling Tile Assembly
<u>Ceiling Tile</u> (110)
11/16” Drill Bit with Cordless Drill

1. Drill four holes that are 11/16” in diameter on the top (floor-facing) side of the ceiling tile to allow the threaded inserts to screw in.
2. Screw 1/4”-20 Zinc Plated inserts into drilled holes. These will connect the main housing to the ceiling tile.

Housing Assembly
<u>Main Housing</u> (210)
3D Printer

1. Using the provided file “MainHousing.STL”, use a 3D printer to print the housing. Refer to printer documentation for instructions on how to print an STL file.
   1/4"-20 Tap drill bit and Tap
2. On the larger outer 4 holes and center hole, tap these holes all the way through. These holes will secure the housing to the ceiling tile.
   M2.5 x 45 Tap drill bit and Tap
3. On the smaller inner 4 holes, tap these holes all the way through. These will be used to secure the Raspberry Pi 4 to the housing.

Camera Assembly
<u>Camera Spacer</u> (2320)
3D Printer

1. Using the provided file “CamSpacer.STL”, use a 3D printer to print the spacer. Refer to printer documentation for instructions on how to print an STL file.

Assembly instructions

1. Screw the Camera Spacer (2320) to the slots of the Main Housing (210) using four M2 10mm Machine Screws (250) so that camera spacer is on the top side of the part and the threads are facing upward.
2. Insert the MicroSD Card (2220) into the Raspberry Pi 4 (2210) to make the Microcontroller Assembly.
3. Screw the Microcontroller Assembly (220) into the main housing with four M2.5 10mm Machine Screws (250) until it is finger tight. Over-tightening the screw may cause damage to the board.
4. Insert the Raspberry Pi Camera Module 3 (2310) on the threads of the M2 Machine Screws. Use M2 Nuts (270) to secure the camera to the spacer.
5. Connect the camera module to the Raspberry Pi 4 using the module’s ribbon cable.
6. Use a 6mm hex key to screw the threaded inserts (120) into the ceiling tile to make the Ceiling Tile Assy (10).
7. Using the four 1/4” – 20 1” Machine Screws (240), screw the Main Housing to the holes in the Ceiling Tile Assy (10) to create the Final Assembly (1).
8. Install the Final Assembly in place of a similar ceiling tile or empty ceiling tile frame over desired tracking area.
9. Connect Power Supply (31) and Micro HDMI (30) cables as needed.

Design Manufacturing Plan

Table 3. Design Verification Plan Table

Planned Test	What it will test (user need or specific spec, whichever is most applicable for your project and sponsor)	Required resources
Detection Test	Test to see if the code runs with no errors and if the camera displays a picture while detecting objects.	Raspberry Pi, camera, computer
Ceiling Mount Test	Testing to inspect whether the hardware will stay on the ceiling	Ceiling tile, Raspberry Pi, Housing, Camera
Rate of Transfer test	Test to see how fast the data from the Raspberry Pi is transferred to a computer.	Raspberry Pi, Computer
Fitment and Weight test	Test to ensure that ESP32 module will fit comfortably on the Romi robots without encumbering them	ESP32 Module, Romi Robots
Resolution test	Test to determine how accurate the camera can track a single Romi Robot with a target accuracy of	Romi Robot, Final assembly, computer
Swarm detection test	Test to ensure that system is capable of tracking multiple Romi robots simultaneously without issues.	Romi Robots, Final Assembly, Computer

Budget

Table 6. Budget Chart

Romi Budget Plan	
Part	Cost ($)
Micro HDMI to HDMI cable	8.95
Raspberry Pi Camera Module 3 Standard - 12MP Autofocus	25
Raspberry Pi 4 model B-8GB RAM x2	150
Official Raspberry Pi Power Supply 5.1V 3A with USB C	7.95
SD/MicroSD Memory Card - 16GB Class 10 - Adapter Included	19.95
Official Raspberry Pi 4 Case Fan and Heatsink	5
Parts Total	216.85
Initial Budget	500
Current Budget	283.15

Idented Bill of Materials

Psuedo GPS
Indented Bill of Material (iBOM)

Assy Level	Part Number	Descriptive Part Name	Qty	Part Cost	Source	URL	More Info
		Lvl0 Lvl1 Lvl2 Lvl3 Lvl4					
0	1	Final Assy					
1	10	Ceiling Tile Assy					
2	110	Ceiling Tile	1		Cal Poly Wood Shop		Ask for tiles for lab 192-118
2	120	Threaded Inserts	4	$3.87	Home Depot	https://www.homede	
2	20	Housing Assy					
3	210	Main Housing	1		3D Printed		
3	220	Microcontroller Assy					
4	2210	Raspberry Pi 4 Model B - 8GB Ram	1	$150.00	Adafruit	https://www.adafruit	
4	2220	SD/MircoSD - 16GB class 10	1	$19.95	Adafruit	https://www.adafruit	
3	230	Camera Assy					
4	2310	Raspberry Pi Camera Module 3 Sta	1	$25.00	Adafruit	https://www.adafruit	
4	2320	Camera Spacer	1		3D Printed		
3	240	1/4"-20 1" Machine Screws	4	$1.38	Home Depot	https://www.homede	
3	250	M2.5 10mm Machine Screws	4	$10.86	Amazon	https://www.amazon	Sold in 840 Pc Kit
3	260	M2 10mm Machine Screws	4	$9.99	Amazon	https://www.amazon	Sold in 660 Pc Kit
3	270	M2 Nuts	4	[-]	Amazon		In M2 Screw kit
1	3	Cables					
2	30	Micro HDMI to HDMI Cable	1	$8.95	Adafruit	https://www.adafruit	
2	31	Power Supply 5.1V 3A w/ USB C	1	$7.95	Adafruit	https://www.adafruit	
	Total Parts		**28**	**$237.95**			

Figure 2. Indented Bill of Materials (iBOM)

Supporting Evidence

We can confirm that our field of view and tracking resolution will meet our specifications by performing an analysis of the camera's field of view. For a camera, we have two types of viewing distances that define our viewing area, horizontal distance Dhor and vertical distance Dvert. In our desired configuration where the camera system is mounted to the roof and facing down, we can calculate the viewing area using trigonometry.

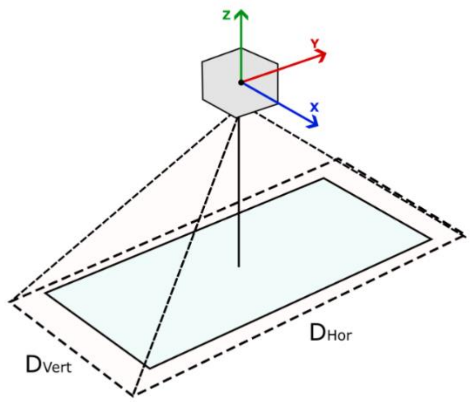


Figure 3. 3D Sketch of camera setup.

Camera manufacturers specify the horizontal and vertical angles for the field of view for any given lens which we'll define as HFOV for the horizontal field of view and VFOV for the vertical field of view. Let's take the case of the horizontal field of view. Assuming a perfectly centered camera, we can model it as a symmetrical triangle of angle HFOV, height Hmount, and distance Dhor. Hmount will be the height at which the system is mounted above the table.

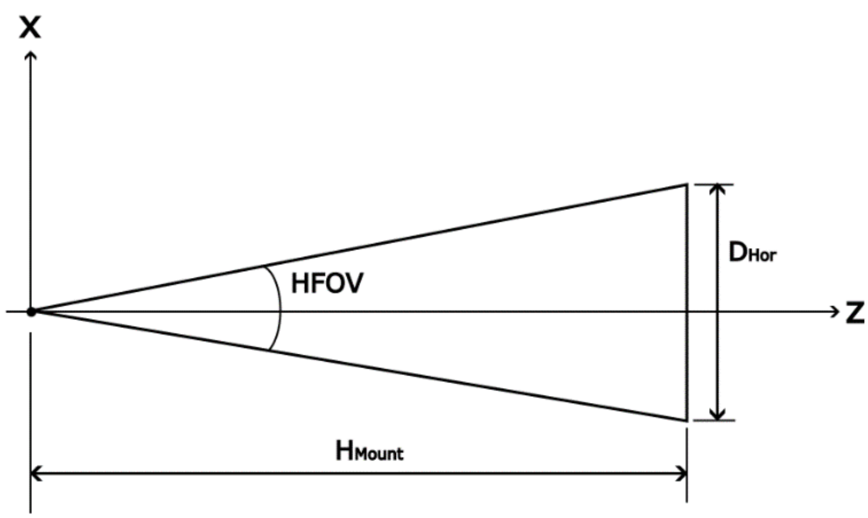


Figure 4. Viewing triangle of camera for horizontal distance

Since the triangle is isosceles, we can split it into two identical right triangles with half the angle and horizontal distance.

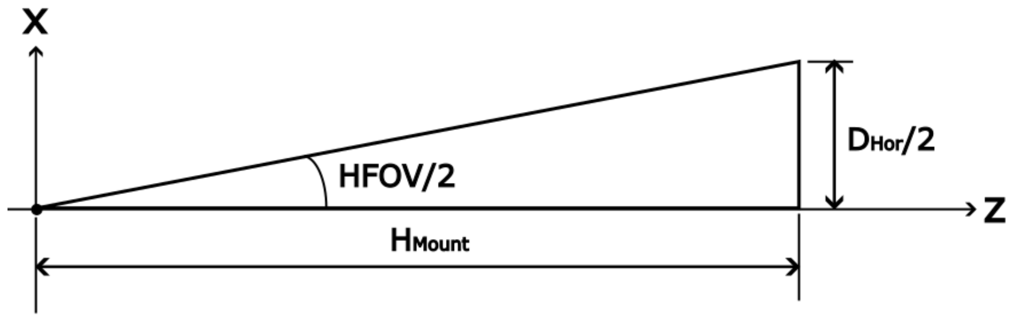


Figure 5. Half of viewing triangle. Trigonometry can now be used to solve for distance.

Using trigonometry, we obtain

$$\tan\left(\frac{HFOV}{2}\right) = \frac{(D_{Hor}/2)}{(H_{mount})}$$

Rearranging the equation,

$$D_{Hor} = 2H_{mount}\tan\left(\frac{HFOV}{2}\right)$$

Solving for the vertical distance D_{Vert} follows a similar process, so similar equation can be used.

$$D_{Vert} = 2H_{mount}\tan\left(\frac{VFOV}{2}\right)$$

To track the whole table, the camera must have a vertical viewing distance of at least 4 feet and a horizontal viewing distance of 8 feet. Using these two distances, we can calculate the viewing area of the camera.

$$A_{View} = D_{Hor} * D_{Vert}$$

Finally, by dividing the viewing area by the pixel count, we can determine the area of each pixel in the image, which will be our tracking resolution area.

$$A_{pix} = \frac{A_{View}}{\#\, of Pixels}$$

A pixel is a square, so using the area of the pixel, we can determine the pixel width, which will be our tracking resolution.

$$W_{pix} = \sqrt{A_{pix}}$$

Now that we have our basic equation, we can substitute our values to determine if our cameras will meet specifications. Measurements taken in the lab give us a maximum mounting height of 7 ft above the table. Accounting for the fact that our camera will not sit flush with the drop-down ceiling, we can assume a reasonable mounting distance of 6.25 ft above the table. To streamline the process of calculating viewing areas, we used an Excel worksheet to run the calculations

Table 7. Camera Calculation

Camera /Lens	Resolution	HFOV	VFOV	Dhor	Dvert	Aview		Apix	Wpix
	[pixels]	[deg]	[deg]	[ft]	[ft]	[ft^2]	[mm^2]	[mm^2]	[mm]
Raspberry Pi Camera Module 3	1.19E+07	66	42	8.12	4.80	39.0	3.62E+0 6	0.30	0.55
Raspberry Pi Camera Module 3 Wide	1.19E+07	102	67	15.44	8.27	127.7	1.19E+0 7	1.00	1.00

From our analysis, it appears that both the Raspberry Pi camera module 3 and Raspberry Pi Camera Module 3 wide will both meet our design specifications.

The ESP32 effectively communicates with our Pi4's Wi-Fi module, ensuring seamless connectivity. Its integration with the Romi Nucleo board, equipped with GPIO pins, ADC, and PWM, facilitates data transmission to and from the Pi4. With low power consumption, especially when powered by the Nucleo, it enables smooth continuous tracking and monitoring tasks. Costing between $10 to $15, the ESP32 offers a cost-effective solution. Its compact dimensions (55mm x 27.9mm x 25mm) and lightweight construction (less than 1 ounce) ensure minimal impact on the Romi's weight distribution.