Portable Unitary Throwable Target Interceptor (P.U.T.T.I.)

Alex Evison, Callum Fuller, Nathan Kimmel, and Jake Tattersall

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* – The Portable Unitary Throwable Target Interceptor (P.U.T.T.I.) is a device aimed to aid soldiers on the frontlines. Drones and thrown devices are just some of the major threats to our troops. The acronym for the device was purposely chosen. It must be Portable and Unitary to make life easy for the trooper. Its main goal is to Target Thrown objects and Intercept them. The P.U.T.T.I. covers a designated area and simulates this interception with a low-powered laser, directed by motors. Computer Vision through YOLO and microwave modules allow for tracking of specified targets at low velocities. A humidity and temperature sensor gives the device confidence that the sensors are under proper conditions. A buzzer will announce the interception of a target so that those around are aware of the imminent and contained threat. By watching out for our troops, the device will aid in saving lives so that our troops can return home safely.**

I. Introduction

Created from the increasing use of drone warfare, the portable unitary throwable target interceptor utilizes two detection methods for detecting slow velocity objects that pass by in front of it. Microwave sensing and computer vision allow for the device to detect objects in front of protecting the flank of users. The project utilizes a single board computer to support object detection software and runs the remaining peripherals on the ESP32 microcontroller. The device comes equipped with a humidity and temperature sensor to ensure proper working conditions throughout the multiple different environments the device may be deployed in. A laser and buzzer also alert the user and show proof of targeting when the camera and microwave sensors work in conjunction with one another to detect the objects in the field of view. This is only achievable through the use of two servos which have a functional rotation of 0-270 degrees of which only 45 to 225 degrees are actually utilized. These servos are controlled through calculations made by the object detection software and microwave sensors to accurately detect and predict the movement of the objects when they appear on the screen. This process is continued throughout the entirety of the time the object is within the view of the camera and microwave sensors and will then reset to a home position to ensure continued, safe, and efficient use of the device.

II. Hardware

A. Microcontroller

The microcontroller serves as the primary interface between the system’s sensing hardware and actuation hardware. While the single board computer performs the computationally intensive computer vision tasks, the ESP32 [1] manages the lower-level peripherals that must respond quickly and reliably during operation. The ESP32 was selected because it supports the major communication protocols required by the project, including UART, I2C, and SPI, while also providing a dual-core processor with an adjustable clock speed up to 240 MHz. This made it suitable for coordinating the radar modules, environmental sensor, display, and actuation components through a single embedded platform.

A major reason for choosing the ESP32 was its flexibility in handling several peripherals at once. The device must receive or interpret targeting information, read environmental data, drive the servos, and control the laser and buzzer without introducing excessive delay. A simpler microcontroller may have been able to manage only part of this workload, but the ESP32 provides enough processing capability and interface support to coordinate all of the supporting electronics of the P.U.T.T.I. device within one controller. This simplifies the hardware design and reduces the need for multiple dedicated control boards.

The ESP32 also helps separate the responsibilities of the full system in a practical way. The single board computer remains focused on object detection and image processing, while the microcontroller handles deterministic real-time tasks such as peripheral timing, output control, and communication with the rest of the hardware. This division of labor improves modularity and makes debugging easier, since the vision pipeline and the embedded control logic can be tested more independently. It also reduces the likelihood that low-level control tasks will interfere with the vision workload.

Another benefit of the ESP32 is its popularity. The platform is widely used in embedded systems and is supported by established documentation, software libraries, and toolchains. For a project integrating several subsystems, this was an advantage because each peripheral could be tested individually before final integration into the complete device. Overall, the ESP32 was selected because it offers a strong balance of communication support, real-time control capability, and practical ease of development for the hardware side of the project

B. Single Board Computer

The single board computer serves as the main processing platform for the vision system and is responsible for running the object detection pipeline on live camera frames. This role required significantly more computational capability than a standard microcontroller could provide, since the project depends on real-time image processing across a wide field of view while maintaining useful frame rates. For this reason, the Nvidia Jetson Orin Nano Super Developer Kit [2] was selected as the primary computing platform for the device. NVIDIA lists the kit with 67 INT8 TOPS of AI performance, making it well suited for embedded AI workloads such as real-time object detection.

A major reason for choosing the Orin Nano Super was its combination of CPU and GPU capability. The platform includes an NVIDIA Ampere GPU with 1024 CUDA cores and 32 Tensor Cores, along with a 6-core Arm Cortex-A78AE processor. This combination allows the device to perform the parallel computations needed for machine learning inference while still supporting the rest of the software stack needed for camera capture, communication, and system control. Since the project depends on detecting moving objects with low latency, this level of processing capability was necessary to make the camera-based detection system practical.

The platform's memory and bandwidth were also important. NVIDIA specifies the kit with 8 GB of LPDDR5 memory and 102 GB/s of memory bandwidth, which supports high-throughput AI and vision applications better than lighter embedded boards. This matters because the project does not simply classify a single image at a time; it must continuously process incoming frames while keeping up with the timing requirements of a real-time response system. The Orin Nano therefore serves as the computational center of the project, allowing the vision system to run at a level that would be difficult to achieve on a lower-power alternative.

Another advantage of the Jetson platform is that it is designed specifically for edge AI development. It supports Linux-based development, GPU-accelerated inference, and common computer vision workflows, which made it easier to integrate the selected camera and object detection software into the broader system. Overall, the Orin Nano Super was chosen because it offers the level of embedded AI performance required for the project while remaining compact enough to integrate into the final system.

C. Camera

The camera is necessary for object detection and is the first line of defense when detecting whether an object has crossed the threshold of our device. The ELP 170 degree camera [3] was ultimately selected due to its high frames per second, USB connection, and field of view that allowed for detection of the entire flank of the users when placed in an adequate position.

D. Laser

The laser is one of the most important output components in the P.U.T.T.I. device because it provides immediate visual confirmation that the system has successfully detected and targeted an object. Without the laser, it would be difficult to demonstrate that the device is not only recognizing the target, but also directing its response toward the correct location. For this reason, the laser functions as both a demonstration tool and a diagnostic aid during testing and calibration.

The selected module was the Adafruit 1054 laser [4], which emits a visible red beam at a wavelength of 650 nm and operates at 5 mW. These characteristics made it a practical choice because the beam is clearly visible indoors while still being much more suitable for controlled academic testing than a higher-power alternative. The visible wavelength is especially important because it allows observers to quickly determine whether the system is pointing toward the expected target, making the physical response of the device easy to interpret.

The laser also complements the buzzer by providing spatial feedback rather than only an audible alert. While the buzzer indicates that a target event has occurred, the laser shows where the system is aiming. This combination makes the device response clearer during operation, since users can both hear that the system reacted and visually confirm whether the projected point aligns with the target. In addition, the laser is useful during tuning because any consistent aiming offset can help reveal mechanical or control errors in the pan-tilt system.

Another reason the laser is valuable is that it ties the detection pipeline directly to a visible physical action. Since the broader project depends on integrating sensing, computation, and actuation, the laser provides a straightforward way to verify that these stages are functioning together. Overall, the selected laser offers a practical combination of visibility, simplicity, and suitability for indoor testing, making it an important part of the final system response.

E. Temperature/Humidity Sensor

The temperature and humidity sensor is included in the P.U.T.T.I. device to monitor environmental conditions that may affect system reliability and operation. Although the primary function of the project is object detection and response, the device is intended to operate in a variety of indoor environments with different temperature and humidity conditions. These conditions matter because some components, particularly the radar-related electronics and surrounding hardware, may not perform ideally outside a reasonable operating range. By measuring temperature and humidity during operation, the system gains an added layer of environmental awareness.

The AM2320 [5] was selected because it combines temperature and humidity sensing into a single digital device and communicates over I2C, which made it easy to integrate with the ESP32. This was a practical advantage because the project already includes several other peripherals, so a sensor requiring minimal wiring and no analog signal conditioning was preferred. The AM2320 is also marketed as having been factory calibrated and designed for standard digital output, which made it suitable for straightforward status monitoring rather than requiring a more complicated measurement chain.

A major benefit of this sensor is that it allows environmental status to be communicated directly to the user. The measured values can be displayed on the LCD screen, giving immediate information about the operating conditions of the device. If the readings exceed the intended range for safe or reliable operation, the system can issue an alert. This improves the usability of the device by showing that environmental conditions are considered alongside the primary detection functions rather than being ignored.

The AM2320 also fits well into the overall hardware architecture because it adds useful monitoring without significantly increasing system complexity. The microcontroller can read the sensor periodically while continuing to manage the rest of the peripherals, making environmental sensing a lightweight but valuable addition to the project. Overall, the sensor was chosen because it provides a simple digital way to monitor conditions that may influence the performance of the broader system.

F. Microwave Sensor

The XM125 radar modules [6] are used to provide motion-based sensing information that complements the camera system. While the camera and vision pipeline identify the type and image location of a target, the radar modules provide additional awareness of object presence and movement within the surrounding space. This is especially important in a system intended to react to moving targets, since motion information can support early detection and improve the quality of the overall response.

Three XM125 modules are used in the design to improve coverage across the monitored region. The XM125 platform is based on 60 GHz radar technology and includes onboard processing hardware, specifically a 32-bit ARM Cortex-M4 microcontroller running at 80 MHz. This was an advantage because it made the modules more practical to integrate as smart supporting sensors rather than as raw devices requiring the main processors to handle all interpretation directly. Acconeer also describes the XM125 family as supporting ranges up to 20 meters, which is well beyond the range needed for the project and therefore provides substantial sensing margin.

The radar modules are valuable because they provide a sensing method fundamentally different from computer vision. Vision-based detection can be affected by poor lighting, lens distortion, and motion blur, whereas radar-based sensing is much less dependent on image quality and instead focuses on object presence and movement. This makes the radar system a strong complement to the camera subsystem. When used together, the radar and camera provide the project with a stronger basis for determining whether an object is entering the monitored area and how the rest of the system should respond.

The XM125 modules are particularly helpful for tracking motion trends rather than only visual appearance. In situations where the target is moving quickly or where the image may not yet provide the most stable information, radar can still contribute useful supporting data. By incorporating the XM125 modules, the project benefits from an additional sensing layer that improves robustness and supports the broader targeting process

G. Servo

The servos are responsible for physically orienting the P.U.T.T.I. device toward the detected target. Once the sensing and computation subsystems determine the approximate direction of the object, the servos convert that information into motion by rotating the platform as needed. This makes them one of the most important actuation components in the system, since accurate detection alone would not be useful unless the device could also move toward the target in a controlled manner.

The project uses a two-axis pan-tilt arrangement, with one servo controlling horizontal motion and the other controlling vertical motion. This configuration allows the device to aim across a broad region of space rather than remaining fixed in one direction. The selected INJORA [7] servos were chosen largely because of their wide 270-degree range of motion, which increases the physical area that the device can access and makes the actuator less likely to become the limiting factor in system coverage. This was especially important because the project pairs the motion platform with a wide-angle sensing setup.

Torque was another major reason for the selection. The INJORA INJS135 servo is rated up to 35 kg·cm, which is important for moving the mounted assembly with adequate stability and control. Since the mechanism must support the aiming hardware and continue to reposition during operation, the servos needed enough strength to move the payload without sluggishness or instability. A weaker servo could introduce lag or inconsistent behavior, which would reduce the visible accuracy of the full system. This made high torque a necessary design requirement rather than simply an added benefit.

Servo speed was also important because the project depends on timely physical response after target detection. The selected servo is rated at approximately 0.14 s per 60° at 6 V and 0.11 s per 60° at 7.4 V, which supports reasonably fast aiming corrections. This matters because the system must react to moving targets rather than static ones. If the servo response is too slow, then the platform may still be moving into position after the target has already shifted, reducing the usefulness of the detection pipeline. The chosen servo therefore supports both broad repositioning and repeated small corrections during tracking.

Fine motion control was another important consideration. The project does not only require large, sweeping movements, but also small positional adjustments to keep the laser and platform aligned with a moving object. If the servos move too coarsely or inconsistently, then even correct targeting calculations may not translate into accurate physical pointing. For this reason, the servo subsystem is a critical link between the output of the sensing system and the visible response of the device.

Overall, the selected servos provide the combination of range, torque, and response speed needed for the pan-tilt mechanism to function effectively. By enabling controlled two-axis motion, they allow the project to move from passive sensing into active target response, making them one of the most important components in the system.

H. Level Shifter

The SN74AHCT125N [8] was included in the design to perform logic-level shifting for the SPI LCD interface. Since the LCD expects 5 V logic while the ESP32 operates at 3.3 V logic, a level-shifting stage was needed to ensure reliable communication between the controller and the display. The SN74AHCT125N is a quad non-inverting buffer with 3-state outputs, and TI specifies it for a 4.5 V to 5.5 V supply with TTL-compatible inputs. This TTL-compatible input behavior is important because it allows a 3.3 V high signal from the ESP32 to be recognized correctly while the chip is powered at 5 V, producing a proper 5 V output level for the LCD. In this project, that made the device a practical solution for shifting the SPI control lines upward without requiring a more complex bidirectional converter.

I. I2C Multiplexer

The TCA9548APWR [9] was used in the I2C portion of the design because the project connects three XM125 radar modules that share the same I2C address. Without additional hardware, identical slave addresses on the same bus would create an address conflict and prevent independent communication with each radar module. The TCA9548A solves this problem by acting as an 8-channel I2C switch, allowing the master to select which downstream channel is active at a given time. TI specifies that channel selection is controlled through the I2C bus itself, and the device powers up with all channels deselected, which helps keep communication predictable during startup.

This part was a strong fit for the project because it allowed the three radar modules to remain on the same overall bus while still being accessed individually. In addition, the TCA9548A supports an operating range of 1.65 V to 5.5 V and includes an active-low reset input, making it practical for mixed-voltage embedded systems and easier to recover from bus issues if needed. Although the device supports up to eight channels, only three are required in this design, leaving room for future expansion if additional I2C devices are added later. In the context of the P.U.T.T.I. system, the TCA9548APWR is therefore an important enabling component because it allows multiple same-address radar modules to coexist within a single coordinated sensing architecture.

Table 1 - Power Table

Component	Supply Voltage	Maximum Current Draw	Maximum Power
MCU	3.3V	300mA	1.2W
Servos	4.8V ~ 8.4V	4A	25W
Microwave Sensors	1.8V	10mA	3.12mW
SBC	19V	2.37A	45W
LCD	5V	1.2mA	8.25mW
Camera	5V	230mA	1.15W
Humidity Sensor	3.1V ~ 5.5V	1.5mA	4.75mW
Temperature Sensor	3.0V ~ 5.5V	1.34mA	18mW
Buzzer	2.0V ~ 5.0V	30mA	90mW
Laser	2.8V ~ 5.2V	25mA	5mW

The voltage supply for the device will come from two sources. 3S LiPo battery supplying 11.1V and an 8S LipO battery that will supply 29.6V just to the SBC. The total power consumption of the device will be around 72.5W.

III. SOFTWARE

The P.U.T.T.I. device uses computer vision to first detect trained targets before setting the rest of the components in motion. While the device employs microwave modules to detect motion, it can be hard to use these modules to get clear definitions of the type of object detected. Using simpler methods such as color search or shape search could lead to the targeting of unintended objects, an especially dangerous idea if, say, spherical objects were to be detected and the device locked onto a human head. As a result, a more robust machine learning approach was required for reliable object identification.

In this project, computer vision does more than simply indicate whether something is present in front of the camera. The system must determine whether the object is one of the intended targets and approximately where it is located in the frame. This is important because later stages of the project depend on both target confirmation and target position. Unlike a simple motion detector, the vision system must perform recognition and localization at the same time while operating in real time.

This decision comes with tradeoffs as well. Lighting, distortion, and motion blur from the camera could all affect the object detection. Poor lighting may turn the entire image dark, preventing objects from being detected. Distortion from a wide-angle lens may prevent the targets from being detected near the edges. Motion blur may further distort the image and keep the object from maintaining its true look. However, the scope of the project would keep these tradeoffs to a minimum. Near-ideal rooms would be selected, and motion blur would be minimized with low-velocity targets. Even with these project constraints, these image-quality effects remained important design concerns when selecting and training the detection model.

While computer vision serves as the first stage of detection, the P.U.T.T.I. device also incorporates microwave modules as a second sensing stage. In the intended operation of the system, computer vision first identifies whether a valid target has entered the camera's field of view and estimates its location within the image. Once that initial identification has been made, the microwave modules provide additional motion-based sensing information to support the response of the device. This two-stage approach is useful because the camera is well suited for classification and localization, while the microwave sensors are better suited for recognizing motion behavior and helping track object movement after detection has already begun.

During this second stage, computer vision and microwave sensing work in tandem rather than independently. The camera continues to confirm the type and approximate image position of the object, while the microwave modules contribute supporting information about movement through the monitored area. This combined approach improves overall system awareness by allowing each sensing method to contribute where it is strongest. Computer vision provides the detail needed to distinguish intended targets from unintended objects, and the microwave modules provide additional motion information that can help the system respond more effectively once the object is already in motion.

Using both sensing methods also improves the practicality of the design. Computer vision alone can be affected by lighting, blur, and lens distortion, while microwave sensing alone may lack the detail required for safe target identification. By assigning computer vision to the first detection stage and then allowing computer vision and microwave sensing to work together in the second stage, the system gains both reliable identification and improved motion awareness. This layered sensing strategy therefore strengthens the overall detection and response pipeline of the P.U.T.T.I. device.

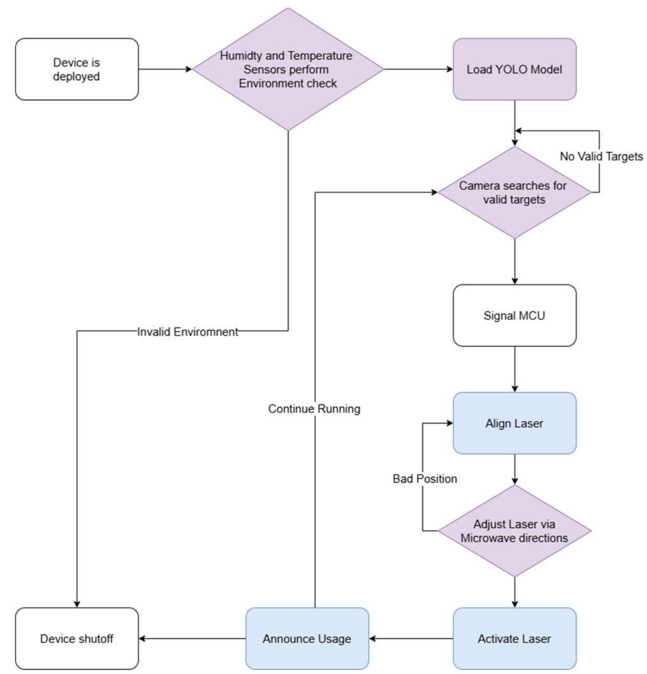


Fig. 1. General software flowchart showing how the code will overall run. The radars and lasers are not engaged until the computer vision

A. YOLO

As previously mentioned, a computer vision model capable of complex decisions was desired. The YOLO model created by Ultralytics was employed to perform this object detection [10]. YOLO stands for "You Only Look Once." It performs a single forward pass over the input image to predict object locations and class labels. It is a real-time object detection model capable of performing at high speeds while maintaining relatively high accuracy. This made YOLO especially suitable for a latency-sensitive embedded system in which detection speed was as important as detection accuracy. The model can also be custom trained on an individual dataset. This was optimal for the project because it allowed the target class to be defined specifically for the selected playground balls while also improving robustness to the project camera's characteristics.

YOLO was also preferred because it offers a practical middle ground between traditional image-processing methods and heavier detection architectures. Simpler methods such as contour detection, color thresholding, or shape filtering may be computationally inexpensive, but they are more likely to fail under changes in lighting, background clutter, blur, or target orientation. More computationally expensive deep learning models may offer strong performance, but they may not be practical for real-time embedded deployment. YOLO therefore represented a strong balance between detection capability, training flexibility, and runtime efficiency.

Another major advantage of YOLO is that it detects and localizes objects simultaneously. This is important because the project does not only require determining whether a target exists, but also where it is located in the camera image. That spatial information can then be used by later stages of the system to guide response behavior. For this reason, YOLO was well aligned with both the project's detection requirements and its real-time constraints.

B. YOLO *Dataset Collection*

The YOLO model used was custom trained on a set of green playground balls, which fit the scope of the target size. Green playground balls were selected because they were visually distinct, consistently shaped, and representative of the target size and range. They would vary in size and have differing logos to show that the model works for multiple types of green playground balls. By selecting several similar but non-identical examples, the intent was to train the model to recognize the broader target category rather than memorize one exact object appearance.

Training the YOLO model required a collection of a dataset. Once the dataset was collected, the images needed bounding boxes drawn over the playground balls to label the images. These images and labels were then sorted into a training directory, a validation directory, and a testing directory. Finally, a YOLO command could be run to train the dataset over the specified parameters. This split was important because it allowed the model to learn from one portion of the data while being evaluated on separate images that were not directly used during parameter updates.

Images were collected using the P.U.T.T.I.'s camera. The images included some still photos of the playground balls at different angles and at different locations in the image to show the base shape of the balls. These images were randomly scattered throughout the training, validation, and testing directories. More images were collected during different sessions from one angle at a time with the balls being thrown from different points. Numerous sessions were used, with the number of images collected per session ranging from 75 to 200 on average. These motion sessions were especially valuable because they reflected the actual operating condition of the project more closely than still photographs alone.

Additionally, negative images were collected to identify what objects were not these targets, and these negative images were scattered throughout the dataset. This helped reduce the chance that the model would treat unrelated scene content as a valid target. The dataset therefore attempted to expose the model not only to correct target examples, but also to challenging non-target scenes. This distinction was especially important because the project required safe discrimination between valid targets and unintended objects.

The variation in dataset images was also important for avoiding overfitting. If all images had been collected from one angle, in one room, or under one lighting condition, the model might have performed well only in that narrow environment. By using multiple sessions, different ball placements, different throw paths, and negative examples, the dataset better represented the practical conditions expected during demonstration.

C. YOLO Training

The training of the YOLO model was performed using the YOLOv8n model. The (n)ano size allowed for faster runtime during demonstration on smaller hardware. The model was trained to run at 640 × 640 size over 100 epochs. A 640 × 640 image size was sufficient for the project, as sizes larger than this would result in possible errors with the GPU. Additionally, the size was not too constricting, as the camera was operating at 1280 × 960 resolution.

The selection of YOLOv8n reflects an important design tradeoff. Larger YOLO variants may improve detection accuracy in some cases, but they also require more memory and increase inference time. Since the project's success depended not only on model quality but also on real-time operation, the lightweight nano version was a more practical choice. In a proof-of-concept system, a slightly smaller model that can run consistently is more useful than a larger model that cannot be deployed reliably on the available hardware.

Including in-motion ball images significantly improved detection performance during live testing and offered the model insight into real detection scenarios. This was important because a model trained only on clean, stationary examples may not perform well once the target begins moving. The training process therefore benefited from data that included motion blur, shifting target position, and real scene conditions similar to those expected during system use.

D. YOLO Performance Metrics

The model, as of current, is demonstrating strong performance. After 100 epochs, the model reached a mean average precision of 0.99479. The confusion matrices generated by the validation and testing results showed a 99.7% true positive rate. Using the images from the dataset, it showed high accuracy. These results indicate that the model was able to both classify the target correctly and place accurate bounding boxes around it under the conditions represented by the dataset.

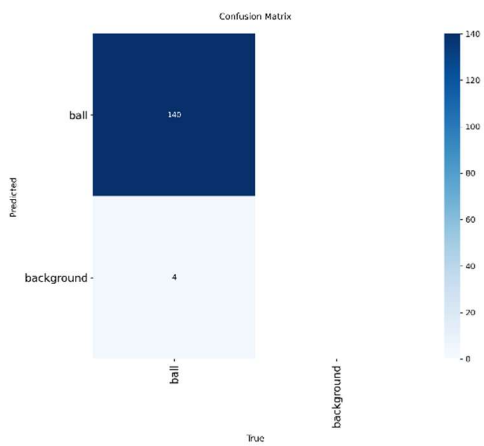


Fig. 2. Confusion matrix automatically generated during training validating the YOLO model. 140 occurrences were correctly detected as a ball, while 4 occurrences were not detected.

Mean average precision is a useful metric because it summarizes detection quality across confidence thresholds and localization performance. In a system such as P.U.T.T.I., this is valuable because the detector must do more than simply identify the presence of a class. It must also estimate the target's position in the image with enough accuracy to support later system actions. Likewise, the confusion matrix helps show whether the model is properly distinguishing the target from non-target content rather than merely finding motion or round shapes in general.

Additional testing in different areas showed that these results were not limited to offline evaluation metrics. The balls were being detected with a strong detection rate in different test areas, including near the edges of the camera. Edge-of-frame detection was especially important because the camera's field of view slightly exceeds that of the microwave sensing region, allowing early visual detection before the target fully enters the microwave's effective coverage area. During these tests, the different balls were thrown from off camera into the camera's field of vision. Of these throws, the balls were detected in 20 out of 20 sessions, showing a high detection rate in the tested rooms. The balls were detected fairly early once entering the field of vision as well, so performance appears promising, though testing in other environments must still occur.

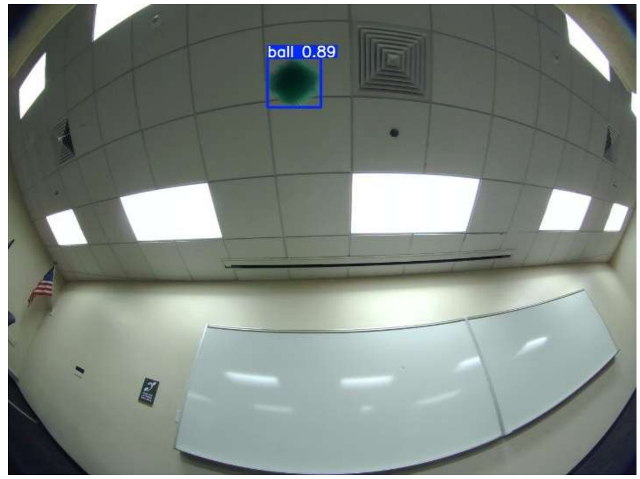


Fig.3. One of the playground balls was detected on the wide-angle camera mid-air. The model had a confidence of 89%, which is reasonably high. The remaining images captured during the rest of the throw showed similar results.

E. YOLO Limitations and Future Improvements

Because the P.U.T.T.I. does not have an extensive array of sensors, several limitations remain, especially within the computer vision itself. The dataset was limited in both size and environmental diversity, which means model performance may decrease in rooms with lighting conditions, backgrounds, or object appearances that differ substantially from those used during training. Small objects below the intended size range may also be more difficult to detect, especially at longer distances. In addition, wide-angle lens distortion and motion blur remain important challenges that can reduce confidence or accuracy in difficult frames. Since the system was tuned to match the project scope, it should not yet be assumed that the current model would generalize to all indoor environments without additional retraining and testing.

Several improvements could strengthen the system in future work. Expanding the dataset with more environments, target variations, and lighting conditions would likely improve robustness. Higher-performance hardware and improved camera performance could also increase inference speed and reduce image artifacts. Additional testing with different target sizes and ranges would help define the operational limits of the system more clearly. Future work could also compare multiple YOLO model sizes to determine whether a better balance between speed and accuracy exists for the final hardware platform.

Finally, future versions could incorporate additional sensors, improved tracking logic, or more advanced models to improve reliability beyond the classroom demonstration setting. In this way, the present model provides a strong starting point while also leaving a clear path for future refinement.

F. Microwave Sensor Software

The software for the XM125 radar module is built around the Acconeer Software Development Kit (SDK). For the purposes of this project, the Sparse IQ Service within the SDK is particularly important, as it is specifically designed for motion detection and supports velocity estimation through Doppler-based signal analysis.

Within this framework, radar data is acquired in the form of discrete sweeps, which are then grouped into frames. The Sparse Service optimizes data gathering by sampling the reflected signals at specific intervals, which is then processed using frequency-domain techniques, namely a Fast Fourier Transform (FFT), to extract Doppler frequency shifts. These frequency shifts are directly proportional to the velocity of the detected object, enabling accurate speed estimation relative to the radar module.

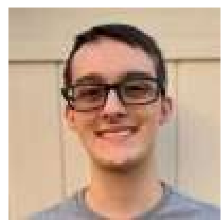
Firmware development is conducted using STM32CubeIDE, which is capable of using SDK provided project configurations and libraries specified for object detection. The development process involves configuring radar parameters, including sweep rate, frame rate, and sample rate. Once processed, the resulting velocity and distance measurements are transmitted to the external ESP32 via I2C, which can then be compared to data from the camera and other radar sensors to determine the projectile's exact trajectory.

BIOGRAPHY

Jake Tattersall will graduate from the University of Central Florida with honors to receive a Bachelor of Science in Computer Engineering in May of 2026. He previously interned for Lockheed Martin in the College Work Experience Program. Shortly after graduation, he plans to resume working on the team full-time.

Callum Fuller is a 21-year-old graduating Electrical Engineering student who will be graduating from the University of Central Florida with honors. Callum hopes to enter the theme park industry working for one of the major theme park companies in Florida such as The Walt Disney Company or Universal.

Alex Evison is graduating with an Electrical Engineering degree with two additional minors in Intelligent Robotic Systems and mathematics. He plans to pursue further education by entering the master's program at the University of Central Florida.



Nathan Kimmel is graduating with an Electrical Engineering degree from the University of Central Florida with two additional minors in Intelligent Robotic Systems and Chemistry. He plans to spend some time in industry, then pursue a career in academia.

ACKNOLEDGEMENT

We would like to acknowledge the help and advice received from Dr. Sonali Das, Dr Swaminathan Rajaraman, Dr. Xun Gong, and Dr. Saleem Sahawneh.

Additionally, we would like to acknowledge all the professors and peers that have allowed us to flourish and help us move forward with the design and development of the P.U.T.T.I.

REFERENCES

[1] ESPRESSIF documentation, https://documentation.espressif.com/esp32-wroom-32_datasheet_en.pdf?utm_source.

[2] "Jetson Orin Nano developer KIT User Guide - Hardware Specs," NVIDIA Developer, https://developer.nvidia.com/embedded/learn/jetson-orin-nano-devkit-user-guide/hardware_spec.html

[3] "Amazon.com : ELP 170degree Fisheye Lens USB camera with 1080p HD resolution, OV2710 CMOS USB camera module 2 megapixel high speed 100fps wide angle mini camera module USB2.0 for

robot,ATM,Kiosk,Eye Tracking Device : Electronics," Amazon.com : ELP 170degree Fisheye Lens USB Camera with 1080p Hd Resolution, OV2710 CMOS USB Camera Module 2 Megapixel High Speed 100fps Wide angle Mini Camera Module USB2.0 for Robot,ATM,Kiosk,Eye Tracking Device : Electronics, https://a.co/d/enpHHCu.

[4] 1054 Adafruit Industries LLC | optoelectronics | DigiKey, https://www.digikey.com/en/products/detail/adafruit-industries-llc/1054/5629439.

[5] 3721 Adafruit Industries LLC | Sensors, transducers | DigiKey, https://www.digikey.com/product-detail/en/adafruit-industries-llc/3721/1528-2564-ND/8558266.

[6] XM125 Acconeer ab | sensors, transducers | DigiKey, https://www.digikey.com/en/products/detail/acconeer-ab/XM125/17883267.

[7] "INJORA INJS235 35KG Waterproof Brushless Servo Large Torque Digital Servo," *INJORA*. Available: https://www.injora.com/products/injora-injs235-35kg-waterproof-brushless-servo-large-torque-digital-servo

[8] Ti, https://www.ti.com/lit/ds/symlink/sn74ahct125.pdf?ts=1728497791408&amp=&ref_url=https://www.ti.com/product/SN74AHCT125

[9] Ti, https://www.ti.com/lit/ds/symlink/tca9548a-q1.pdf?HQS=dis-dk-null-digikeymode-dsf-pf-null-wwe&ts=1746590777638&ref_url=https://www.ti.com/general/docs/suppproductinfo.tsp?distId%3D10%26gotoUrl%3Dhttps://www.ti.com/lit/gpn/tca9548a-q1

[10] Ultralytics, "Train," Ultralytics YOLO Docs, https://docs.ultralytics.com/modes/train/