Magic Sensors

Team Name: sddec21-09
Members: Abdelaziz Hassan, Alyssa Marshall, Ben Pierre, Chinar Kaul, Jared Hermon, Ryan Lanciloti
Advisor/Client: Dr. Daji Qiao

Problem Statement

- Currently in the market, door sensors (used in home systems) are battery power or need to be wired to a control unit
- The goal of this project is to design and implement innovative “magic sensors” which are wireless a batteryless
- These “magic sensors” will report the status of the door (whether it is open or closed) to control unit with power source

Users

- Anyone who wants to install the security system and then never touch it again. The only interaction needed between the user and the system is arming and disarming the system via the application on their phone

Uses

- This product will be deployed by homeowners, business owners, or property owners who wishes to have a non-powered, tamper-resistant system of detecting door states

Hardware Testing

- Verification was done to ensure that all of the wires were connected correctly between the esp32s

Software Testing

- A flashing blue led was used to indicate if the esp32 was sending data to the server
- An acceptance testing document was used to ensure that same testing procedures were followed each time

Results

- Testing accuracy of machine learning model = 40-60%

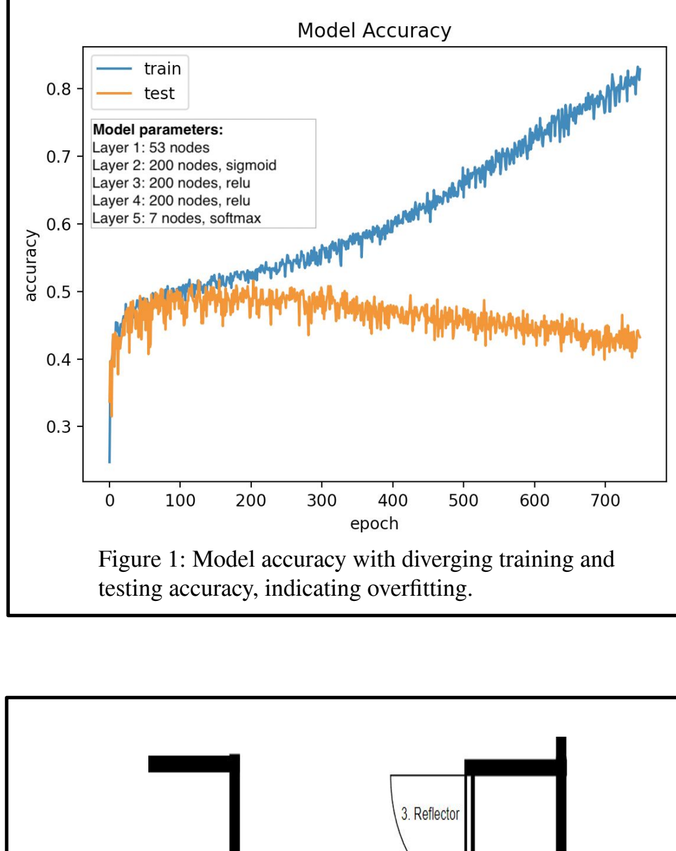

Figure 1: Model accuracy with diverging training and testing accuracy, indicating overfitting.

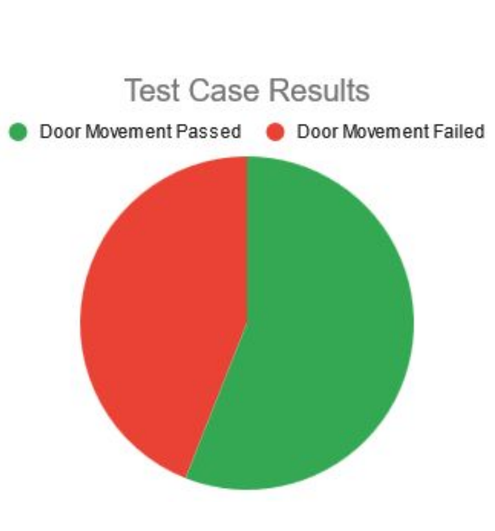

Figure 2: Test case results.

Functional Requirements

- Detect a door’s state up to 10 feet away from the base unit
- No powered no wired sensor ot the door
- Capable of arming and disarming system
- Detect door open or closed with 95% accuracy with false-positive and false negative reports falling under 5%

Non-Functional Requirements

- $300 budget for total system
- The door module must be less than $30 to allow a user to add more sensors to their system in the future
- UI must conform to current design stylings
- The door module must weigh less than one pound and be less than 4” by 4” when installed

Operating Environment

- System must function in environments with noise, ie household appliances, pets, and other objects

Standards

- **Circuit**
  - US power circuit wiring color codes
- **Hardware**
  - CISPR and IEEE Electromagnetic Interference Standards
- **Software**
  - PEP 8 and Arduino Styling Guides

Primary Teams

- Front End Android Team
- Back End Team
- Data Collection/Hardware Team
- Testing Team
- Machine Learning Team

Iowa State Resources

- Electronic and Technology Group (ETG)
- Dr. Mani Mina, Dr. Adisak Sukul, Dr. Andrew Bolstad

External Resources

- Espressif API’s
- “ESP-IDF” Jonathan Mueller[Online]
- “ESP32 CSI Toolkit” Steven M. Hernandez[Online]
- Arduino IDE, ESP IDF, Python, Flask Server, Keras

Design Approach

Figure 3: Map of system components.

Figure 4: Receiver and relay ESP32s.

Figure 5: Server ESP32s.

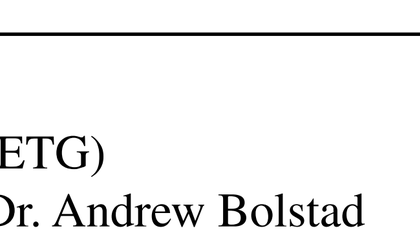

Figure 6: System diagram.

Functional Modules

- ESP32: One ESP32 establishes an access point (AP) and and the other ESP32 connects to said Access Point
- Aluminum: This is our reflector material and it reflects the RF waves most effectively for our purpose
- Machine learning: This component will receive packets from our python relay and interpret them to determine door states
- CSI: Channel State Information included within WiFi to track the movement of a door to determine if it’s open or closed. we use three transmitters and one receiver to send and receive information about angle and time of flight of the WiFi signals as they propagate through the room
- Server: This component links the others, does all of our business logic, and handles metrics, performance, and notifications to the end user