iSPY: Image Surveillance and Protection Yielder

Alejandro Lopez Zelaya [1], Armani Garcia [2], Gabryella Baldaci [1], Luke Reyes [2], Robert Josephson [2]

[1] CREOL, College of Optics and Photonics, University of Central Florida

[2] Department of Electrical Engineering and Computer Science, University of Central Florida

***Abstract* — As mediums of communication become susceptible, there is a need to find alternatives. Susceptibility stems from compromised wired mediums like cables or wireless mediums like WiFi. To prepare, this paper details the design of a surveillance system employing a secure medium of communication – light. Wired and wireless connections are nonessential. iSPY can be divided into two segments. On the transmission end: live video, a wide shot of the surrounding, is captured, then analyzed through computer vision and sent using free space optics. At the receiver end: signal decoded and the scene displayed with computer vision information.**

***Index Terms* — Free space optics, optical wireless communication, computer vision, optical design, bit-stream, PCB design, communication system.**

I. Introduction

The idea of a new kind of surveillance system arose when we noticed how flawed the most popular systems are. Current market options have been failing users with their frail reliability once it comes to internet connection, making it an easy target for hackers [1], [2].

The key innovation of iSPY is the use of optical wireless communication (OWC) by means of point-to-point free-space optical communication (FSO). This approach reduces the vulnerabilities that arise when using a conventional method such as Wi-Fi or radio. FSO ensures proximity-based security, which is what we aim to achieve as optical signals are much harder to intercept without physical access to the system. To enhance the surveillance capabilities, we also make use of machine learning algorithms to achieve computer-vision, which enables iSPY to detect and count individuals in the monitored area to a high degree of accuracy.

The hardware architecture of iSPY is built around three key sub-systems: image acquisition, FSO communication, and electrical hardware. Employing a wide field-of-view and computer vision software, the image acquisition sub-system captures and preprocesses live video. The evaluation board compresses the data into bitstream, and the FSO sub-system transmits it securely over the air in a free space optical channel to a receiver unit, which decompresses and displays that data to an external display. The electrical hardware sub-system includes the two PCBs involved in the entirety of the system, the first being Raspberry Pi 5 and the second being a custom design that can handle efficient data processing, decompression, and signal integrity. Below is a render of the final product.

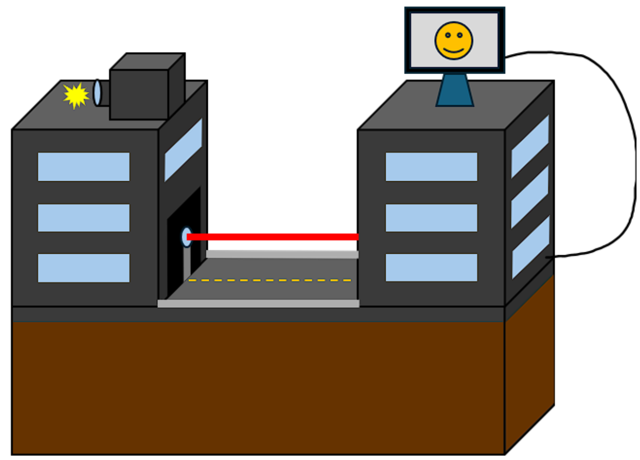


Fig. 1 Render of iSPY system to be presented.

iSPY is a comprehensive surveillance system featuring a wide range of custom electrical and optical hardware, along with specialized computer algorithms, designed to advance modern surveillance technology.

II. Goals and Objectives

When designing iSPY, we envisioned a system that would be cheaper, more secure, and more efficient than current market options. Cheaper means working within a specified budget. More secure means it is harder to eavesdrop on the transmitted information without physically blocking the communication path or altering the signal intensity. More efficient means to yield more useful information to the end-user.

To accomplish this objective, we set goals, such as: obtaining a wide field of view using less than 4 lenses; transmitting and receiving optical signals over a working distance of 0.25m to emulate long-haul distances; detecting bodies and counting the number of individuals recognized from video feed; implementing codecs for real-time compression to achieve low latency; allowance for farther data transmission. Each of these goals can be achieved to a high degree.

III. ENGINEERING SPECIFICATIONS

The core specifications for iSPY can be found in the table below. The first three specifications are to be demonstrated through live demos, videos and showcases.

TABLE I
ENGINEERING SPECIFICATIONS

Subsystem	***Parameter***	***Specifications***
Imaging	FOV	80° (H) 50° (V)
Computer Vision	Accuracy of detection	75%
Communication	Emulated Distance	> 10km
Image Processing	Clock Speed	16MHz
Imaging	Framerate	30 fps
Power	Voltage Output	3.3V DC reg.

IV. SUB-SYSTEMS

Our system is a one-directional, simplex, communication system. The following sub-systems discussed below enable iSPY to operate as intended.

A. Image Acquisition System

The optical design for our image acquisition system began with the choice of the sensor, and not with the lenses as one can expect. Since sensors are usually not custom-made, and they follow standards in the industry – we understood that little to nothing could be changed about them. Therefore, choosing the right sensor is the starting point of many optical engineers when designing such systems.

Several parameters came into play when deciding on a sensor, such as its size, resolution, frame rate, shutter type, and of course, interface compatibility with the other parts of iSPY. The sensor size dictates nearly all operational metrics of the imaging system, and since one of our core goals was to obtain a wide field of view (FOV), this parameter was extremely important [3]. An equation that shows the strong relationship between sensor size and the parameters is:

$$\text{FOV (in degrees)} = 2\ \tan^{-1}\left(\frac{w/2}{f}\right) \quad (1)$$

Where f is the focal length and w is the sensor width [4].

FOV is an angle measurement of how much of a scene is visible to a device or individual. It is the parameter that lets the viewer know how much they can see. Equation (1) shows that a large sensor will increase our FOV. However, larger sensors are more expensive and require larger lenses, which are also on the costly side. This is where tradeoffs began. This project was self-funded and to accomplish everything we had in mind we chose to focus on the overall functionality and not the highest quality for every single component.

Just as important as the sensor size is the focal length – a short focal length is necessary to achieve a wide FOV. The simplest definition of focal length is the distance between a lens and its focal point. However, for a lens system consisting of more than one lens, the combination of focal lengths, the effective focal length, is the measurement we need.

After researching the different CMOS sensors available, we concluded that the sensor in the Pi Cam Version 3 was the best option. The sensor size is 6.45 x 3.63mm, which requires a focal length of ≤ 3.7mm in order to obtain a horizontal FOV of 80°, and a vertical FOV of 50°. The optics that come with the sensor do not meet spec and must be replaced.

The next step was to design a lens assembly for this system. Obtaining a short focal length is a challenge, especially with the limiting factors of time and money. Lens design for a wide FOV camera require several surfaces and are prone to aberrations since light is coming from different angles. The optical design software Zemax was used for simulating our lens system. However, we could only use lenses that were already available (off-the-shelf) since custom-made lenses would significantly increase the total cost of our project. This limitation introduced some challenges. After several iterations of simulations, the best fit for our system was found.

The final design uses a pinhole as an aperture stop and entrance pupil. A stop is a hard physical aperture that limits the amount of light into an optical system, and since the pinhole is also the first element in the assembly, it acts as the entrance pupil [5]. The pinhole then can be compared to a filter, it reduces the noise (amount of light) in the system.

Through Zemax, we found that an entrance pupil of diameter 460 µm would work. The closest precision pinhole available was a 400 µm.

For the first lens we went with an aspheric lens. This lens introduces less aberrations to the system. Aspheric lenses do tend to be more expensive than conventional lenses; however, by using an aspheric lens we reduce the size of our system and aberrations. Additionally, a single aspheric achieves the same (sometimes even better) result as two to three lenses. On top of that, a surveillance system needs good image intake and using an aspheric lens was an investment on the overall system, it ensured higher resolution.

The second and last lens is convex-plano. Its convex side follows the first lens, and it helps converging the light onto the sensor. As the last element in this system, this lens must cover the sensor entirely, to avoid vignetting, and this is the reason why its diameter is 9 mm.

Both lenses have an anti-reflective coating that maximizes the transmission of the wavelengths we want (visible range) and blocks the remaining.

The final image acquisition system then consists of a pinhole, an aspheric lens, a convex-plano lens and a CMOS sensor.

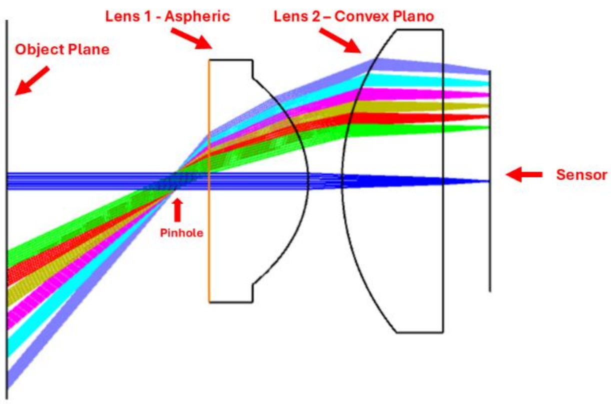


Fig. 2 iSPY's final lens design.

This design enables us to meet FOV spec and its performance is comparable to current market options containing more lenses. By reducing the amount of lenses, we use less space, which would decrease the production cost if mass produced.

B. Computer Vision

The computer vision aspect for our project began by determining which algorithm and model would be best suited for detecting bodies and recognizing them for counting purposes. The algorithm and model determined how the overall structure of the computer vision system would function. If the model was not efficient enough then the accuracy of detection would be less than our prospective accuracy of 75%.

Many features – such as which neural network the algorithm used, either: convolutional neural networks (CNN), recurrent neural network (RNN) or natural language processing (NLP) – were considered when deciding the algorithm and model [6], [7]. The use of Deep Neural Networks was found to be more common in our current time due to the fact that they are a hybrid process and decrease the number of resources used for object detection [7]. Once the information on the neural networks was gathered this led to the different coding libraries that use coding commands that handle computer vision tasks.

The coding libraries that were focused on were as follows C++, Java and Python. They all had their very own library that worked for computer vision. The C++ library, which is known as Convolution Architecture for Fast Feature Embedding (CAFFE) was a modular library and used CUDA for GPU computations which was referenced in Chapter 3 Section 2 under C++ Computer Vision Library. However, this C++ library was limited to certain computer vision task which limited the usage for our project.

The next library considered was a Java based library known as BoofCV which used various features from image processing like image association to geometric vision such as 3D structure estimation and could render features in multi-dimensional spaces. Although the Java library was able to handle more tasks, it performs slower on certain basic computer vision aspects such as detection.

The Python library was the last one we focused our attention on, and we focused on three different Python libraries: TensorFlow, SimpleCV and OpenCV. TensorFlow was able to develop models on mobile as well as embedded devices and is regularly updated with new features. Although, TensorFlow is very resource heavy. SimpleCV was a collection of libraries and is used for more prototyping than actual real. SimpleCV is limited in computer vision features compared to other libraries for real project type applications. OpenCV has optimized algorithms and the version that is most used is OpenCV-Python which combines Python with OpenCV C++ API. Due to the vast variety of algorithms and computer vision programmability, OpenCV-Python was chosen as the library to use for our project.

OpenCV has many algorithms that can handle various tasks. Some algorithms handle body detection such as Haar Cascade and Histogram of Gradients (HOG) [8].

Haar Cascade used Haar features that detect patterns as well as cascade which breaks down the process of detection and has the ability to exclude irrelevant regions in images [9]. Also, Haar Cascade has pre-built/trained packages known as classifiers. HOG used the gradients for feature extraction and utilizes classifiers.

After researching the two algorithms that worked with OpenCV and OpenCV-Python we did more research into newer algorithms, mainly the YOLO (You Only Look Once) algorithm. We only considered the most current versions at the present time: YOLOv9, YOLOv10 and YOLOv11.

YOLOv9 has two techniques for increasing learning capabilities and a few different models like lightweight (tiny and small), medium (medium) and large (compacted and extended) models [10]. YOLOv10 was built from a prior version of YOLO and had two components that allowed YOLOv10 to handle accurate detection in real time applications. Similar to YOLOv9, YOLOv10 had lightweight (nano and small), medium (medium and balanced) and large (large and extra-large) models [11]. YOLOv11 was the most current YOLO algorithm at the time and used custom implementation on the old YOLOv8 algorithm improving performance both in accuracy as well as efficiency. YOLOv11 has the same type models, however, it doesn't have a balanced model like YOLOv10.

YOLO-NAS is a searchable network for architecture's utilized for certain computer vision tasks. YOLO-NAS has similar models like YOLOv10 and YOLOv11. However, YOLO-NAS has INT8 version which are 8 bit integer quantized models [12]. After researching the different YOLO algorithms and models, we considered the following to compare: YOLOv10, YOLOv11 and YOLO-NAS.

After comparing the algorithms and their corresponding models we found that YOLOv11 nano model was the best choice for iSPY.

There was another area we researched which was differences among certain aspects. One of those areas was the difference between classifier and object detector. Haar Cascade and HOG used classifiers where YOLO used object detectors. Classifiers are used for classification on objects where object detectors detect the object with labels and bounding boxes.

The other difference we examined was the difference between body detection and body recognition. Body detection boxes out an individual from the background without labeling meaning it doesn't classify the object as a person. Body recognition boxes and labels the individual based off its features meaning that it classifies the object as a person.

C. Free Space Optics System

The most significant novelty for iSPY lays in the means of communication – optical wireless communications (OWC). OWC includes all methods and systems for computer communication that use light to transmit data wirelessly, instead of RF frequencies. Unlike optical fiber communication, light is not confined to a transmissive medium like an optical fiber; instead, light is transmitted in an unguided medium, like air, water, or the vacuum of space, by controlling the divergence and directionality of light within the OWC medium using optics. Point-to-point OWC systems enabling satellite-to-satellite, air-to-air, satellite-to-ground, or ground-to-ground communication using lasers are known as free-space optical (FSO) systems. FSO systems enable ultra-high bandwidths, high beam directionality, robustness to electromagnetic interference, a high degree of spatial confinement (making each frequency highly reusable in small areas), and inherent physical security (tight confinement means small area for detection) – all reasons that make OWC a suitable and appropriate means of communication for surveillance systems like iSPY.

To create a working OWC system on a budget, a point-to-point FSO link was created using optical transceivers, optical fibers, optical fiber collimator lenses, and a thoughtful selection of neutral-density filters. A general diagram of the FSO link is shown below.

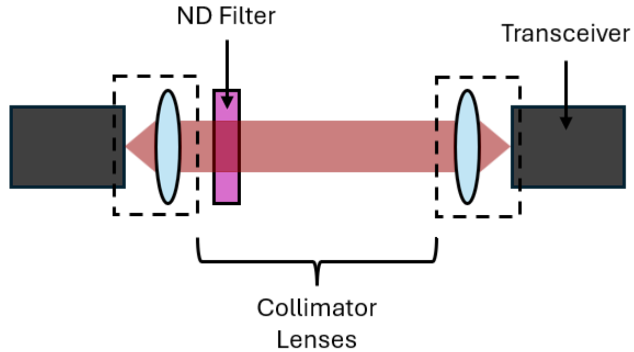


Fig. 3 General FSO link schematic.

Optical transceivers are the ideal optoelectronic device for receiving and transmitting optical signals through an optical fiber and, with and after careful selection, can be used for fiber-enabled FSO. The optical fibers are necessary for extracting and launching modulated laser signals to and from the transmitter and receiver of an optical transceiver, respectively, while also mitigating losses. The optical collimators are necessary for confining light to tight spaces in free space. Neutral density filters serve the purpose of protecting the sensor within each optical transceiver and emulating the optical loss experienced by the optical signals due to atmospheric absorption.

The optical communications hardware for this system is chosen with careful consideration of the optical power budget, collimated beam divergence, and external source radiance. The optical power budget maximum and minimum values are determined by the optical transceiver parameters of max transmission power and receiver noise floor, respectively. The sources of loss for the FSO system include all components light in free space encounters. For this reason, an optical transceiver with a power budget greater than 17 dBm was found. The collimated beam divergence is determined by the light source and the collimating optics of choice. Fiber collimators can be engineered or pre-packaged off-the-shelf; to save resources, a pre-packaged and AR coated singlet collimator was obtained. The external source radiance depends on the free-space launch conditions. For this situation, the numerical aperture of the optical fibers and lenses protect the system from external sources of light. As for the optical fibers chosen, they must be compatible with the optical transceiver. High-power-budget optical transceivers are typically for long-haul optical fiber communications via single-mode optical fibers with an LC/UPC connecter. Since high-power-budget transceivers were obtained, the only option was to use single-mode fibers. Fortunately, the fiber collimators obtained were for single-mode fibers, making them an ideal choice. If multi-mode options were explored, excess of $500 would have to be spent on optics alone.

Lastly, to facilitate the exchange of electrical and optical data, media converters were obtained.

D. Electric Hardware - PCB

The design for the PCB on the latter side of the system has three major components and has gone through multiple changes in the overall design. The three major sections of the designed PCB are the MCU, the voltage regulation daughter board and the ethernet communication daughter board.

The developed PCB is responsible for receiving a bitstream video feed via Ethernet, relaying the data to a host computer for monitoring, and controlling status indicators for system feedback. The board integrates an ESP32-S3-WROOM-1-N16R8 microcontroller unit (MCU) for data handling and an Ethernet module utilizing the W5500 controller for network communication. Additionally, a power regulation system ensures stable operation, supplying 3.3V to critical components while maintaining power integrity via a 5V USB-A source.

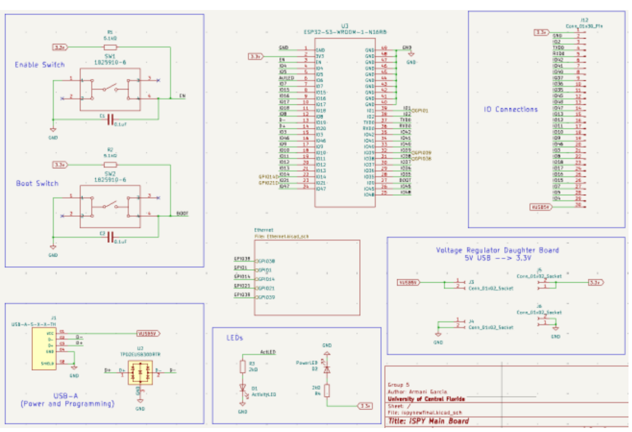


Fig. 4 iSPY Board #1: Base Carrier

The designed PCB serves as a critical node in the iSPY optical security system, responsible for processing and forwarding bitstream video data. It is composed of several key components. The ESP32-S3-WROOM-1-N16R8 microcontroller unit handles Ethernet communication, processes incoming data, and relays the information to a computer via a serial connection. The W5500 Ethernet controller provides hardware TCP/IP stack support, enabling efficient data transfer via Ethernet. A 3.3V voltage regulator converts the 5V USB-A power input to a stable 3.3V supply for the MCU and Ethernet module. Two indicator LEDs provide real-time feedback, with one connected directly to the 5V power line to indicate power status and another interfaced with an MCU GPIO pin to indicate data reception activity.

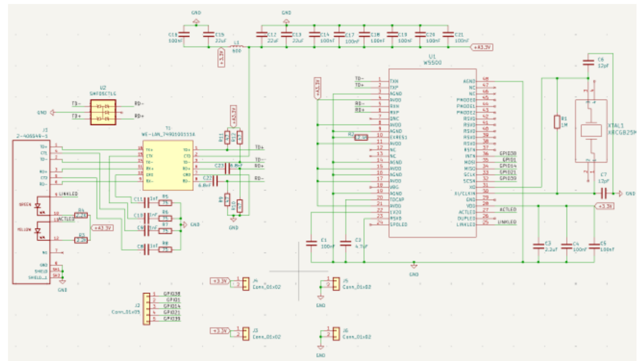


Fig. 5 iSPY Board #2: Ethernet Daughter Board

The PCB is powered via a 5V USB-A connection, which is regulated down to 3.3V using a dedicated voltage regulator daughter board. This ensures stable operation for both the ESP32-S3 and W5500 Ethernet module. The direct 5V line is also used to power an onboard LED, serving as a power-on indicator.

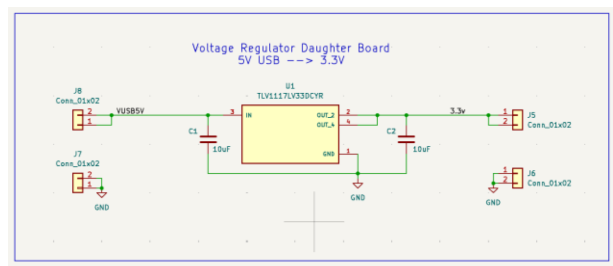


Fig. 6 iSPY Board #3: Voltage Regulator Daughter Board

The W5500 Ethernet controller facilitates reliable, low-latency communication by offloading TCP/IP processing from the MCU. The module interfaces with the ESP32-S3 via the SPI bus, allowing efficient data transfer while minimizing processing overhead on the MCU.

To enhance usability and debugging capabilities, the PCB incorporates two LEDs. The first LED is a power indicator that is directly connected to the 5V rail and remains illuminated when the board is powered. The second LED, controlled by an ESP32-S3 GPIO pin, flashes to indicate active data reception and transmission.

E. Image Processing and Data Transmission

The data captured using YOLO must be properly processed to allow for efficient and smooth data transfer. It began by determining which codec would suit the needs best for live data compression and transmission. A codec will allow for compression of the data into bitstream so that we can transfer the data via free space optical communications.

H.264 was chosen due to its efficiency and clarity of image after compression and decompression. It is widely used and has a variety of documentation. FFmpeg can utilize H.264 well and is able available in python which is the code that will be used for this project.

FFmpeg will be taking the data captured using YOLO. First, it will take the feed of the video and compress it into H.264 file format. Second, it will convert the feed from H.264 format to bitstream. The feed can then be transmitted using the FSO system.

The ESP32-S3-WROOM-1-N16R8 microcontroller will receive the bitstream signal and utilize FFmpeg to decode the bitstream data into a viewable format. This will convert the bitstream into an H.264 file format. The code will then take the H.264 file and convert it into a viewable format.

V. Prototyping and Testing

A. Image Acquisition System

The designed optical system came to life once all the parts were purchased. The biggest task was to ensure the separation between all elements was exact. The distance from the pinhole to the first lens was 1 mm, the distance from the first lens to second lens was also 1 mm, and from the second lens to the sensor was 1.3mm.

	Surface Type		Comment	Radius	Thickness	Material	Coating
0	OBJECT	Standard		Infinity	Infinity		
1		Standard		Infinity	5.000		
2	STOP	Standard		Infinity	1.000		
3		Standard	354105-A	Infinity	2.937	D-ZK3...	THORA
4	(aper)	Even Asphere	354105-A	-3.214	1.000		THORA
5	(aper)	Standard	45082	7.060	3.000	N-SF11	EO_M...
6	(aper)	Standard		Infinity	1.382		EO_M...
7	IMAGE	Standard		Infinity	-		

Fig. 7 iSPY's lens data.

Different mounting structures were tested until we reached the final look. This structure was inspired by Matryoshka dolls, also known as nesting dolls, in which we start with a larger piece and inside we find the smaller ones. Each piece is the mount for one element (pinhole and lenses). On figure 8, the cad model and the final structure can be seen.

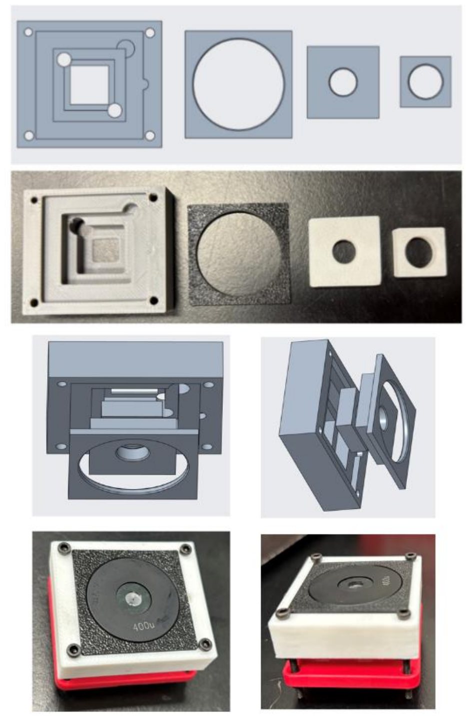


Fig. 8 Cad model for image acquisition's final structure

Using this structure, we were able to reach the required distance between the elements. This was proven by carrying out a test to measure our FOV. We used grid paper and marked the extremities of what was captured by the camera. Since the size of the grid squares is known, we could find the real measurements and from that we used the formula below to obtain the FOV.

$$\theta_{V\ or\ H} = 2\ \tan^{-1}\left(\frac{Scene\ dimensions}{2d}\right) \quad (3)$$

Where the variable d is the distance from the camera to the grid paper, and the scene dimension is the real distance. The figure below is a screenshot of a live video obtained by our system.

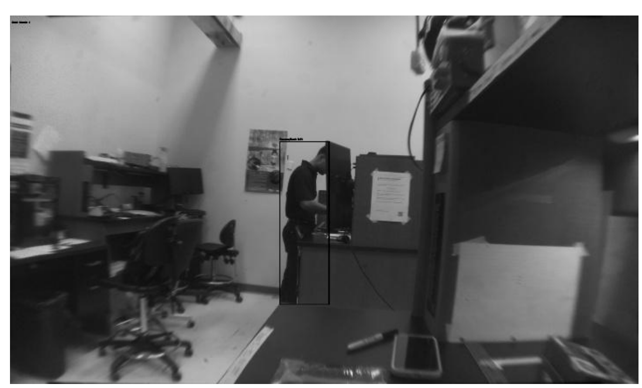


Fig. 9 Live video capture by iSPY

B. Computer Vision

The computer vision aspect started to really form after prototyping the original program. The original program took a Zemax made image and detected people in that image.

Fig. 10 Results from YOLOv11 on Zemax image

Although the results were very impressive there were flaws in the code and we needed a program that can handle video. The original program focused on only detecting people with a confidence score or level above 50% and didn't count the number of individuals in the image which would later be worked on during the testing for video.

Once a camera was obtained rigorous testing began by first changing the program to take in video frames. After reading in video frames another issue occurred which was we were not just detecting people but other objects. The issue was solved by changing the class to index 0 which is for the object "person" this allowed the program to only detect people in the video frame.

Another problem that was determined happened when designing the counter for the number of people recognized in each frame. The counter was based off of the confidence score meaning that if the programs computer vision model was less than 50% confident that the object was a person it wouldn't count that person even if the person was boxed out by a bounding box which lead to improper counting.

In order to fix the counter issue object tracking would be implemented into the program as a result of this implementation can be seen in Figure 9. After the counter was fixed the final change made was for aesthetic purposes and working on the Raspberry Pi 5 microcontroller device which utilized the Image Acquisition System as shown in Figure 9.

The change for the aesthetic purpose was changing the color scale of the frames to gray scale and having red labeling as well as red bounding boxes.

C. PCB Testing

The testing phase for the custom PCBs used in the optical security system was crucial in refining the design and ensuring system reliability. Initial testing focused on validating the electrical integrity of the assembled boards. To verify that power was properly distributed, we used a digital multimeter to measure voltages at key nodes throughout the board. This included confirming that the 3.3V regulator consistently supplied power to both the ESP32-S3-WROOM-1-N16R8 microcontroller and the Ethernet communication daughter board, and that the 5V power line from the USB-A port was present and stable. The presence of power was also indicated visually by an onboard LED connected directly to the 5V rail, which served as a straightforward check for power delivery.

We were able to successfully flash a basic program to the ESP32-S3 that allowed us to control the GPIO-connected activity LED. This confirmed that the MCU could be programmed and that GPIO functionality was intact. However, further functional testing was limited by earlier design limitations.

One significant issue discovered during initial testing involved the Ethernet daughter board. The original design incorporated a LAN8720 Ethernet transceiver. Although this transceiver is commonly used in similar applications, it was ultimately found to be incompatible with the ESP32-S3. This incompatibility prevented successful Ethernet communication and led us to redesign the Ethernet daughter board using the W5500 Ethernet controller, which offers SPI communication and a more reliable interface with the chosen MCU.

Another critical adjustment emerged regarding video output. In the original concept, the design included an HDMI output intended to display video data directly from the MCU. However, further research revealed that the ESP32-S3 does not natively support HDMI signal generation. Attempting to implement this feature would have significantly increased the complexity of the PCB design and introduced a high probability of failure. As a result, the HDMI output section was entirely removed in the second revision, and we instead opted to transmit the video data to a host computer for visualization via serial output.

Testing the first revision of the main board also exposed several hardware-related design flaws. One such flaw was an incorrectly selected diode that unintentionally blocked the 5V power line from the USB port, preventing proper system startup. Additionally, the absence of a programming interface chip, such as the CP2102 USB-to-serial converter, meant that external flashing of the MCU was only possible by using a manually connected CP2102 breakout board through open GPIO pins. While functional, this approach added unnecessary complexity and limited development flexibility.

These early problems made it clear that we needed to think ahead more and do better research during the design phase. In our revised board, we corrected the diode issue, added the CP2102 chip for easier programming, and made sure all the necessary connections were in place. The new design was much more reliable and easier to work with, giving us a solid platform to move forward with.

D. Optical Link Testing

The optical link came to fruition naturally. First, all parts were evaluated individually for sub-system performance. Optical media converters were tested in tandem with optical transceivers and optical fibers in duplex mode, shown below.

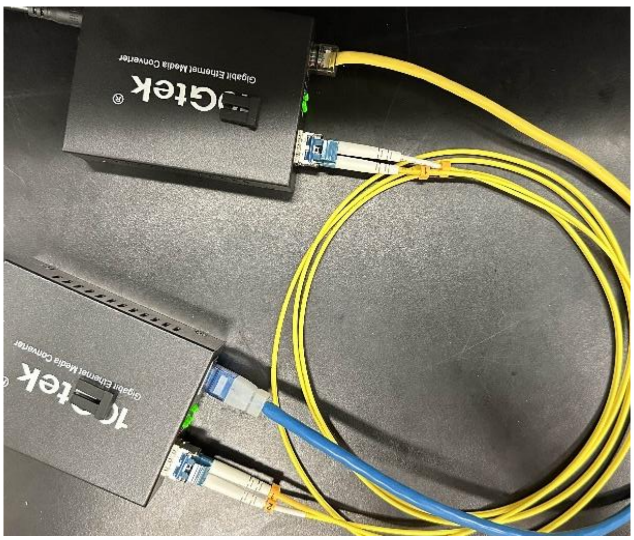


Fig. 11 Duplex optical communication.

This preliminary test demonstrated that optical transceivers are capable of 1 Gbps data transmission speeds when connected to a personal computer. The optical fibers were tested with respect to the output optical power from each transceiver and demonstrated minimal loss on an optical power meter (0.2-0.6 dBm). The fiber collimators expressed an insignificant amount of loss (0.1 dBm) due to the collimators being engineered to fit with the type of fiber and for being AR coated.

Optical alignment proved to be difficult and the main source of noise to the system. A slight misalignment prevents sufficient optical powers from being incident on the detector. However, when optical alignment is achieved over the working distance of 0.25 meters, the system only expresses a loss of 1.5 dBm. With the introduction of the neutral density filter, the detected signal decreases by 3.5 dBm. With MODTRAN simulating atmospheric optical attenuation of 0.1 dBm/km for 1550 nm light, the neutral density filter is emulating a working distance of 35 km. The prototype optical link is demonstrated below.

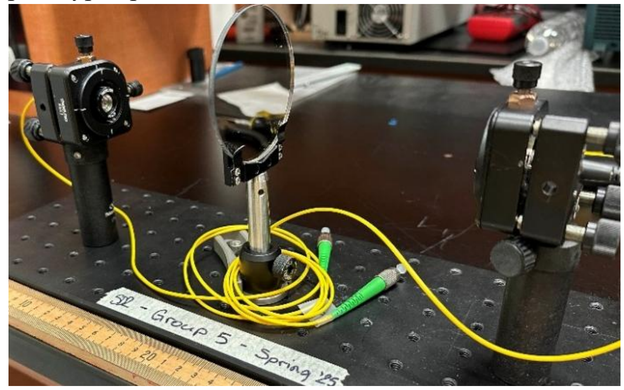


Fig. 12 Optical link

E. FFmpeg Testing

In the beginning of the project, the speed at which FFmpeg could compress was tested using high resolution and high framerate videos. Initially, the tested video input was around 30 seconds of 1080p 60fps video being compressed to 720p 30fps video. This was tested to push the limits of the system by using a higher resolution and double the framerate that we intended to use in our final design. The Raspberry Pi 4 Model B was the initial board tested. This test yielded a result that showed that the board would be able to do real time compression in an efficient manner without much loss of quality.

The second test was to test how to convert video data into bitstream and see how efficient the data conversion is. The tests showed that conversion into bitstream using FFmpeg was more than efficient and would work quite well in our system. A 1080p 30fps video was converted into an H.264 file and then converted into a bitstream file. A separate code was tested to then convert the bitstream back into H.264 file format and then back into a viewable format. This was all able to be done in a timeframe that will be within our needs.

VI. Conclusion

Over the course of Senior Design, we have strived to develop a secure and efficient surveillance system that utilizes Free Space Optical communication and computer vision. Research and tests have been done to ensure that a product is designed that displays innovative solutions to current issues with security systems.

Our group members have acquired knowledge throughout our time here at UCF and during Senior Design. We will use the knowledge we've gained to further our studies and research. We look forward to the development of iSPY and the advancement of security in the coming years.

References

[1] S. M. Kelly, "That security camera and smart doorbell you're using may have some major security flaws | CNN Business," CNN. Accessed: Aug. 30, 2024. [Online]. Available: https://www.cnn.com/2024/03/09/tech/smart-home-cameras-hackers-security/index.html

[2] N. Vigdor, "Somebody's Watching: Hackers Breach Ring Home Security Cameras," *The New York Times*, Dec. 15, 2019. Accessed: Aug. 30, 2024. [Online]. Available: https://www.nytimes.com/2019/12/15/us/Hacked-ring-home-security-cameras.html

[3] "CMOS Sensor Size Comparison Guide and Lens Calculator," Commonlands Optics. Accessed: Sep. 14, 2024. [Online]. Available: https://commonlands.com/blogs/technical/cmos-sensor-size

[4] R. Paschotta, "Field of view." Accessed: Sep. 03, 2024. [Online]. Available: https://www.rp-photonics.com/field_of_view.html

[5] J. M. Geary, *Introduction to Lens Design: with practical ZEMAX examples*, 1st ed. Richmond, Virginia: Willmann-Bell, Inc, 2022.

[6] Junyi Chai, H. Zeng, A. Li, and E. W. T. Ngai, "Deep learning in computer vision: A critical review of emerging techniques and application scenarios," *Mach. Learn. Appl.*, vol. 6, p. 100134, Dec. 2021, doi: 10.1016/j.mlwa.2021.100134.

[7] N. O'Mahony *et al.*, "Deep Learning vs. Traditional Computer Vision," in *Advances in Computer Vision*, vol. 943, K. Arai and S. Kapoor, Eds., in Advances in Intelligent Systems and Computing, vol. 943. , Cham: Springer International Publishing, 2020, pp. 128–144. doi: 10.1007/978-3-030-17795-9_10.

[8] "Body Detection with Computer Vision | Latest | Instrument." Accessed: Sep. 26, 2024. [Online]. Available: https://www.instrument.com/latest/body-detection-with-computer-vision

[9] A. Tam, "Using Haar Cascade for Object Detection," MachineLearningMastery.com. Accessed: Sep. 26, 2024. [Online]. Available: https://machinelearningmastery.com/using-haar-cascade-for-object-detection/

[10] Ultralytics, "YOLOv9." Accessed: Oct. 15, 2024. [Online]. Available: https://docs.ultralytics.com/models/yolov9

[11] S. Arya, "YOLOv10: Revolutionizing Real-Time Object Detection," Analytics Vidhya. Accessed: Nov. 03, 2024. [Online]. Available: https://www.analyticsvidhya.com/blog/2024/07/yolov10-for-realtime-object-detection/

[12] R. Vaidya, "Deci's YOLO-NAS: Next-Generation Model for Object Detection.," CodeX. Accessed: Oct. 18, 2024. [Online]. Available: https://medium.com/codex/decis-yolo-nas-next-generation-model-for-object-detection-8ccb7f2013a7

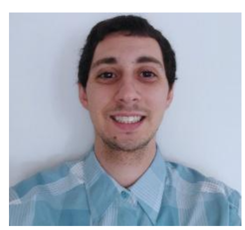
Bibliography

Alejandro Lopez Zelaya is a senior majoring in photonic science and engineering with a minor in bioengineering. After graduation, he hope to attend medical school and prepare myself as an engineering physician.

Armani Garcia is a senior majoring in Electrical Engineering. Plans to enter the engineering workforce immediately after graduation.

Gabryella Baldaci is a senior at CREOL. Her research focuses on laser surface treatment of metal alloys and their applications. After graduation, she is preparing to launch her career in the photonics industry.

Luke Reyes is a senior at UCF pursuing a degree in computer engineering, aiming to enhance his knowledge and skills for future endeavors.



Robert Josephson is a senior in Computer Engineering, interested in biomedical technology. Aiming to work as an R&D engineer in biotech or biorobotics after graduation.