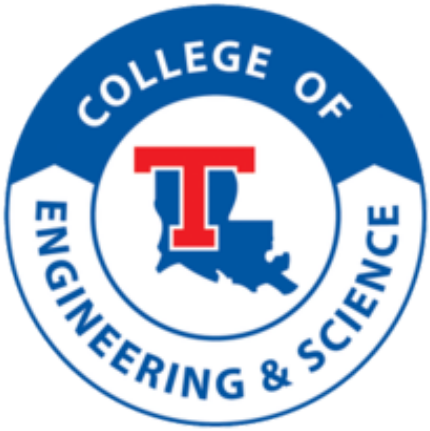


Substation Intrusion Detection System

Justin Bramlett

Hebane Guehi

Ethan Roberts

Reese Seals

Advised by:

Dr. Matthew Hartmann

Dr. Miguel Gates

May 14th, 2025

Abstract

The Substation Intrusion Detection System is an advanced security system that aims to enhance the protection and reduce unwanted intrusion of electrical substations - critical facilities that control and monitor the electricity supplied to communities, homes, and businesses. Our system synthesizes the features of machine learning, network tunneling, protective firewalls, infrared cameras, triangulation, and data collecting to create a product that can be implemented into any pre-existing security system. This innovative system strives to provide 24/7 comprehensive surveillance security and threat-detection capabilities using YOLO (You Only Look Once) for object detection, camera triangulation (Stereo Vision) for distance calculations, and virtual zones using PyTorch to establish a relationship between object boundary boxes and zone coordinates. Two OV5647 5MP camera modules are used to calculate the distance of an intruder as well as build redundancy by only displaying threats if both cameras detect a person that exceeds a confidence score of 0.5. The Substation Intrusion Detection System aims to collect and send this data securely through tunneling by establishing public and private keys via WireGuard. By utilizing these various technologies, our system's goal is to ensure that electrical substations remain secure by minimizing the risk of damage and loss of power to communities.

Table of Contents

List of Figures

List of Tables

Engineering Design Specification

Project Background

The Substation Intrusion Detection System aims to act as a deterrence and provide information regarding the security of power distribution substations. Substations provide electricity to millions of people and they must perform well and are under constant surveillance. On April 16, 2013, in San Jose California, there was a devastating attack at a substation. For 20 minutes gunmen hit the cooling fins of transformers causing 17 of 21 of them to overheat and stop working. PG&E luckily was able to reroute power in time but Jon Wellinghoff states that this incident “could’ve brought down all of Silicon Valley” [9]. Wellinghoff goes on to state that if there were targeted terrorist attacks on certain substations taking down less than 20 would knock out power for the entire country [17]. Not only are physical attacks an issue, but so are cyber-attacks. A recent cyber attack in 2021 on the Colonial Pipeline raised questions about security issues on American infrastructure. While this wasn’t specifically an attack on the electrical grid, this event could just as easily happen to the electrical grid. Darkside was the one that launched this ransom attack demanding 4.4 million dollars in compensation. [9]

Security for substations currently involves fencing, surveillance cameras, motion detectors, multi-factor authentication, data encryption, etc. [14]. These measures are typical for high-priority substations and lower-priority substations may have very little to no security. The Substation Intrusion Detection System aims to use security measures already in place and improve on them.

Project Objectives & Scope

The Substation Intrusion Detection System is not designed to prevent attacks but to act as a deterrent and quickly provide information to security personnel. The main aspects of the project include object recognition to detect people, triangulation to pinpoint the location of intruders, and provide a secure way to transmit the information to a host device. The objective of implementing these things is to allow a computer to do much of the detection to take out the human error aspect. The object recognition is in place so humans will be the only ones detected by the system. Triangulation of cameras is put in place for two reasons. The first one is so zones can be implemented for the security system. Some substations are near roads, sidewalks, or neighborhoods. The triangulation can detect how far away a person is so the system won't go off every time a person is present. The second reason is to build redundancy with object recognition. Sometimes glares or random objects, especially at night will cause false positives with the system. If the system has two cameras that means both cameras will need to detect a person to perform triangulation. Finally, digital security measures will be put in place to ensure that there is a secure connection between the cameras and the device that is streaming the video feed.

To ensure the system is running as intended, different tests will be carried out to take into consideration the performance of the system. The object recognition will go on three different tests. The first one is pictured, then a video, and finally a camera feed. Giving pictures to the object algorithm can ensure that it can accurately detect different people depending on height, age, race, sex, etc. This will also give an idea of the algorithms' weak points when it comes to angles and lighting. Next, a video feed will be given to the algorithm. The point of this stage is to

ensure the algorithm can track people as they move across the frame. Finally, it will be given a camera feed. This stage will ensure that the algorithm can continuously run with a life feed that it has never seen before. For the triangulation, it will be tested on if it activates based on a given threshold and with accuracy on distance. Both cameras will need to detect a person that exceeds a confidence rating given by the user. If the triangulation runs after the threshold is reached by both cameras it is running as intended. Next, the distance calculated by the triangulation will be tested. A person will stand at a known distance, for example, 5 feet, and then it will be compared to the distance calculated by the triangulation algorithm. Finally, the secure connection will be tested across different networks to make sure a stable connection is met.

Specific Aims

The aim of the project is to develop a machine that can assist in intrusion detection and build up security for substations. This will be done by using object recognition to detect people, create a virtual coordinate system with stereo vision and virtual zones, have a control system that will send warnings, and establish a tunnel to safely transmit that information to the control room.

This system can be broken down into several subsystems such as an object detection system, stereo camera system, triangulation system, virtual zones, alerting system, VPN integration, and graphical user interface (GUI). The stereo camera system allows for the system to establish a 3D virtual coordinate plane to track the distance and location of potential intruders. Object recognition via YOLO assists in filtering out unwanted detections. The virtual zones will allow the user to mark off unwanted areas that will then tell the alert system when to activate depending on the conditions. In order to transmit this data from a remote location to a control

center safely the camera feeds will be tunneled with WireGuard to prevent tampering from unauthorized guests. Finally, the GUI implementation gives the end user a way to interact with the system, check for false positives, and troubleshoot accordingly.

In order to test the validity of this system separate tests were done to troubleshoot different parts of the system. Real vs. machine distances were recorded to account for errors. Following this "intruders" stood inside the virtual zones to verify that the alert system would respond. Latency was tested to see how the different algorithms slowed down the stream in comparison to a raw video feed. The alerting system then runs through a separate thread to ensure video delays will not prevent delays in the logic.

Project Deliverables & Requirements

The Substation Intrusion Detection System is designed to detect intruders that come close to the substation. The system is designed to perform outdoors meaning it will have UV protection, water resistance, and temperature resistance. To ensure protection from UV radiation and water an acrylic spray will be used to fill gaps to prevent water from entering and protect the material from UV radiation. For temperature resistance, neoprene material will be used. Neoprene material has a thermal conductivity rating of 0.05 W/(m*K) indicating it is very resistant to changes in temperature [13]. This material is often used in scuba diving applications allowing divers to swim for hours at close to freezing temperatures. The Raspberry Pi module can operate between temperatures of 0-85 degrees Celsius. The Raspberry Pi 4 module has dimensions of 88 mm (3.46 inches) x 58 mm (2.28 inches) x 19.5 mm (0.77 inches). The system enclosure must house two of these with camera modules. The estimated dimensions of the enclosure will be

about 4 inches x 5 inches x 1.5 inches to leave room for components, insulation, and for maintenance purposes. Some considerations regarding camera placement are still in progress. More information will be given under Technical Details.

The load capacity of our system is intended to draw a maximum of about 5 Amps. Two Raspberry Pi 4 modules working at full capacity draw about 2.5 Amps each. Two Raspberry Pi 4 Day/Night camera modules will be used drawing about 100 mA each. To power the system two SugarS Plus rechargeable battery pack modules will be used. Each battery pack has a 5000mAh battery life meaning the system could operate for 2 hours without an external power source.

The enclosure design also keeps in mind routine maintenance. According to NFPA 731 (Standard of the Installation of Premises Security Systems), control systems and security cameras must be oriented in a way so that personnel can do routine maintenance. It's intended to design a door that will allow ease of maintenance. By using magnets, similar to how Air pods use them, they will keep the door shut during different outdoor conditions, but it will easily allow personnel to have access to any components.

Even with these precautions, there are likely still issues that could arise with the system. With this system being very software-heavy the most likely issues will arise with wireless connectivity to the Raspberry Pi. In the event of losing connection two possible solutions are under consideration. The first one is a SSH (secure shell) protocol. SSH allows the user to open a command-line terminal on the Raspberry Pi from a different device. The process involves creating an empty file called ssh and putting it on the Raspberry Pi's SD card. All you will need after that is the Raspberry Pi's IP address and it can be controlled via computer. Another

possibility is using a dynamic DNS. This would give the Raspberry Pi a fixed domain name that could be accessed remotely even across different networks. The dynamic DNS might be a better option since IP addresses often change.

Primary Intended Use

The intended use of this project is that it will be used with security cameras at substations. The triangulation of cameras is highly dependent on the distance between the cameras that run the algorithm. If the distance between cameras is altered, then it needs to be updated in the software otherwise there will be inaccuracies. It is also recommended that the same camera modules are used for both cameras. Camera resolution, dimensions, and lens can affect the performance of the system. If two different cameras are used this will also most likely cause inaccuracies. Consistent maintenance must also be done for the system as well. If there are storms with strong winds or rain testing might need to be done to ensure the cameras are still lined up properly. Despite these considerations, the system will still run but it is up to the end user to be set up properly to make sure the algorithm runs correctly.

Predictable Unintended Use

Theoretically, our project doesn't have to be used for substation security. If users have a camera feed at any given location, they could use this system. Some setup considerations will need to be taken into account. An overhaul of CCTV cameras might need to be done. Some places have cameras of poor quality or the systems in place have different camera modules that can affect performance. To ensure that there is a stable connection between the camera modules and the

host computer, the user will require some networking knowledge or be willing to pay someone to do the setup process for them. This system is mostly software-based. So, theoretically, it could be set up for a multitude of different systems.

Special Purpose Features

The Substation Intrusion Detection System is intended to boost security for electrical substations. Current security at substations involves fencing, surveillance cameras, motion detectors, multi-factor authentication, data encryption, etc. [17]. This system will expand on that by introducing object detection and triangulation of cameras. Currently (unless some information is classified or outdated), substations do not incorporate these algorithms into their systems. Triangulation is used in some applications, such as Hawk-eye which is used in Tennis refereeing, which makes these ideas feasible to use for boosting security for electrical substations.

Constraints

When designing a system many factors come into play. Some of these constraints are factors including economic, environmental, political, safety, etc. This system needs to be affordable and ethical so companies can easily integrate it into their security. The substation intrusion detection system is designed to work for the public's benefit. So, many of these constraints are important to take into consideration.

Overall, our system is very software-heavy. Many of the aspects of our project require processing power to carry out object recognition and camera triangulation. Because of that, it is a top

priority to ensure that the processes are efficient and use as little processing power as possible so the algorithm can be run on the cheapest hardware possible. This is being simulated by Raspberry Pis taking in a camera feed and streaming it wirelessly via the internet to a laptop. The laptop will act as the host computer for a control center that will have access to the substation's security cameras. The affordable cost is also being taken into consideration through the housing that is made from acrylic plastic. Acrylic is also waterproof and UV resistant making it an affordable solution to protect the hardware from the outdoors.

Environmental impacts are also a growing concern as the world is trying to reduce the amount of CO2 emissions that are put into the atmosphere. Computer vision training has an impact on carbon emissions due to the intense processing power it has on the graphics card. For assumptions to go into a better explanation a Nvidia 2070 Super graphics card will be used and will be assumed to be running at full capacity. The Nvidia graphics card will be assumed to use 215 Watts of power in this state. The training model for this project ran for 1.5 hours and it consumed 322.5 Watts. This is the equivalent of driving a small passenger vehicle for one-third of a mile [5]. This training model generated 0.124 kg of Carbon Dioxide emissions in this short time compared to the average it generates daily of 0.107 kg of Carbon Dioxide emissions. More complex models often run for several hours or days depending on the desired outcome of the model. Apart from the training model, there are concerns about some of the materials that are used in some of the hardware. The camera modules and Raspberry Pi 4 are made from materials such as Lead, Tin, plastic, etc. that need to be disposed of properly when decommissioned. Even despite this, under normal operating conditions, there is no real potential harm to the environment.

Generally, this project is projected to have a positive social impact. The main purpose is to prevent intrusions into substations and improve the support of law enforcement by giving adequate data promptly. Object recognition is used in many security applications including home security, identity verification for smartphones, and surveillance systems which directly correspond to this project. All these applications have one thing in common, they are designed to provide security in a user-friendly manner. In control centers, there is usually someone sitting at a desk watching footage that covers several different areas. There are countless distractions that could prevent someone from performing their job and missing bizarre activities that occur on the feeds. This system is designed to always run into account for these distractions and let the user know when something goes wrong. This will help the company that provides power to communities to act quickly to ensure that they can keep providing to their customers despite the circumstances.

When combining machine learning and surveillance footage, there are ethical and political issues that need to be addressed. One of the most well-known issues is discrimination and biases in the machine learning data. With facial recognition data in technology like Apple iPhones, there are major errors that are found within the machine learning models. Studies have shown that the error rate for correctly detecting white men is about 0.8% and the error rate for detecting women of color can be as high as 34.7% [12]. In Minnesota, they have laws and regulations - the Government Data Practices Act - that allows the government to be transparent on how it collects, uses, and maintains data [3]. Many states have laws like these however they don’t include facial recognition data. To combat this the data will be obtained from an open-source platform that has a diverse amount of data to ensure that it is as ethical as possible. This project intends to use

open-source public data as well as sources where that data comes from. Much of the data for this project comes from Unsplash, which is a public platform where people upload pictures for commercial use. None of the pictures are modified in any way and they are only used as reference data. It is also important to note that privacy issues are a concern with object recognition data. Some companies have taken part in unethical practices with facial recognition and data sharing. Ring Doorbell has allowed law enforcement to access surveillance footage without a warrant [6]. It is important that the data received in this project is confidential.

In major projects, public health is also a factor that is taken into consideration. In this project, there is no immediate concern regarding public health but there are still some topics that should be looked into that could indirectly be affiliated with this system. Some public health factors could be the effect of CO2 emissions on humans and wildlife and the consequences of power outages on human health. CO2 emissions can affect water quality, air quality, and even mental health. Poor air quality and water quality can affect human health and cause premature death [12]. Attacks that occur on substations can cause outages that can cause food to go bad or have worsening impacts on sick people. Brian Terhorst is a PGE customer [2]. This company provides electric utilities to the West Coast of the United States. Terhorst suffers from Late-onset Disease which weakens his muscles to the point where he cannot even breathe on his own. Terhorst stated in situations like this he has “no means to recharge” his equipment and he is “completely dependent on electric devices to live” [11]. This outage was not caused by an attack. However, attacks are usually unexpected and can have a great impact on people such as Terhorst.

Finally, this system must be compatible with different systems. This project is targeted for substation use due to some of the things that are implemented into it. However, as long as someone has security cameras and a control computer it can be used with this algorithm. It is noted that it may be difficult for someone with limited technical experience to set this system up and operate it. This is due to the manual connection setup over a network as well as the configuration of some settings to ensure that the system is running properly. Much of this project is software-based, and any hardware can be supplemented with other alternatives. In this case, the decommission of this system should be similar regardless. Any hardware that is used should be wiped clean of data and recycled to the nearest E-Waste recycling centers. These centers will ensure that all of the data or information on a device is wiped properly for personal security reasons [8]. They will also dispose of or recycle the hardware properly to reduce the carbon footprint. Never throw away any electrical or electronic hardware without determining the proper way to recycle the product [7].

Technical Details

The Substation Intrusion Detection System is designed to aid in gathering relevant information during targeted attacks on substations to ensure that power companies can keep providing to their customers. This system will use object recognition to detect any intruders and triangulation between two cameras to detect how far an intruder is from the substation. If an intruder is approaching the substation pictures of them will be taken and an alarm system will be set off.

The purpose of camera triangulation is to build redundancy and to prevent the alarm from constantly going off near populated areas similar to the one shown in **Figure 1.**

Figure 1: Substation near roadway

Many substations are near parks, roads, or sidewalks and to prevent the alarm from going off every time a person is detected a desired distance can be set to prevent the alarm from going off. The information gathered by the cameras will be sent to a control room. Wiregaurd will be used to tunnel the information to the desired spot to keep any transmitted data from being tampered with.

Engineering Approach

The purpose of this project is to aid control room personnel in detecting unwanted intruders from entering or tampering with substation equipment. This prototype will simulate object recognition, triangulation between two cameras, an alarm system, a recording system, and a secure way of transmitting data.

Object recognition between two or more cameras will build redundancy and minimize false positives. This is because each camera will have a slightly different perspective on a given area. Bugs, lighting, weather conditions, etc. can affect the camera lens' view. With two cameras, it

makes it less likely that outside conditions will affect the performance of the object recognition algorithm. In terms of the algorithm used in this system, YOLOv8 was decided upon. YOLOv8 is a convolutional detection algorithm. This means that in one pass, it checks for the confidence that an object is detected and the IoU (intersection of union). YOLO creates a grid on a given image and finds which area contains the coordinate for the center of the boundary box of the object it is confident in. At the same time, it compares the data it is receiving to data saved in the machine learning algorithm. It then calculates the area of overlap between the boundary boxes of the saved data and data it is currently receiving (IoU) [15]. Then the ratio between these two scores will be the confidence rating score that an object is detected. This ratio is then compared to the confidence score that the user sets, which is a number between 0 and 1. If the user sets the confidence score of 0.5 and the object detected has a score of 0.5 YOLO will recognize this as a person. If multiple objects want to be detected, i.e. people, animals, etc., then a class will be assigned to the given object. In this system, only one class is used (0) people, only people in general are being detected. YOLO or algorithms similar to it are actually used in self-driving vehicles. According to Ultralytics, a research group working on object detection, there are different levels of automation within the self-driving industry ranging from level 0 which is no automation to level 5 which is full automation [16]. Currently, Mercedes-Benz is at level 3, which is conditional automation which means that the algorithm can control the vehicle and can manage most driving tasks. If the companies that are implementing self-driving features trust this algorithm it should be more than enough to handle the scope of this project.

Next, camera triangulation is used to build redundancy and prevent false positives. Camera triangulation relies on the imaging from multiple cameras so it can calculate distances. **Figure 2** shows how the distance is reliant on the camera's focal length, baseline, and camera disparity.

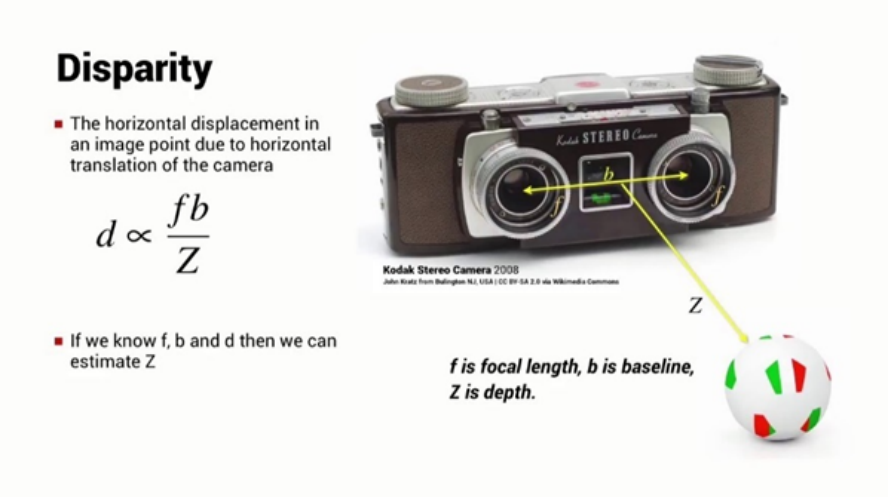


Figure 2: Lens calculations

To improve the accuracy of the algorithm with calculating the distances of people an interpolation function has been added to scale and offset distances to minimize error. The error of the system increases as the person is farther away from the system. This is because the relationship between distance and disparity is hyperbolic and the change in disparity is very small at farther distances (i.e. 30ft).

Another way to combat possible errors in the system is to add virtual zones in the system that are adjustable to accommodate the needs of the end user. The zones box off areas that relate to different actions. For this prototype by default, there are two zones: a blue and red zone. The blue zone is designed to give the person operating the cameras in the control room a warning if someone is approaching the substation while the red zone is the deterrent to scare intruders away.

To transmit this information securely from the substation to the control room, a WireGuard VPN will be used to encrypt and tunnel the data to the host. This will all be ran through a virtual machine instance from the Google Cloud Platform which will act as a relay server. This allows us to set up a tunnel to hidden devices in private networks that do not have access to their router's port-forwarding settings. The purpose of this is to forward the distance calculations, virtual zones, and other data that can be accessed by other machines.

For consistency reasons, it is important to use the same camera modules for however many cameras are being used in a given project. This is because the focal length and resolution of the cameras will be the same, making the calculations the machine has to do more accurate. For ideal cases and the scope of this project so far, the cameras will be perfectly in line to reduce the amount of math the algorithm needs to perform. The cameras do not need to be perfectly in parallel, as long as the machine is aware of the angle the cameras are at. This angle can be manually set, or it can be automatically detected if the camera has a gyroscope in it. The triangulation in this project has taken inspiration from Hawk Eye which is camera triangulation technology used in sports. Hawk Eye is especially used in tennis and it aids the referees in confirming if the ball is in or out of bounds in real time. It uses the same triangulation principles used in this project. **Figure 3** shows the basic trigonometry the algorithm uses to detect distances.

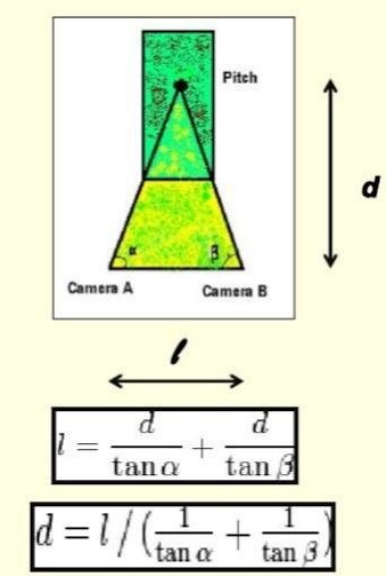


Figure 3: Hawk Eye triangulation calculations

The main difference between Hawk Eye and the Substation Intrusion Detection System is that HawkEye uses several cameras while this prototype only uses two cameras. However, the principle is the same. There is a reference camera similar to how humans have a dominant eye. And the camera disparities and distances are referenced off of that camera [1].

For the cameras' housing, many factors had to be taken into account. Because the camera system is going to be placed in an outdoor environment, our design has to consider the possible weather conditions that can occur. The cameras must be able to operate under all forms of weather in order to perform consistently and reliably. For our design, we decided that a slanted, rounded-off roof with an overhang on the top of the camera would be best. **Figure 4** represents a rough layout of the camera housing.

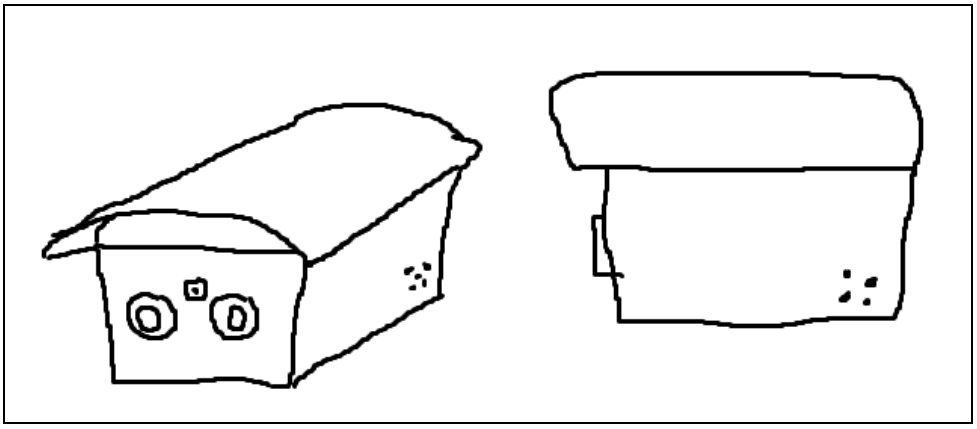


Figure 4: Sketch of camera housing

The entirety of the camera housing will be made of 3-D print plastic to prevent any UV damage to our system. The front side of the camera housing, however, will be made of clear acrylic to allow the cameras to look through it. One concern that we have is that the acrylic screen may cause a glare on the cameras. Until further testing, we are unsure if we will need to punch holes for the camera and their respective UV lights to look through. This fix will be generally easy, as we can simply punch these holes in with a milling machine using precise measurements. We will not be testing our system in various weather conditions until we feel comfortable to do so.

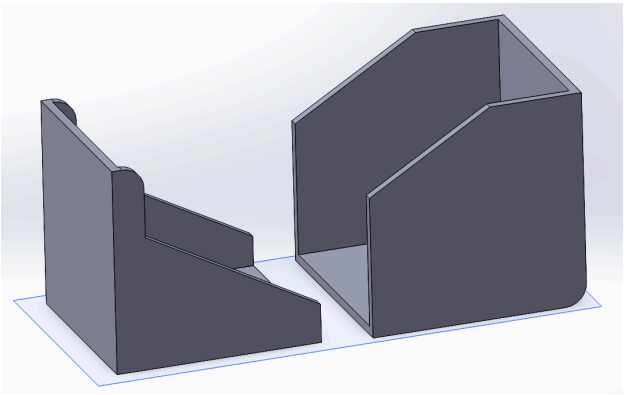


Figure 5: Rough draft of camera housing

The figure shown above, **Figure 5**, represents the rough draft of our camera housing. Initially, our team wanted the camera housing to be distinguishable from a standard camera system that you could find in any other substation (or any other location). However, this proved to be impractical as the systems that we designed were not very weatherproof. So, we ultimately decided to not stray away from the standard camera design.

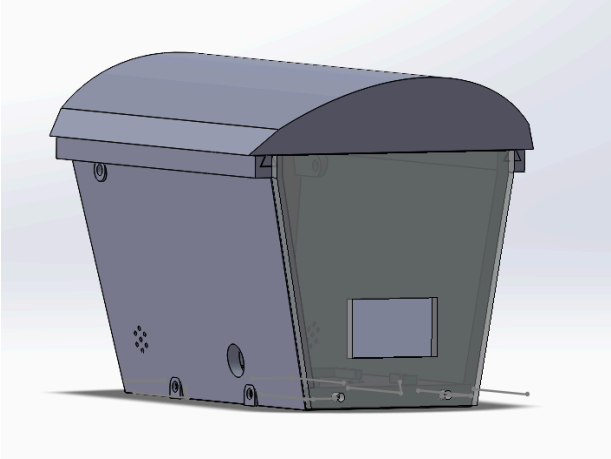


Figure 6: Updated SolidWorks Prototype

Figure 6 represents the most recent 3-D prototype that we have assembled. This system was made using SolidWorks. To prevent our system from overheating, we want our housing to be breathable. We will accomplish this by providing the interior of our systems with fans, a heatsink, as well as punching holes and lining them with a protective mesh covering. In the backside of our system, we will place speakers that will be used for the alarm system. These speakers will also have holes to project audio out of that will additionally be lined with the same protective mesh used for the system's airflow.

To effectively utilize our system's triangulation capabilities, we will need to place the two cameras apart from each other at a consistent distance. The two cameras that we will be using will both have their own protective housing. The two cameras will be placed around 1 foot apart. We will accomplish this by connecting our two cameras with an aluminum profile bar that we received during our freshman year. This will keep our system as in line as possible.

Prototype Development and Results

With the progression of this project, the process of unit testing which is the process of getting individual components or algorithms to work before integrating them together. This process is widely used in programming applications because of how quickly overwhelming it can get when trying to incorporate multiple things at once. If more than one thing is tested for the first time and something goes wrong, it makes it more difficult to figure out the error and where to start troubleshooting. This is why this method is used in the fabrication of this system.

When trying to decide what should be worked on first, a flow chart was made to figure out what parts of the system were dependent on the others. It was discussed that the heart of the system was object recognition as all the other processes will be dependent on that.

Before any of the algorithms can work the system must pick up a live camera feed. Proceeding that, the object recognition machine learning algorithm was worked on since it is the heart of the system. To start working on the object recognition algorithm data needs to be collected. For the first data set 80 personal pictures were used as reference. These were uploaded to a website called CVAT which is a free platform to use to annotate pictures so it can be recognized by the

YOLO library in Python. These annotations are converted into a text file that displays the x and y coordinates of the center of the boundary box, the height of the boundary box, the width of the boundary box, and the object class which in this case will always be zero because only people will be detected. After going through some troubleshooting errors ensuring the file directories were correct and the proper Python libraries were installed, the YOLO object algorithm ran with 80 pictures and 100 epochs.

Figure 7: Object recognition results sample

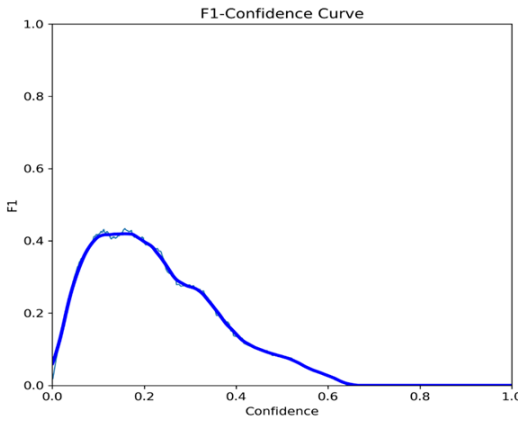


Figure 8: Confidence curve

Figure 7 shows a sample of the results and **Figure 8** shows the confidence curve after the algorithm ran. With the displayed results, it can be seen that the performance of the algorithm is not ideal. The displayed confidence scores are very low, there are duplicate boxes, and many people are not detected. The confidence curve also displays useful information. The horizontal axis is the independent confidence score. This is the score that the user will insert into the program. The vertical axis displays the confidence score ratings the machine gives based on what the desired rating is. The reason the user imported score should not be zero is because there

would be a lot of false positives and the reason the score should not be super high is because the machine would never be able to detect a person. According to the curve, the rating with the highest results is around 0.18 which gives an average confidence rating of 0.4. This is a very low rating which can cause some major problems with false positives.

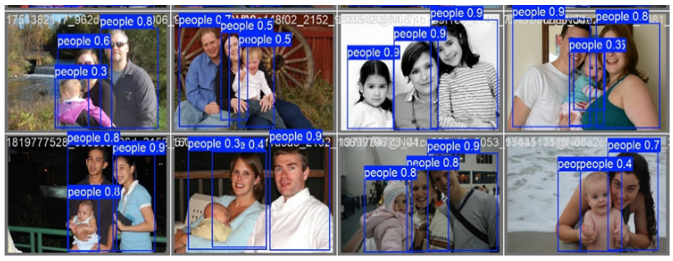


Figure 9: Updated database results

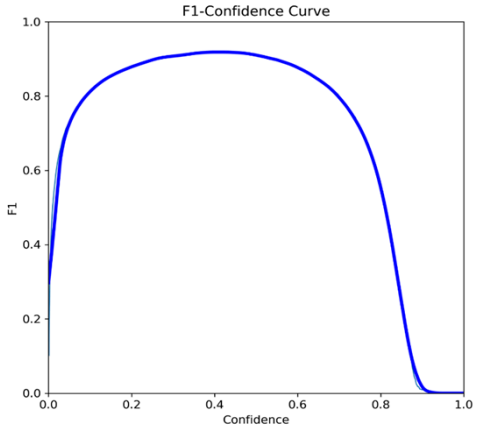


Figure 10: Updated confidence curve

Upon improvement, 750 pictures were added to the database from a free website called Unsplash. Now the database has 830 pictures that were trained with 20 epochs. **Figure 9** shows a sample of the results and **Figure 10** is the confidence curve. The results from this pass-through displayed much better results. Very few confidence scores are low, many of the scores are very high, and every person is detected. The confidence curve also displays much better results. At a user threshold of around 0.5, the machine's confidence is up to 0.9. These scores are much more practical and theoretically should display the needed results in the application it will be used for.

Next, the camera triangulation, camera implementation with the Raspberry Pis, and physical housing were in the works of being designed. First, camera triangulation was tested with webcams so the logic of how the algorithm will work can be worked on without the use of the Raspberry Pi cameras. This was decided because the implementation of the triangulation algorithm was expected to take the majority of the time, and it could not wait for the Raspberry Pi implementation. The way the triangulation is intended to work is based on the object recognition between the two cameras. Each camera detected must exceed a confidence score rating higher than the user-set score before the camera triangulation program runs. This is the building redundancy to account for different lighting and camera errors. If both cameras exceed a confidence score rating of i.e. 0.5 the main object recognition code will call on the triangulation program to determine the distance an object is from the system. Once the boundary box is drawn around a person, the coordinates of the center of the boundary box are known already so the triangulation algorithm will base the distances on those coordinates. This is to keep the tracking consistent because if it was based on off-human features, they may not be visible to all cameras and it could cause more errors. Using the center of each boundary box will be more consistent when trying to determine disparity. In the program, the first camera detected by the device will also always be the reference camera. Something to note is the two webcams have different resolutions, focal lengths, and viewing angles, and the distance between them is not perfectly set meaning currently the calculated distance is not accurate. However, the logic of the algorithm works and as someone steps away from the camera the distance will increase and vice versa. As the Raspberry Pi is implemented more testing will be done to ensure the system is accurate.

For this project, Raspberry Pi 4b is used. The newest model for both new and used is almost double the price. The 4b model is also used in the Louisiana Tech CSC 130 series, meaning for this prototype these modules can be borrowed to save money. The use of the Raspberry Pi was decided to take away processing power from the machine running the main algorithm and to simulate sending data to a control center. The goal that needs to be achieved is to have the Pi upon boot up to take in a camera feed as well as a stream that feeds over a network so the control room can have access to the feed. It needs to be able to stream on boot up to reduce set up time for demonstration and so if there is an outage in a real application it can get back online as soon as it receives power, so it doesn't have to be reconfigured every time. Since the Pi is streaming to a Windows OS it will stream the feed over VLC. To do this, the MJPEG streamer as well as VLC must be installed on the Pi. MJPEG is the format the OpenCV library in Python likes. Additionally, it must be tested that the Raspberry Pi detects the camera module which in this case is the Day-Night Vision OV5647 camera module that can stream a 1080P video feed at 30 frames per second. To automate the setup a SystemD file must be created that will detect a camera feed as well as stream it through a desired DNS. This feed can be accessed as long as the device that needs to receive the stream knows the Raspberry Pi's IP address. However, there are two issues that need to be addressed. One device's IP address changes constantly, meaning a static IP address needs to be used and the system will be demonstrated on Louisiana Tech's campus. The school's internet has security protocols meaning the IP address of the Raspberry Pi cannot be configured on the school's network. To account for this a LAN (local area network) will be set up. An external router that has its own network will be used so the Raspberry Pi can be given a static IP address. A bash script was written that will put the Raspberry Pi modules in a

mode that will have them stream the video feed over a network. The script will cause the camera feed to be streamed via VLC over a local network. This file on boot up will wait 10 seconds to ensure all of the necessary drivers start up properly as well as wait for a consistent network connection to prevent the stream from crashing.

The Raspberry Pi modules were placed inside of a 3D modeled housing with acrylic screens that allow the cameras to see as seen in **Figure 11**.

Figure 11: Final Housing

This housing houses the Raspberry Pi and alarm control system. The camera is mounted on the inside of the acrylic screen and is held up with an acrylic tape so the camera is level and stable to allow accurate readings for the algorithm. The slide roof allows for easy access to the interior of

the housing, as well as providing the weatherproof design that we are aiming for. The slanting on the side walls and the curved roof encourage the rain to run off instead of building up on the exterior of the system. In the floor of the interior, there was an extruding slot for the Raspberry Pi to sit. On the screen of our housing, the square hole was implemented to reduce glare and any lighting inconsistencies that our cameras could pick up. The opening of our screen could be smaller to optimize weatherproofing. However, we were experiencing a lot of delays during our printing process. Our first print took over 6 weeks for us to receive it. Ultimately, we did not have enough time to perfect the screen, much less the housing altogether.

When streaming data from substations to control centers, it's important to protect data through encryption and encapsulation to ensure the utmost security. Using the open-source tunneling software Using the open-source tunneling software WireGuard, data travels through a secure tunnel that encapsulates packets, protecting against wiretapping, main-in-the-middle attacks, and other intrusions. Wireguard will also encrypt the data to prevent data leaks (if somehow intercepted). The tunnel could be created through two techniques such as reverse tunneling and regular tunneling. Reverse tunneling will be an option when port-forwarding isn't possible on one side of the connection, preventing one peer from finding the other to set up a tunnel. The client hidden inside the private network will initialize the tunnel by connecting to the public endpoint and then building the tunnel from endpoint back to host, securing the path in reverse from both endpoints without the need for port forwarding. Regular tunneling will proceed where the connection is built while connecting from client to server, usually in situations where port forwarding is available or both have public IPs. Both sides will need to exchange public keys and have their own private keys, creating an asymmetric encryption. The private key is kept secret

and the public key is shared. This enables secure and authenticated communication using asymmetric encryption. These keys will be generated through Powershell commands through terminals upon request through GUI. The setup of WireGuard will also be through GUI, using Powershell to install it for the user. Configuration of endpoints, ports, addresses, etc. will be through GUI in the network connection setup. Activation and deactivation of tunnels will also be through the GUI as well as showing a list of the current active tunnels; these features will all be available on the private network page on the GUI.

Streaming of data will be through a video and audio packaging open-source software called FFmpeg. Through the setup of the 2 RPIs connected to the router which connects to the control system at the substation, the two streams of data will be sent to the control system which will then send the data through the Wireguard tunnel to the control center. The transmission of data will be through the UDP (User Datagram Protocol) to port 5004. User Datagram Protocol is a common protocol used for video-streaming, which prioritizes speed and efficiency. Ensuring efficient delivery of packets of video feed, prevents packet loss or delay of packets. Through the use of its low overhead through simple headers and lack of control mechanisms, it's highly suitable for real time applications and makes it lightweight. These delays could result in errors in triangulation if it occurs too frequently, which is why our main focus was minimizing such cases over other factors in video streaming.

In order for the end user to make adjustments with little to no experience, we designed our GUI to be highly user friendly. There are many different pages that allows the user to activate

WireGuard, configure cameras, view network logs, view recordings, and calibrate virtual zones. The GUI even has a dark mode.

Figure 12 is the display for setting up Wireguard.

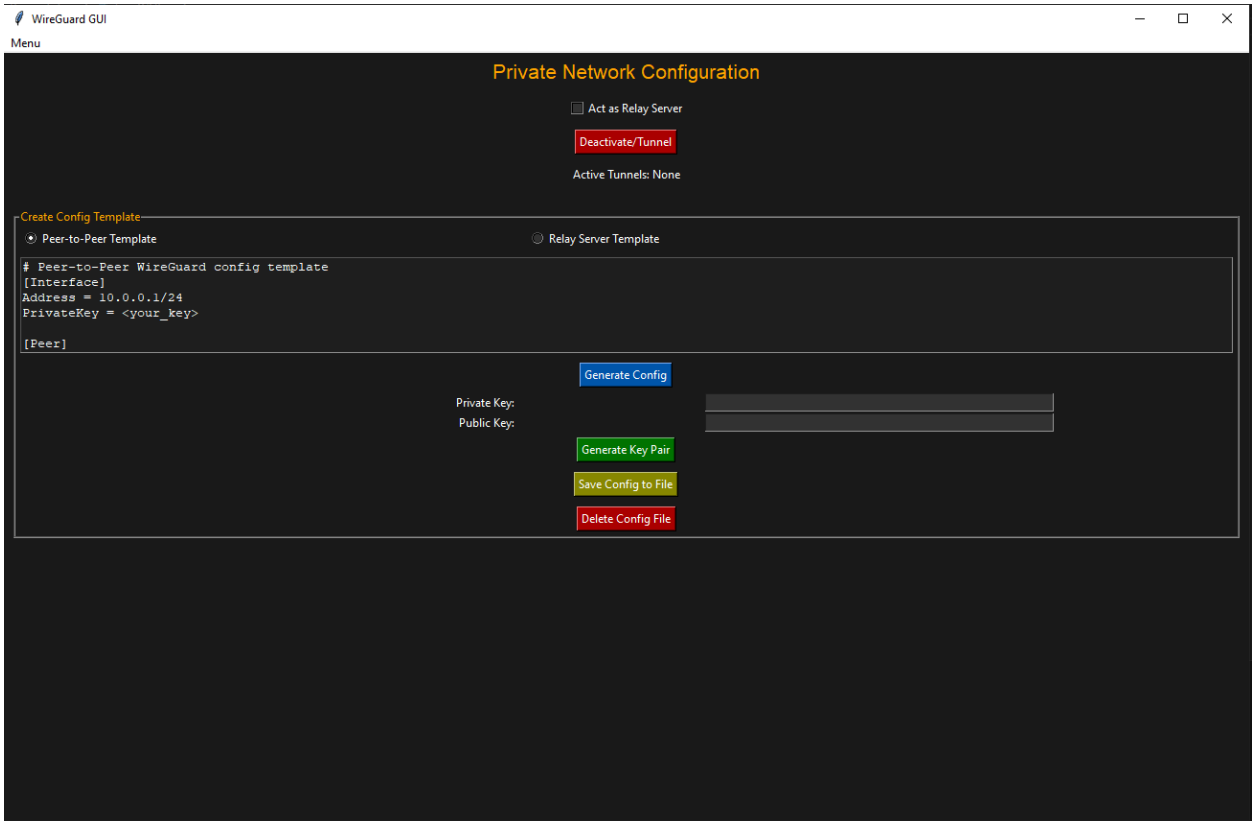


Figure 12: Network Configuration Page

This allows the user to quickly set up and activate the tunnel used between peers. The “Deactivate/Tunnel” button is used for activating or deactivating a tunnel of the user's choice. Clicking the button prompts the user to pick either activate (yes) or deactivate (no), which is then followed by a filedialog scroller to choose your wireguard configuration file. Below that lies dynamic text which shows current active tunnels which updates every 5 seconds. In the middle

sits a text entry box that has three buttons associated with it. “Peer to Peer Template” will be used for classical regular tunneling, the user just needs to input the endpoint’s ip and port, and the private and public keys generated. “Relay Template” would be used for reverse tunneling situations, which will also need a relay server setup by the user. The generate config button automatically generates either template selected when pressed. “Generate keys” button will generate keys for the user to use in their configuration files, they will need to be saved by the user though as they are not saved through the program. “Save config file” and “Delete config file” are used for saving the template file for use as a tunnel and for deleting any unwanted configuration files.

Figure 13: Network Logs Page

Figure 13 is the network logs that will display the attempted connections to various points in the system with their time stamps for troubleshooting purposes. It also displays error messages and alerts that certain functions are running such as starting tunnels, sending or receiving, or starting the AI program.

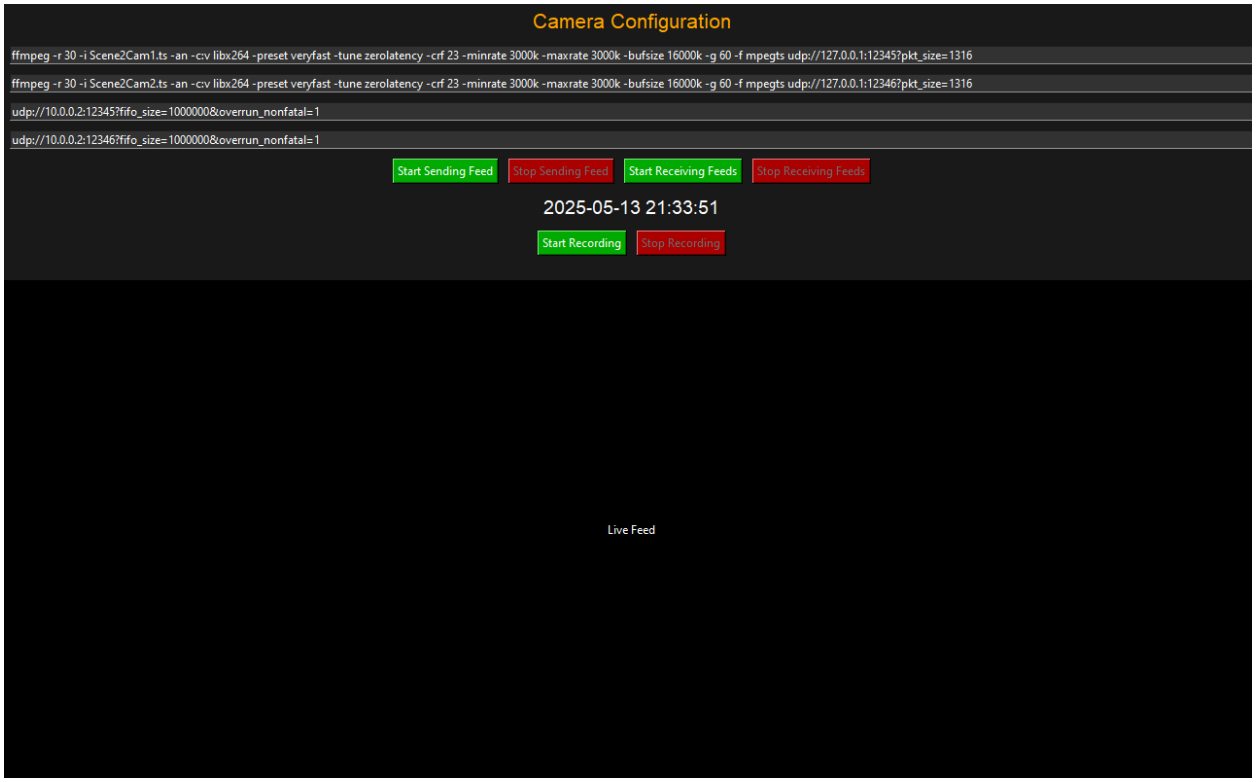


Figure 14: Camera Configuration Page

The camera configuration page shown in **Figure 14** has a set command that gives the user the ability to modify and change their configuration of ffmpeg to their application. This page allows the user to be able to start sending feeds, receiving feeds, and recording live feeds.

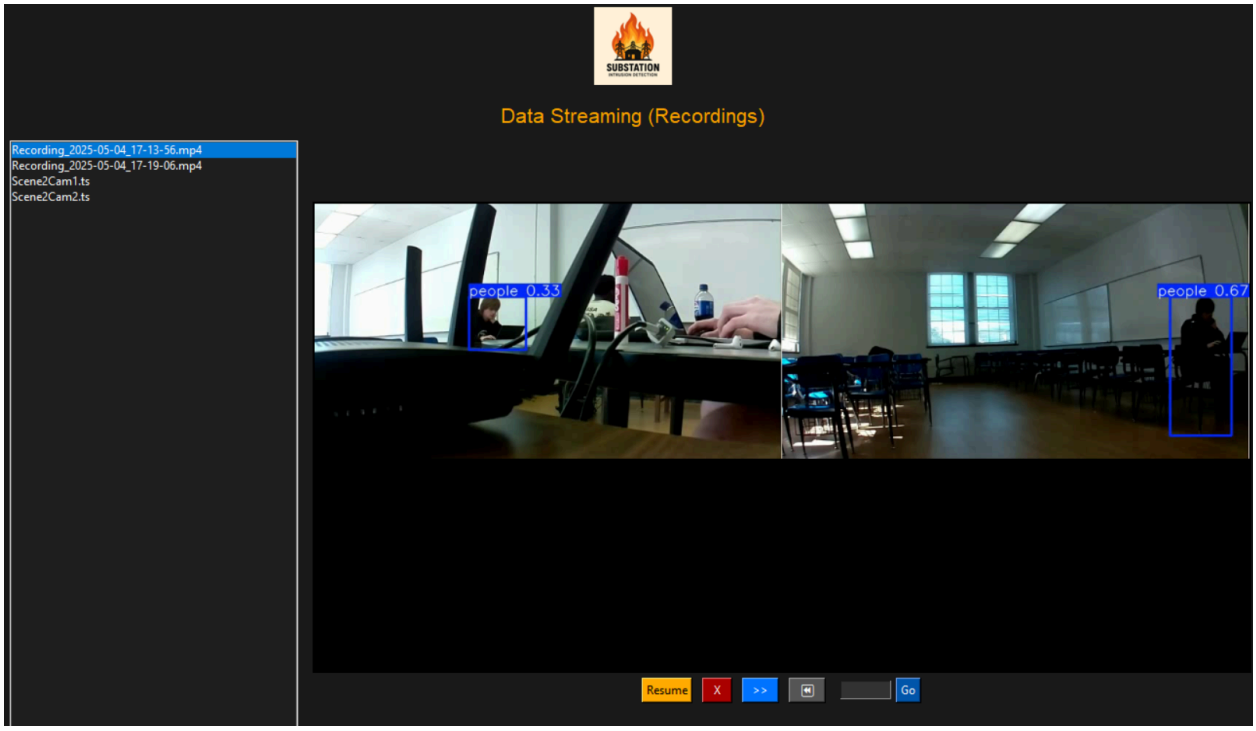


Figure 15: Data Streaming Page

Data streaming is very important as it's where you can access recorded footage from the live feed as seen in **Figure 15**. It features all the footage saved with timestamps in the names, its play through the GUI. The video playback also includes a couple of buttons like pause, stop, speed up, rewind 5 seconds, and a button to skip to a certain timestamp. All footage is saved in a recording folder in the folder with the AI program and GUI program.

The last main page of the GUI is shown in **Figure 16** and it is the virtual zones page.

Figure 16: Virtual Zones Page

On this page the user can adjust the zones, add more, or delete them. Any adjustments are saved on a JSON file so the user does not have to worry about losing their data if the system is shut down. Current format on the left column is X coordinate on the screen and the right column is the Z coordinate being the distance from the camera, creating a zone projected on the ground by passing the use of a 3rd coordinate while still being 3D.

Testing, Data, and Data Analysis

During the fabrication of our prototypes, various testing was done to ensure that different aspects were working as intended. The object detection, camera implementation, and connection between the Raspberry Pis were able to be tested individually while the triangulation needed to be tested after the camera implementation and object detection was implemented.

The testing with object recognition has been partially discussed under prototype development, but this section will touch up on some things that were not discussed previously. After the first attempt of training, the object recognition algorithm a video feed that was not in the training set was given to the program to test the performance. **Figure 17** shows a screen capture of the video feed as it was running. A video feed was used for testing so the tracking of the detection could be tested to see if the algorithm would work on a live feed. At this point, the program is not very accurate, but it is working as intended. This was the only test done at this stage because of how inaccurate the system was performing.

Figure 17: Object tracking training video

Later, after the size of the data set was increased, it was decided to use a live camera feed instead of the video used in the earlier test. This was because there was a slight watermark on the video that could potentially affect the performance of the machine. **Figure 18** shows the output of a single camera feed.

Figure 18: Single camera output feed

For the Raspberry Pi and camera implementations, there are not really any more extra testing features to implement. However, the connection between the Raspberry Pi and the host computer was tested over a local network. **Figure 19** shows where the connection between the host computer and the Raspberry Pi module was tested. Instead of putting the static IP address for the Raspberry Pi here, the IP will be posted in the main Python code where the function that calls on opening the camera feeds is.

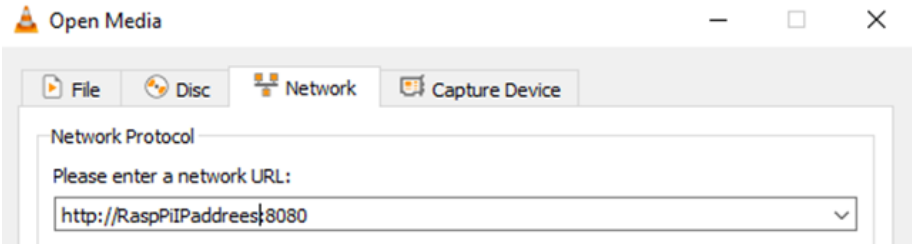
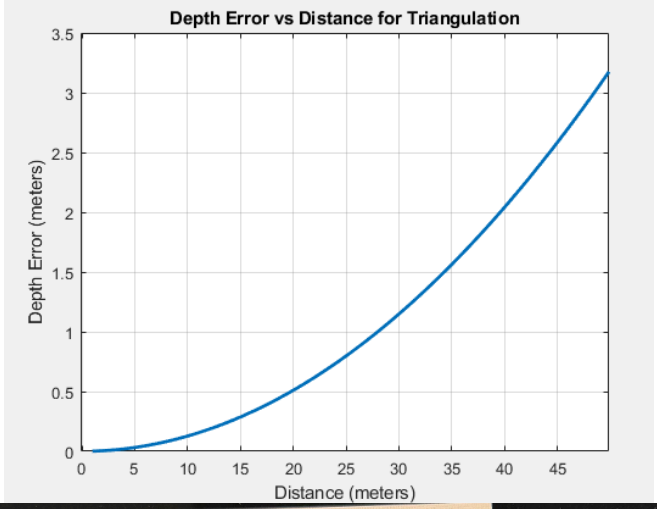


Figure 19: Host computer and Raspberry Pi connection terminal

Following the connection between a host computer and the Raspberry Pi, the triangulation was able to be tested. To do this the cameras have to be a specified distance from each other. For testing purposes at this stage of the project, the cameras are at a fixed distance of 12 inches from

each other. In MATLAB a program was run to see the theoretical error of the system that is shown in **Figure 20.** **Figure 21** shows the first test.

Figure 20: Error vs. Distance graph



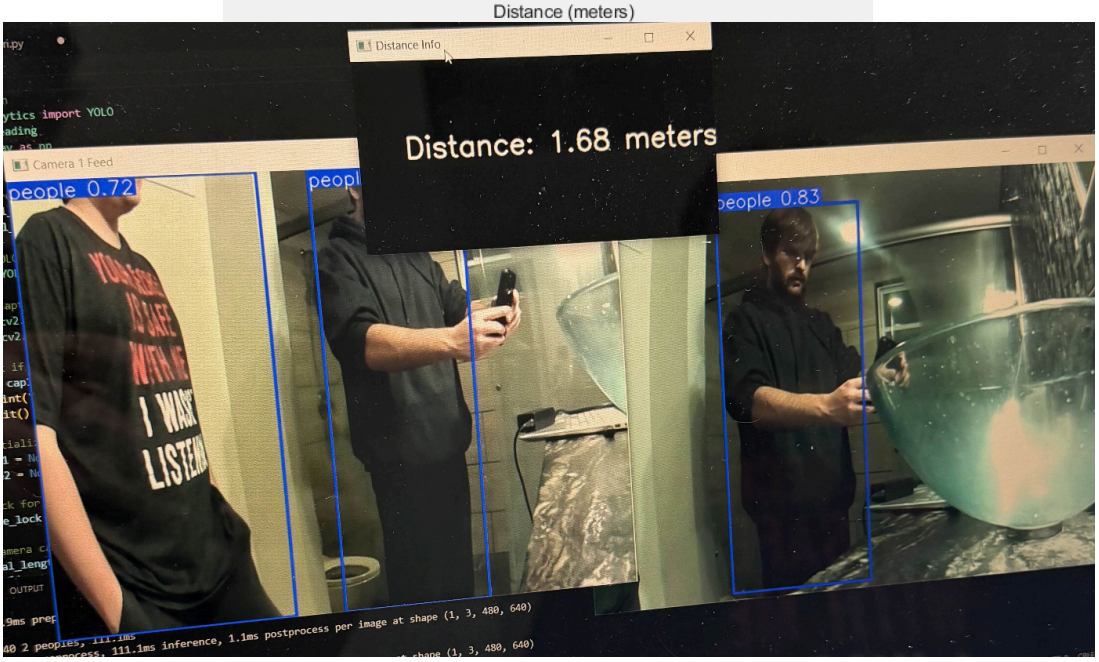


Figure 21: Error vs. Distance Test

The theoretical error is based on the image disparity as that is how the program relies on the distance calculations. The reason the error increases as the distance increases is because the

image disparity becomes small and the algorithm has a difficult time picking up a consistent reading.

It was then decided to have the code output distance in feet to better visualize how accurate the distances are and because the targeted audience of this prototype are American consumers.

The maximum distance the triangulation can reach is about 45ft. It was discovered that this was the maximum distance for a couple of reasons. The first is due to how close the cameras are set up. Because they are close together (18 inches apart for our prototype) the disparity mismatch is almost zero with our current cameras which will display inconsistent readings. The second reason is likely due to our machine learning data. This is also the range where a person is inconsistently read. All of the collected data for this system is of people that are close to the camera. Keeping these considerations in mind this could likely benefit someone that would like to improve on this prototype.

Recently, test data was gathered to see how the system performs at different distances. Distances were measured out on the floor every five feet starting at ten feet. These distances were graphed in Excel to visualize the error. **Figure 22** shows the measured data.

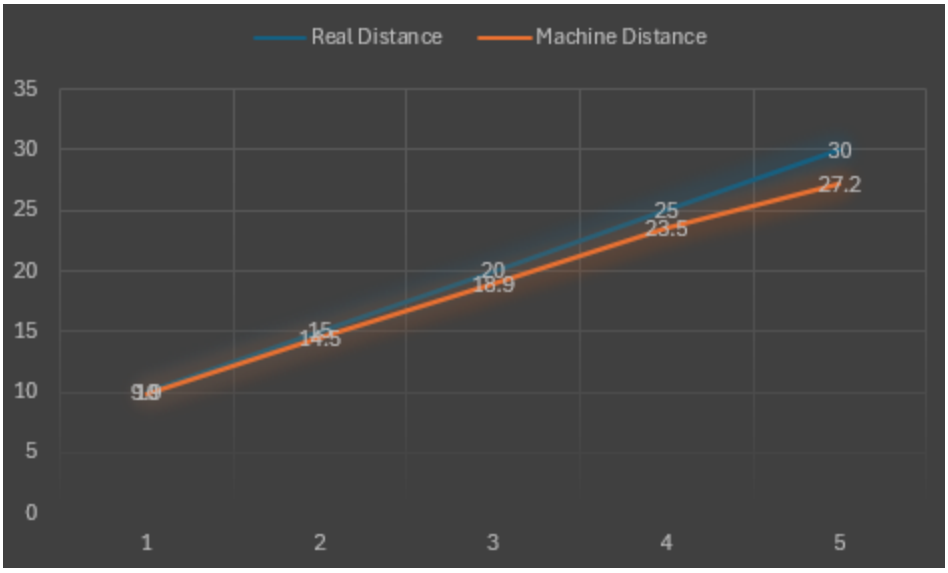


Figure 22: Triangulation Test Data

Looking at these figures, the distances seem pretty reasonable. These distances were taken after many adjustments were made to the algorithm to try and improve them and make them more accurate. Interpolation scaling functions were used to try and reduce the error especially at far distances. The distance displayed for the end user also relies on previous distances from the past couple of frames to prevent huge spikes or dips if the cameras get out of sync. The measured distances started at ten feet because if a person was too close to the cameras, it was pretty inconsistent at picking up readings. Previously, it was considered to take different points of the boundary boxes to get a moving average of disparity to hopefully prevent these spikes. This later was discovered that it did not work and actually gave worse results. This is because of how YOLO works. It analyzes people frame by frame meaning a different boundary box is drawn on each frame. This causes the size of the boundary boxes to fluctuate slightly which makes the output disparity more inaccurate than before. A potential solution that could allow this method to be used in the future is using a GitHub repository called ByteTrack. This algorithm is fast and

lightweight. It can quickly keep track of objects that are moving in a frame [18]. This could be used as a first line of defense so then YOLO does not have to operate every frame and only when a new person enters the frame. ByteTrack has more efficient motion tracking compared to YOLO which could lead to more accurate distance outputs.

After these distances were taken care of there were improvements that needed to be made when tracking multiple people. The first person that stepped in frame has accurate detections but everyone else that steps in frame does not. To fix this problem disparities and distances are stored in matrixes that will be accessed by a matching algorithm. The matching algorithm used in this prototype is a centered based disparity matching algorithm and it references the boundary boxes detected across both frames. Let's say there are three people in frame person A, person B, and person C. The algorithm will look at person A's boundary box and compare the disparity from that box on frame 1 to all three boxes in frame 2. The box that has the lowest disparity is the matching person. This process is done with the remaining people in frame as well. Once one of the boxes are "claimed", the algorithm will not force matches with those boxes anymore to prevent false readings.

Preceding this, the virtual zones were added into the system that would act in parallel to the distance calculations to detect potential intruders. These virtual zones are drawn on the ground with a virtual coordinate system. The two zones used in this prototype are the "blue" zone that spans between 14 - 18 ft away from the system, and the "red" zone spans between 10 - 14ft.

Once the distance calculations and virtual zones have been fully implemented it was time to add a control system with an arduino uno that would activate an alarm if there was an intruder. There

is also a command that will send a notification to the control room if an intruder is detected as well. This response system consists of both hardware and software. On the hardware side there is an arduino Uno board that controls a speaker with a transistor that acts as a switch. The reason for that is because the speaker has an internal resistance of 8Ω and is supplied 5V which requires 625mA to power the speaker while the arduino pins can only supply up to 40mA. The transistor amplifies current to power the speaker from the 5V rail and also receives PWM signals to its base which determines tones and frequency of sound generated by the speaker. Various testing was done to determine the setup of the circuit and the size of resistors that needed to be used to allow the speaker to be loud but not draw so much current that could potentially damage components. We also attached a simple LED set-up to act as flashing lights to mimic prevalent alarm systems.

The arduino also needed to be able to communicate wirelessly with the host computer and it was executed with a HC-05 bluetooth module. To test the connection and find the correct port a software called PuTTY was used to confirm the module functioned properly. We then set up our circuit as shown in **Figure 23.**

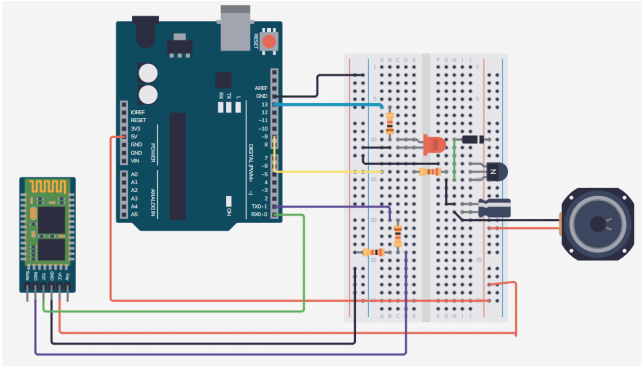


Figure 23: Alarm Circuit

On the software side there was also a notification system that would send information to the control room. This was set up with the MINEText library in Python. It sends notifications via gmail over an ATT SMS gateway so the user can access the notifications on their phone. This configuration can be implemented with other mobile service providers using their respective SMS gateway codes. After everything was set up the alarm system was tied to the red zone while the notifications were tied to the blue zone.

Cost Analysis

Below is a summary table of items that were purchased and have either been implemented in the current system or aided in the testing of certain functionalities.

Table 1. Cost of Purchased Components

Item	Speaker	3D Print	Acrylic Screens	R-Pis	Arduino	Blue-tooth	R-Pi Dual Cameras	Router	Misc. materials
Price	$6.49	$29.88	$10.68	$62.99 x2	$27.60	$9.99	$24.99 x2	$34.99	~$15

Total: ~$310.59

We did not include the cost of the laptops/operating systems used due to the concept of our project being capable of being implemented into any preexisting system. Additionally, some of these components were not purchased entirely out of pocket due to them being provided in our

coursework. Although we still technically purchased all of these components, the actual cost that we paid could vary depending on the website/vendor we accessed them through and any student discounts we may have been eligible for. For our case, we simply collected the highest price of the item we could find within a reasonable range. For example, if we found a site listing the component at an unreasonably high price in comparison to the other prices we found, we would not consider it for our cost analysis. Regardless of only collecting the higher prices, our system is still significantly cheaper than closely related security solutions already on the market.

Table 2. Existing Security Solutions

Item	Hardware	Installation fees	Monthly charges
Price	$1000 - $2500	$300 - $500	~$50

Total: $1300 - $3000 + monthly charges

These security prices were collected from reolink.com [4]. As you can see, our system is vastly cheaper than the security systems that exist currently. These systems are typically implemented in commercial settings, such as gas stations or grocery stores. Although they may comply with different regulations and standards, our system follows the requirements standard for substations.

Design Alternatives

When considering design alternatives for the security system, several factors need to be taken into account, including cost, scalability, and ease of use. As we lack essential resources such as time and money, the following list contains tentative design modifications and additions to the current systems;

1. Facial recognition to distinguish between allowed personnel and unauthorized intruders.
2. Using RFID to detect unauthorized access to restricted areas from a distance.
3. Detection of guns and other dangerous items.
4. Audio detection of gunshots to alert law enforcement officers.
5. Adding capture of video feed to alert messages.
6. Using more cameras for more accurate distance calculations.
7. Using more cameras for a wider field of view.
8. Development and implementation of an AI firewall.
9. Layering interior of camera housing with temperature resistant material to improve weatherproofing.
10. Being able to detect other environmental influences that could be accredited to substation obstruction, such as gas emission or fire detection.
11. Being able to detect when there is any change in digital system traffic that is not typical for the system.
12. Implementing a false alarm system.

The possible adjustments and additions to this system aim to enhance its effectiveness and seal any vulnerabilities in it.

Future Expansion of Project

Given the way this prototype is formatted there are many different functions that could be improved upon in the future. From camera calibration, more complex object tracking, geo location, and more user adjustments in the GUI, there is a lot of room for improvement.

Firstly, the distance calculations could be close to perfect at any distance with a few implementations. These implementations are checkerboard camera calibration, Gaussian process, and ByteTack. The checkerboard calibration involved taking several pictures with a checkerboard that would allow the program to recognize errors in the camera's hardware such as camera distortion along the edges [10]. The Gaussian process is an algorithm that is more advanced than the interpolation technique that is implemented into the current prototype. Theoretically, you would give it preset distances and it would use those distances to give a confident “guess” of how far a person is. The reason that this algorithm is good is because it doesn’t assume a fixed relationship between distance and disparity (i.e. linear). It would also allow the user to recalibrate the system if the user is not happy with the end result. Finally, as previously mentioned ByteTrack is an efficient and lightweight tracking algorithm that could more accurately pinpoint where a person is located.

Our system proves to be sustainable over time due to the fact that it implements a lot of open-source materials. Additionally, our software itself is also open-source. This means that our

system can be updated or tailored at any point in time. Because our product follows the standards in place for substations, it also falls within the maintenance protocols respective for substation software. With this, the technicians would be responsible for learning our system and being able to troubleshoot any issues that may arise. Because of this, we made our system as easy to use as possible and even created ways to calibrate the triangulation zones.

Conclusion

Overall, our proposed project, the Substation Intrusion Detection System, aims to enhance the current security systems that are already in place in electrical substations today. Because substations are the most vital component of our communities' power infrastructures, we believe that their security is imperative to be as reliable and fully functional as they can possibly be. By addressing both physical and cyber security vulnerabilities, our proposed multi-layered security approach aims to effectively overhaul outdated security measures into a much more versatile system. Key features of our proposed system include: a triangulated camera network with both night and day vision capabilities, integrated audio inputs and outputs, real-time object detection and tracking, an AI firewall, and encrypted data transmission.

Through many phases of testing, our group has made significant progress toward our end goal for this project. We have successfully implemented object recognition and tracking, triangulation calculations, and night and day vision cameras. Our group is almost completed with Wireguard and the data encryption, we just need to buffer a few things out in order to get it fully functioning. The housing of our system has been submitted for printing and we will have that completed before the next term. That will leave next term for fixing and modifying pieces of our system to make it as efficient as possible. At the end of it all, we would also like to make our software open-source for anyone to access and modify however they see fit.

Ultimately, the implementation of this project will significantly enhance the resilience of critical infrastructure, pose an effective deterrent for substations, and ensure continued reliability of electricity and power. By developing an open-source software that can be fluidly implemented

into any pre-existing system, the Substation Intrusion Detection System not only provides immediate benefits for substations but also has the potential to be tailored and configured into a multitude of different security systems. As threats to these facilities grow, our system would be capable of saving companies and communities millions of dollars annually.

Citations

Datasheets

Raspberry Pi operating temps:

https://copperhilltech.com/content/The%20Operating%20Temperature%20For%20A%20Raspberry%20Pi%20–%20Technologist%20Tips.pdf

Power consumption for Raspberry Pi:

https://www.pidramble.com/wiki/benchmarks/power-consumption

Camera module:

https://www.amazon.com/MELIFE-Raspberry-Camera-Adjustable-Focus-Infrared/dp/B08RHZ5BJM

More Raspberry Pi specs:

https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/

Arduino uno:

https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf

Amplifier:

https://cdn-learn.adafruit.com/downloads/pdf/adafruit-max98357-i2s-class-d-mono-amp.pdf

Speaker:

https://www.daytonaudio.com/images/resources/285-103-dayton-audio-ce32a-4-spec-sheet.pdf

Accelerometer:

https://www.analog.com/media/en/technical-documentation/data-sheets/adxl345.pdf

NVidia Graphics Card:

https://www.nvidia.com/content/geforce-gtx/GEFORCE_RTX_2070_SUPER_User_Guide.pdf

Sources

[1] admin, "How does Hawk-Eye work in tennis?," *Soccer summer camps and academies all over the world*, Oct. 04, 2017. https://www.ertheo.com/blog/en/hawk-eye-work-in-tennis/

[2] B. Habegger, "'I may not be alive tomorrow' | PG&E power shutoff threatening lives," *abc10.com*, Oct. 24, 2019. https://www.abc10.com/article/news/local/wildfire/i-may-not-be-alive-tomorrow-pge-power-shut off-threatening-lives/103-d15b23b9-6ecd-4944-8e51-6e07c6b004b5

[3] "Ch. 13 MN Statutes," *www.revisor.mn.gov*. https://www.revisor.mn.gov/statutes/cite/13

[4] "Cost of Home Security System 2025 [Latest Update]," Reolink.com, 2025. https://reolink.com/blog/cost-of-home-security-system/?srsltid=AfmBOoqO_UMsTWdAv4O1N pSXbQ1SBrlCjwVnBc7A6vYP9QhCpO7jhp4G.

[5] EPA, "Greenhouse Gas Equivalencies Calculator," *www.epa.gov*, Aug. 28, 2015. https://www.epa.gov/energy/greenhouse-gas-equivalencies-calculator#results

[6] Federal Trade Commission, "FTC Says Ring Employees Illegally Surveilled Customers, Failed to Stop Hackers from Taking Control of Users' Cameras," *Federal Trade Commission*, May 31, 2023. https://www.ftc.gov/news-events/news/press-releases/2023/05/ftc-says-ring-employees-illegally-surveilled-customers-failed-stop-hackers-taking-control-users

[7] Great Lakes Electronics Corporation, “How Are Electronics Recycled? A Step-By-Step Guide | Ewaste,” *Great Lakes Electronics*, May 26, 2022. https://www.ewaste1.com/how-are-electronics-recycled/

[8] K. Wood, “Cybersecurity Policy Responses to the Colonial Pipeline Ransomware Attack,” *Georgetown Law*, Mar. 07, 2023. https://www.law.georgetown.edu/environmental-law-review/blog/cybersecurity-policy-responses-to-the-colonial-pipeline-ransomware-attack/

[9] Mark, “Calibration Checkerboard Collection,” *Mark Hedley Jones*, Apr. 14, 2018. https://markhedleyjones.com/projects/calibration-checkerboard-collection

[10] National Institute of Environmental Health Sciences, “Human Health Impacts of Climate Change,” *National Institute of Environmental Health Sciences*, Nov. 07, 2022. https://www.niehs.nih.gov/research/programs/climatechange/health_impacts

[11] “(197h) Noble Gas Infused Neoprene Closed Cell Foams for Ultra-Low Thermal Conductivity Textiles | AIChE,” *Aiche.org*, Apr. 06, 2025. https://www.aiche.org/conferences/aiche-annual-meeting/2018/proceeding/paper/197h-noble-gas-infused-neoprene-closed-cell-foams-ultra-low-thermal-conductivity-textiles

[12] Raspberry, “Design Life-Cycle,” *Design Life-Cycle*, 2014. https://www.designlife-cycle.com/raspberry-pi

[13] R. Fergus, “Biased Technology: The Automated Discrimination of Facial Recognition | ACLU of Minnesota,” *www.aclu-mn.org*, Feb. 29, 2024. https://www.aclu-mn.org/en/news/biased-technology-automated-discrimination-facial-recognition

[14] R. Kundu, “YOLO: Real-Time Object Detection Explained,” *www.v7labs.com*, Jan. 17, 2023. https://www.v7labs.com/blog/yolo-object-detection

[15] *The Future of Electric Power in the United States*. Washington, D.C.: National Academies Press, 2021. doi: https://doi.org/10.17226/25968.

[16] Ultralytics, "Models," *docs.ultralytics.com*. https://docs.ultralytics.com/models/

[17] "Vulnerable U.S. electric grid facing threats from Russia and domestic terrorists," *www.cbsnews.com*. https://www.cbsnews.com/news/america-electric-grid-60-minutes-2022-08-28/

[18] Y. Zhang, "ByteTrack," *GitHub*, Nov. 09, 2023. https://github.com/ifzhang/ByteTrack