Paper ID #44591

Visualizing the Invisible: Object Detection via Wi-Fi Signal Mapping Emulation

Benjamin Lubina, Gannon University

Benjamin Lubina is currently an undergrad in Cybersecurity at Gannon University, He runs the school Cyber Defense Club, competed in challenges and competitions, and published several papers in the field of machine learning. He has 3 years of experience with software development, cyber risk assurance, and data analysis.

Dr. Ramakrishnan Sundaram, Gannon University

Dr. Sundaram is a Professor in the Electrical and Computer Engineering Department at Gannon University. His areas of research include computational architectures for signal and image processing as well as novel methods to improve/enhance engineering educa

Visualizing the Invisible: Object Detection via Wi-Fi Signal Mapping Emulation

Benjamin Lubina and Dr. Ramakrishnan Sundaram

lubina001@gannon.edu,sundaram001@gannon.edu

Abstract

This paper introduces a software tool designed to emulate and analyze Wi-Fi signal strengths from an array of ESP32 devices. This paper is a continuation piece to another work which outlines the hardware creation of the ESP32 array. This software serves as a companion to the already developed array and is meant to be integrated into the accompanying hardware setup. The primary goal of this tool is to create theoretical images of objects situated within the array by leveraging the variations in Wi-Fi signal strength caused by these objects. We present a comprehensive method that utilizes the unique properties of ESP32 microcontrollers to capture Wi-Fi signal metrics to generate a visual representation of the physical space and the object(s) within it. The inputs to this software mimic those provided by the hardware array and employs advanced algorithms to process the metrics made by the array. This technique, often referred to as Wi-Fi imaging or Wi-Fi based material sensing, has significant implications for various applications, including security, terrain mapping, navigation in visually impaired environments, and smart home systems. The results show our system's capability to detect and visualize objects of different sizes and hint at basic composition of materials. Additionally, the paper discusses the challenges and limitations encountered during the research, such as signal interference and the resolution of generated images, as well as software limitations and integration challenges. Our findings suggest that this Wi-Fi based imaging approach, while still in its nascent stages, holds great potential for various practical applications. The paper concludes with future research directions, emphasizing the need for enhanced algorithms and more sophisticated ESP32 arrays to improve accuracy and resolution of the Wi-Fi imaging process.

Background

The ESP32 Wi-Fi chipset, engineered by Espressif Systems, is a powerhouse in the IoT domain, distinguished by its Tensilica Xtensa® Dual-Core 32-bit LX6 microprocessor, capable of operating at up to 240 MHz. It boasts a substantial 520 KB of SRAM and supports a variety of memory interfaces, including SPI and Quad SPI, enhancing its versatility. In terms of wireless capabilities, the ESP32 excels with its support for IEEE 802.11 b/g/n Wi-Fi standards and dual-mode Bluetooth, making it a comprehensive solution for Wi-Fi networking tasks. The chipset's interface arsenal is impressive, featuring UART, SPI, I2C, PWM, and more, alongside 34 GPIO pins that cater to a wide range of digital functions and sensor integrations. Security is a paramount aspect of the ESP32, with features like secure boot, flash encryption, and cryptographic hardware acceleration ensuring robust protection. Power management is ingeniously designed for efficiency, with the chipset operating in active, sleep, and deep sleep modes to optimize energy usage. This, combined with its adaptability to various environmental conditions and a range of packaging options, makes the ESP32 an ideal candidate for diverse applications, from wearable electronics to complex IoT systems, embodying a perfect blend of power, connectivity, and efficiency [6]. An example of this device can be seen below in Figure 1.

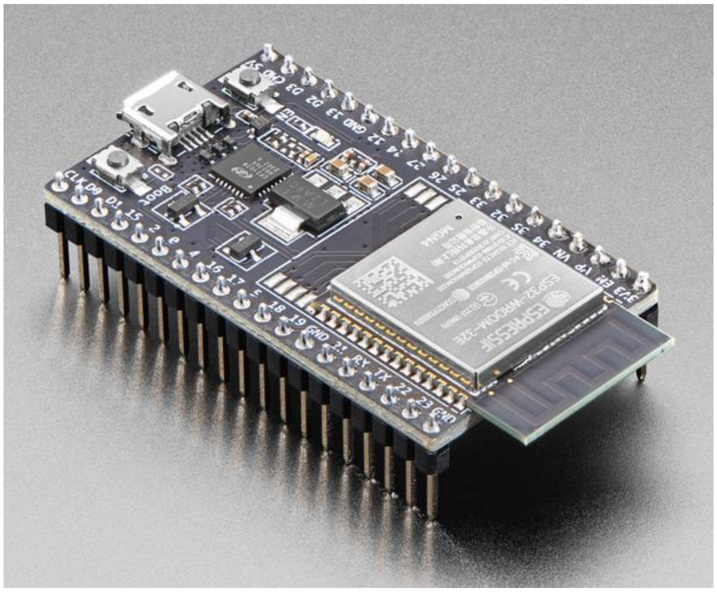


Figure 1: ESP32 Wi-Fi module

Because it is a is a low-cost, low-power system on a chip (SOC), it makes an ideal candidate for the nodes of a wireless sensor network for radio frequency mapping of space.

The exploration of Wi-Fi imaging and material sensing is a rapidly evolving domain, leveraging the ubiquity of Wi-Fi infrastructure and advancements in signal processing and microcontroller technology. This field stands at the intersection of wireless communication, computational algorithms, and practical applications applicable to diverse areas such as security, healthcare, and navigation. Not only this but in the context of using ESP32 microcontrollers for Wi-Fi-based sensing, there is plenty of literature to draw from in this field [1][3].

There are several sensing methods that have appeared in recent years. One foundational aspect of this field is the use of Channel State Information (CSI) for sensing purposes. Intricacies of methods like CSI-based Wi-Fi sensing have been explored, particularly under the modern Wi-Fi 6 standard. This research is crucial for understanding the potential and challenges of Wi-Fi-based imaging, as it sheds light on the impact of advanced Wi-Fi features on sensing performance [1]. Another key development in Wi-Fi sensing is the application of reconfigurable intelligent surfaces (RIS), which exposes a novel framework for enhancing the resolution of Wi-Fi-based imaging using off-the-shelf Wi-Fi devices and RIS. The approach aligns with the goals of the ESP32-based imaging system, aiming to overcome resolution limitations inherent in conventional Wi-Fi devices [2].

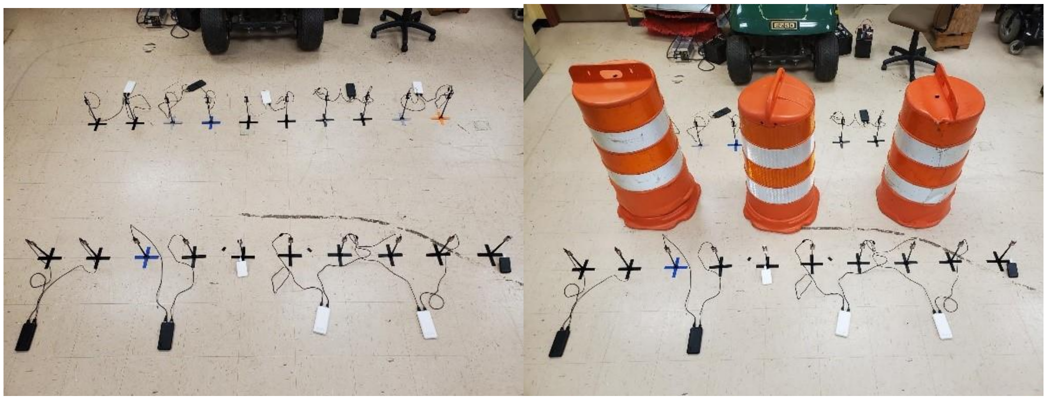
The practical challenges and implementation aspects of using ESP32 microcontrollers in Wi-Fi sensing have significant history and provide a reference or basis on what and what not to do. This information includes real-world deployment of ESP32-based systems, highlighting the discrepancies between simulation and actual performance, which is critical for the development of robust Wi-Fi imaging systems. [3] Furthermore, there already exists potential generic software frameworks for Wi-Fi-based sensing, emphasizing the need for accessible tools in this research area. This aligns with the ESP32-based system's goal of integrating advanced sensing capabilities with user-friendly software interfaces. This broadens the scope of applications for Wi-Fi imaging systems, suggesting their utility in automated and intelligent systems [4][5].

Hardware

The ESP32 Wi-Fi module, as depicted in Figure 2, is securely positioned on a 3-D printed stand, serving as an individual wireless sensor node. This module derives its power from a portable charger, which is connected to its USB port via a USB cable. Multiple of these can be strung together to form a network mesh that will connect to each other and whose strength of connection can be affected by objects being placed within the network. This has been enumerated upon in the RF Wireless Sensor Network guide done previously by Z. Dickinson, T. Seelnacht, and R. Sundaram at Gannon's ECE Department, but will be summarized here [6].

Figure 2: ESP32 Wi-Fi module

In the described setup, 20 of these ESP32 Wi-Fi modules are strategically arranged in a 2x10 array configuration. This specific arrangement is designed to effectively capture images of objects placed between the modules. By utilizing a grid-like formation, each ESP32 module functions as a node within a larger network, collaboratively working to analyze the Wi-Fi signal variations caused by objects in their vicinity. This collective operation allows for a more comprehensive and detailed imaging process, leveraging the combined data from all 20 modules to construct a clearer and more accurate representation of the objects positioned within the array. The 2x10 array formation is particularly advantageous for enhancing the spatial resolution and accuracy of the imaging process, making it a robust setup for detailed Wi-Fi-based sensing and imaging applications. A working setup of a 2x10 array, henceforth called a 10x10 wireless sensor network, is shown below in Figures 3a and 3b [6].



(a) (b)

Figure 3: Wireless sensor network (a) empty (b) obstructed

To coordinate and facilitate this setup, a 300 Mbps Wireless N Nano router is connected to the base station (laptop) through the USB port. Each transmitter and receiver sensor node is programmed differently. One row of ESP32 nodes are designated as the transmitters and are uploaded with TX32.ino code. The other row are receivers and given the receiver code, RX32.ino. Instructions on setup and code base are in the technical guide that was also created by the other students at Gannon University [6].

A Python script to test the operation of the network is loaded on to the laptop or PC at the base station. This Python script retrieves the RSS (strength of signal) values received by each receiver node from each transmitter node using the Wi-Fi router. This information is then extrapolated into a grid and shaded values to construct a visual, the software of which will be discussed in the next section.

Software

In previous prototypes of the grid data visualization program used in conjunction with the wireless sensor network, there were notable limitations, particularly in the analog representation of results and a tendency to produce false positives in detection. These earlier versions lacked the sophistication needed to accurately interpret the complex dynamics of Wi-Fi signal interactions within the network, leading to inaccuracies and less reliable imaging outputs. Figure 4 is an image pulled from this old design showing the described problem. Figure 4 was the output provided from the sensor input in configuration 3b [6].

Figure 4: Obstruction Visualization

The following is a quote from the paper on the primitive software design: “The white shaded areas are the estimated location(s) of the obstructions. In this case, the program can identify that there is some interference in the center of the grid. However, it is not able to differentiate the three separate entities that are physically present. The program also displays two false positives in the form of individual white squares. These errors may be due to the current implementation of the shading algorithm, which gives all intersecting lines equal weight when calculating the value of a specific grid square” [6].

To address these challenges and enhance the overall efficacy of the system, a significant improvement was made in the data processing algorithm. The key advancement involved the utilization of Received Signal Strength (RSS) signals from the transmitters to the receivers within the 10x10 wireless sensor network and developing a proper algorithm to translate the 2 rows into a 10 by 10 grid. The program could more accurately calculate grid values, considering the nuanced variations in signal strength caused by objects within the network's range and potentially allowing for an increase in the array's size. Below is a visualization of this algorithm.

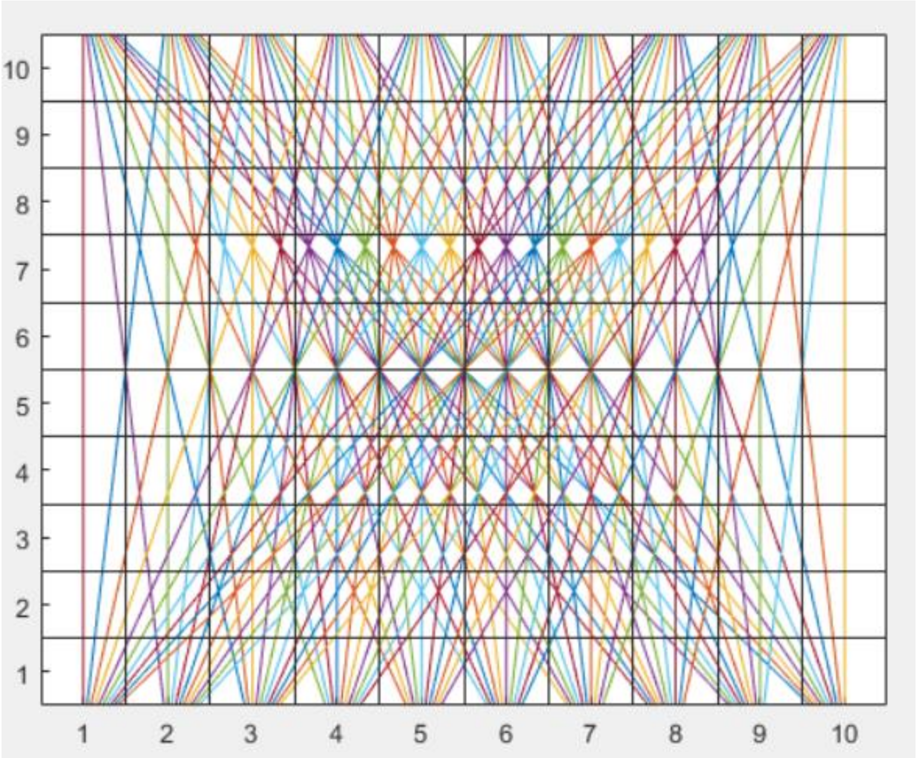
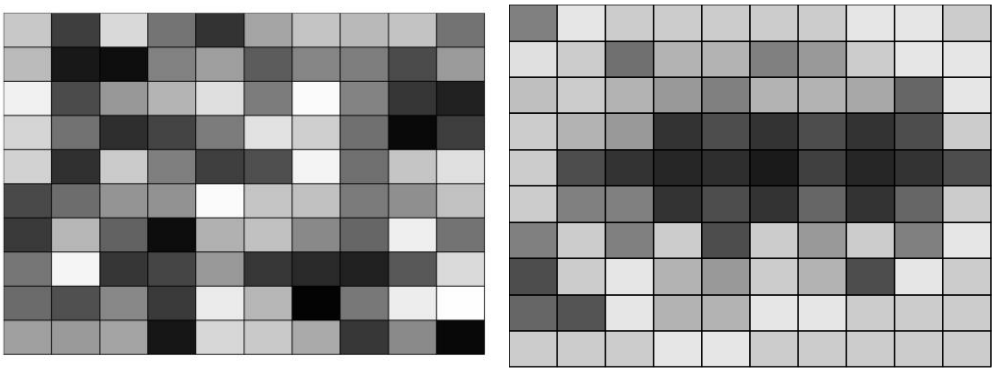


Figure 5: Grid Algorithm Abstraction.

This refined approach allowed for a more precise interpretation of how Wi-Fi signals were being altered by physical obstructions, leading to a significant reduction in false positives and an improvement in the fidelity of the detection process. The integration of RSS-based analog calculations enabled the program to better leverage the capabilities of the 10x10 wireless sensor network, resulting in a more reliable and accurate grid data visualization. This advancement marked a crucial step forward in development. Figure 6a and 6b show the output of a revitalized program that implements these theories. 6a is a randomized output done during initial configuration and 6b is an image made a setup similar to the one shown in Figure 3b. Unlike the original application, the darker squares show greater interference indicating an obstruction in the path of the signal. The various value on this spectrum for the squares in the grid give a better chance of showing an image with greater granularity than previously provided by its predecessor. To achieve this improvement, the new program was coded using Python 3 and the matplotlib and numpy libraries.



(a) (b)

Figure 6: Visualization (a) random (b) obstructed

As seen above, the visualization is far more granular and significantly improves on the original design, but there is noise in the system that gets normalized out of the video while collecting information over time. Our grid system's modularity allows for flexible resizing to match the hardware sensor array, enabling it to adapt to various spatial requirements and application needs. This adaptability is further enhanced by the program's capability to continuously generate images, synchronizing with the refresh rate of the Received Signal Strength (RSS) signal values from the ESP array. This dynamic approach ensures real-time updates and adjustments in the imaging process, providing an up-to-date and accurate representation of the environment based on the current data from the sensor network. The combination of a modular grid and continuous image generation makes our system highly efficient and responsive, suitable for a wide range of Wi-Fi-based imaging and sensing applications.

Conclusion

This program creates a fresh approach on top of a previously built system in object detection using Wi-Fi signal mapping, harnessing the capabilities of ESP32 microcontrollers. Initially introduced as a novel software tool, its purpose was to create theoretical images of objects within an ESP32 array by capitalizing on signal strength variations. This innovative method, straddling the intersection of Wi-Fi imaging and material sensing, opens up a plethora of applications, notably in security, navigation for the visually impaired, education, and smart home systems. Despite facing challenges such as signal interference and resolution limitations, the study has laid a strong foundation for further research. It beckons future advancements in algorithmic sophistication and hardware enhancements to refine this nascent technology. Ultimately, the potential of Wi-Fi-based imaging in transforming our interaction with the surrounding environment is immense, promising a new era of technological innovation and practical applicability.

For future work, the program offers several avenues for enhancement and expansion. Firstly, improving the program could involve refining algorithms for better accuracy in object detection and resolution. This would entail more sophisticated data processing techniques and possibly integrating machine learning models for enhanced interpretability of Wi-Fi signals. Expanding the program's application into education could involve developing interactive tools or simulations that allow students to explore the principles of Wi-Fi signal processing and imaging. This could be an excellent tool for teaching concepts in physics, computer science, and engineering, making abstract ideas more tangible and engaging. Moreover, adapting this technology for drones and mobile applications opens up vast possibilities. Drones equipped with ESP32 arrays could use this technology for advanced terrain mapping, search operations, or agricultural monitoring. Mobile applications could leverage this technology for indoor navigation, particularly aiding visually impaired individuals in unfamiliar environments. These expansions not only enhance the practical utility of the program but also broaden its impact across various sectors, from education to real-world applications in diverse fields.

Bibliography

1. M. Cominelli, F. Gringoli and F. Restuccia, "Exposing the CSI: A Systematic Investigation of CSI-based Wi-Fi Sensing Capabilities and Limitations," 2023 IEEE International Conference on Pervasive Computing and Communications (PerCom), Atlanta, GA, USA, 2023, pp. 81-90, doi: 10.1109/PERCOM56429.2023.10099368.
2. Y. He, D. Zhang and Y. Chen, "High-Resolution WiFi Imaging With Reconfigurable Intelligent Surfaces," in IEEE Internet of Things Journal, vol. 10, no. 2, pp. 1775-1786, 15 Jan.15, 2023, doi: 10.1109/JIOT.2022.3210686.
3. V., M. P. J., de Souza, B. J. O., Lamenza, T. de S., & Endler, M. (2022, November 19). Practical challenges and pitfalls of bluetooth mesh data collection experiments with ESP-32 microcontrollers. arXiv.org. https://arxiv.org/abs/2211.10696
4. Kindt, P. H., Turetta, C., Demrozi, F., Masrur, A., Pravadelli, G., & Chakraborty, S. (2022, April 6). WiFiEye -- seeing over WIFI made accessible. arXiv.org. https://arxiv.org/abs/2204.01830
5. Hunter, W., Arun, A., & Bharadia, D. (2023a, May 22). Wiros: WIFI Sensing Toolbox for Robotics. arXiv.org. https://arxiv.org/abs/2305.13418
6. Z. Dickinson, T. Seelnacht, and R. Sundaram, RF Wireless Sensor Network, Technical guide, 2022.