AI in Home Security: The Potential of Sound and Facial Recognition AI

Jayant Rathi*, Jerapong Rojanarowan and Narong Aphiratsakoun

Vincent Mary School of Engineering
Assumption University of Thailand
Bangkok, Thailand

Abstract: This paper explores the usage of AI in home security, specifically focusing on Sound and Facial recognition AI in a home security robot. The purpose of this study and the project was to explore the ways in which AI could enhance home security and monitoring to create a more comprehensive approach towards preventing home emergencies as well as intrusions. Through a review of existing studies and a prototype robot, we were able to ascertain that the use of AI can significantly improve the accuracy and reliability of home security systems with multiple advantages over basic CCTV systems used widely today. This study highlights the importance of being able to equip everyday homes with accessible AI tech and the advantages to homeowners with the added piece of mind and reduced risk of leaving homes or dependents unattended. The implications of this research extend far beyond the fields of home security, as the use of AI becomes increasingly more common in everyday life.

***Keywords*—Artificial Intelligence, Robotics, Facial Detection, Sound Recognition, Line Tracking**

I. Introduction

AI models in security have mostly focused on working with CCTV cameras, working to track/detect motion like Spot Cam, or in basic door lock facial recognition systems. In addition, the usage of robotics in Home Monitoring was quite limited until the release of the Amazon Astro bot [1]. The Amazon Astro bot includes features like home monitoring using a smartphone app and cameras on the robot, glass breaking sound detection, and smoke and monoxide detectors. It is also great for home and pet monitoring as you can direct it to go into specific rooms or towards specific people and you can interact with them using the video calling features. The robot can map out the house, homeowners are able to tell it to avoid certain areas, and stay put in certain others.

It is a convenient solution to a home monitoring problem that would otherwise have to be solved by cameras. The issue with cameras is that without AI, they can only observe, and the anxiety that a homeowner will have to go through every moment he is not checking the camera feed. This anxiety reduces when he knows that in the case that an intruder is detected or a dangerous sound is detected, he will be notified and he will then have the ability to look into his house.

We believed that an approach involving AI will not only be more effective than a simple CCTV setup at protecting a home, but would also be easier and more user friendly for the home owners to interact with.

For an AI home security system to work it must be easy enough to use daily and reliable enough to be able to provide the customer with a sense of peace when they are away. In order to do so you need to be able to cover multiple bases, which is why our focus was Sound and Facial Detection with the ability to simply notify the homeowner if any sort of danger was detected. We created a movement system for our robot that would allow it to travel around in a loop to be able to stop and focus on certain "Action" areas that would be the most probable locations if any danger did exist.

The manuscript received November 20, 2023; revised December 21, 2023; accepted December 25, 2023; Date of publication December 30, 2023

*Corresponding author: Jayant Rathi, Vincent Mary School of Engineering, Assumption University of Thailand, Bangkok, Thailand (E-mail: jayant12rathi@gmail.com)

II. Home Security Robot

A. Danger Sound Detection AI

For our Sound AI, we started from scratch with a basic understanding of Yamnet and TensorFlow. Using transfer learning from Yamnet (a pre-trained Deep Neural Network model trained on 521 sounds) to extract embeddings from sound files, we tried to create a new model using Keras and TensorFlow libraries to detect specific sounds that we wanted to be able to recognize such as Glass Breaking, Baby Crying, Body Falling, Screaming, Coughing and ambient noises. We needed to use Transfer learning to change the models, as the previous pre-built ones were not specific to our use case and so not nearly as accurate or fast as we needed them to be. [2]

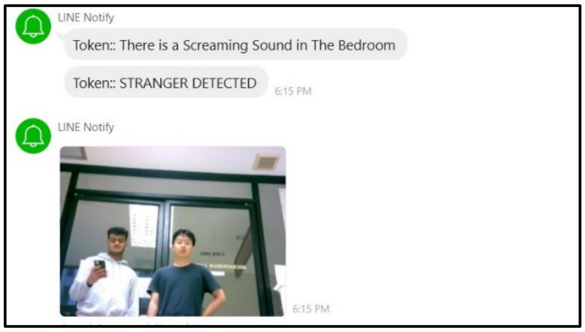
With this model created, we then went onto implementation. Using a microphone, we had to collect sound bites, process them into 1-second chunks of Mono Wav files. After which they were put into the model which extracted its embedding’s and was then ready for classification. This model gave us a satisfactory accuracy of 74%. However, upon further research it was evident that using a SVM model after extracting the embeddings from YAMNET would provide a better result, and upon its implementation, our accuracy results went up to 80%. With the model prepared and the testing complete, we then implemented the entire system on a 4GB RAM Raspberry Pi and connected Bluetooth microphones as our sensors. A Power bank powered the Raspberry Pi, the Camera and Microphones.

This system runs constantly while the robot is on and it scans for danger sounds whenever the system is on and whenever one of the danger sounds are detected the entire process of monitoring begins and it prompts the robot to move towards the area where the microphone is placed.

B. Facial Recognition AI

For our Facial Recognition AI we relied on the OpenCV libraries and through transfer learning made a model suited for recognizing the faces of 3 members of a family (in this particular case). Once our model was made we moved onto the implementation within our robot. We knew that the camera had to be able to initiate without much time delay and be able to run at a suitable frame rate to be able to detect faces while the robot was moving around the house to the specific action locations.

As soon as the Sound AI identifies a dangerous sound, the Facial Recognition program would initiate and with a delay of 3 seconds, during this time, the robot will start moving into position. The Camera (720p and 60Hz) and AI then start to scan for faces, once having captured a face it will go through the model and try to classify the face. If classified, a notification will be sent to the homeowner with a picture stating that a specific member of the family is at home. However, if the face detected is not recognized, then a notification is sent to the homeowner with an alert that an intruder is in a particular part of their home.


LINE Notify

Token:: There is a Screaming Sound in The Bedroom

Token:: STRANGER DETECTED 6:15 PM

LINE Notify

6:15 PM

Fig. 1. Output Messages on LINE

C. Robotic Movement System

For our robots’ movement system, we relied on a line tracking based system where in a loop was created, with multiple doors/action points in mind for the robot to be able to stop and monitor in multiple areas (such as doors or windows in a house) the homeowners mobile. We made multiple positions that the robot can stop at and monitor as well as a base where the robot can rest. The demarcation of these different points is done through colored tape and a color sensor.

For the Hardware used in this aspect, we ran the program off of an Arduino Uno which was connected to its own independent 2000Mah Lithium Battery Pack. We connected the Arduino to 2 DC brushed motors to power the two back wheels and the front was supported by a singular free wheel. Rotation was achieved by tuning the speeds of the two power motors to allow the robot to change direction by making one wheel run faster than the other.

We used 4 IR sensors to ensure that the Line Tracking along the movement was as accurate as possible to allow the robot to have a smooth line throughout the “Test Area”. The Multiple IR sensors allowed us to accurately gauge when the robot was wearing off course so that it would quickly be able to correct itself without being too far from the intended path route. Along with the IR sensors, A PID sensor was also used to ensure an even smoother operation of the movement system.

D. Integration and Implementation

One of the biggest challenges for us was the integration of three completely separate programs to come together to work seamlessly in a singular system that depends on communication between all three aspects so that all features would work well with each other.

As the system starts, the robot will only be asked to move from its base if the microphones kept at the corners of the home identify a dangerous sound. After that takes place, the robot will move along the paths to the right color spots/areas where the sound was heard, during this process the camera and Facial AI will be initiated to ensure that everything is running properly by the time the robot reaches the correct spot. Once the camera has been running for 15 seconds, the Face AI will stop and the focus is diverted back to the sound AI that will have continued to work this entire time. After this 15-second period, the robot will return to the base position and switch off the camera and facial detection AI.

III.Application and Advantages of AI

By utilizing Artificial Intelligence in our project, we were able to automate and improve various aspects of home security systems. We were able to utilize libraries such as TensorFlow, Yamnet, PyAudio, wave, librosa, numpy and more for Sound Detection AI, and libraries like pickle, CV2, OpenCV, face_recognition, and imutils for the Facial Recognition AI.

A. Danger Sound Classification AI

For the Sound Recognition AI we used a SVM model that allowed us to create a “one vs many” algorithm, that worked on the basis of identifying what each particular sample sounded the most like, based on the training data. If a particular sound sample sounded close to 75% as similar as a specific trained category, it would be classified as such. If no sound can reach the 75% mark on any of the 5 main classifications, then it will be registered as “Noise” and treated as ambient noise.

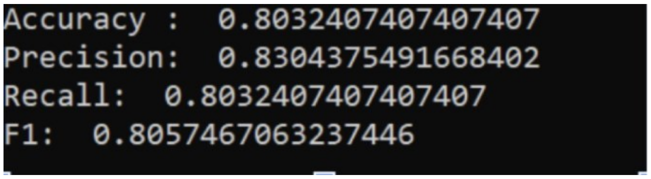
In order to train the model, we had to collect over 2200 one-second chunks of sound data for each of the 6 categories. All this data had to be preprocessed into Mono Wav files to be inputted into the model for training purposes. After this process with a 100-epoch training system, we were able to test the data on a validation and test dataset that gave us satisfactory results

In the Context of our Danger Sound Detection AI, We had to go in depth and understand how sound signals can be visualized and analyzed. We had to understand many concepts about the Fast-Fourier Transform[3], the Mel Spectrogram and its related techniques and tools, in order to analyze the frequency content of an audio input. An Audio signal consists of different single frequency sound waves that can be captured as individual amplitudes over time. The Fourier Transform allows us to segment the signal into its individual frequencies and amplitudes, thus converting it from the time domain, to the frequency domain. This output is a Spectrum that shows the signal’s frequency content and allows the computer to visualize it better. What we also encountered was trouble in the sound detection system in a noisy environment due to the sensitivity of the microphone used. Many a loud noise would be picked up as a scream, but in order to remove

To correctly compute the Fourier Transform, we can use the FFT (Fast Fourier Transform), which is a commonly used algorithm in signal processing. By applying FFT on overlapping windowed segments of the signal, we can obtain a Spectrogram that shows how the signal’s frequency has changed over time. In most Sound Detection AI systems, the Mel scale [4] is a crucial tool that converts frequencies to a scale that is more consistent with how humans perceive different frequencies. The Mel scale is based on the idea that equal differences in pitch are perceived as equally spaced by the human ear. By grouping frequencies into bins that are equally spaced on the Mel scale, we create a Mel Spectrogram that shows how the energy is distributed across different frequency bands

Once the sound signals are converted into a format that is more consistent with how humans perceive sound, the AI system is trained to understand and differentiate the patterns in a Mel Spectrogram to detect and classify different types of sounds. This technique is used with every single sound file that passes through our sound model in order for it to understand it. In its real time application that we utilized in our project, as sound is picked up by the microphone, it is saved into a specific folder. There it goes through the preprocessing format where it is cut up into 1 second chunks and saved as Mono Wav files. From this point onwards, the model picks up each file and runs it through a process where YAMNET extracts the embedding (Mel spectrograms) from these sound files, and our model then tried to classify the sounds using these “Visual representations” of the sounds detected.

After the classification is done and one of the categories is found to be suitable, the output is provided by the program in the form of Line notifications and a status screen on the computer running it. After this, the file is deleted to make room for the next second’s sound file. This all happens in real time to be able to provide accurate and fast sound classification. By using these techniques, we were able to create a Sound Recognition AI that could effectively replace a watchman/guard that has to be placed at maybe even multiple points around a house. Using this AI based system with the high accuracy of 80% allows the home owners to have a peace of mind considering that fact that even if something is wrong with their family members, such as an elderly member taking a fall or a child crying, they will be alerted and they will be able to check up on them. Whereas in most security systems in use today, that is simply not a feature that is utilized.



```
Accuracy :  0.8032407407407407
Precision:  0.8304375491668402
Recall:  0.8032407407407407
F1:  0.8057467063237446
```

Fig. 2. Results of SVM Sound AI Model

B. Facial Detection AI

For our Facial Detection AI, we use the help of the OpenCV Python Library. It is a popular open-source computer vision library that consists of a wide range of image processing and computer vision algorithms. With its widely used API and its large community of support it seemed like the best option to employ.

Using OpenCV we work on our images by using the concepts of HOG or Histogram of Oriented Gradients [5], It is a popular feature descriptor used for object detection and image recognition tasks. The HOG descriptor allows the AI to understand the "Image" provided to it in a "language" it can understand. The HOG descriptor is calculated by dividing an image into small cells and computing the gradient orientation within each cell. The gradient orientations are then binned into a histogram and normalized to account for variations in lighting and contrast.

The resulting histogram of gradients can be used as a feature descriptor for facial detection. This HOG descriptor is what we use to train and test and implement in our model, each image that is received through the USB webcam is analyzed in such a way that the program is able to understand its change in gradients and directions which allows it to understand patterns within someone's face. That is what allows our model to detect the difference between two different faces and recognize one that it has been trained on.

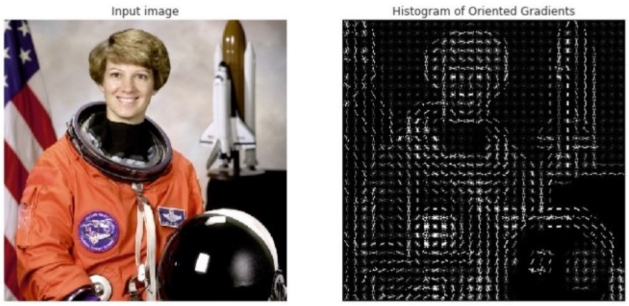
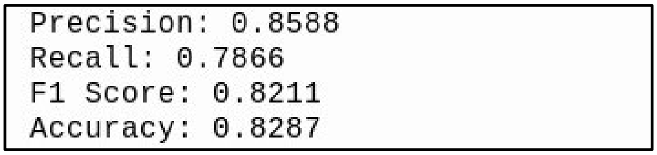


Fig. 3. HOG depiction of Picture

With HOG being a relatively simple and computationally efficient method of feature description, it is well suited for real time image and face detection. For the training process we utilized 3 known faces with 680 pictures each, and while testing we were able to achieve the satisfactory results of 82% accuracy. The HOG descriptors and the "mapping" of the picture are analyzed into a pattern that represents a certain face. This "encoding" of that face is then saved within the memory of the model as a family member.



```
Precision: 0.8588
Recall: 0.7866
F1 Score: 0.8211
Accuracy: 0.8287
```

Fig. 4. Results of the Facial Detection AI

When analyzing pictures in real time as they come in from the camera, the program records a box shaped frame around a face that it detects. This frame is then taken to the model where it is then turned into a HOG descriptor and the pattern of the histogram created is then compared to the ones that it has been trained on and remembered as family faces. If the pattern found is not recognized, then a Line message is sent to the home owner and an output text along with a picture as captioned with the label "Intruder" and sent to the homeowner as well. In our case, the system turns on as soon as a sound is detected, it runs for 20 seconds and then shuts down by itself as it leaves the "action" spots and heads back to the base. Within these 20 seconds, if it does detect a face, it will take 1 picture every 5 seconds if it notices a family member and every 2 seconds if it is not.

Certain Issues that we faced included the concept of facial detection around different backgrounds, especially with varying background light levels, because of this we choose to stick with only activating the camera near a set location or the door rather than have it on and detecting in multiple areas.

C. Movement Program

We used a combination of PID and IR sensors to ensure that the movement system stays as stable and smooth as possible. The basic principles involved in an infrared sensor involve a photodiode or phototransistor (infrared detector) which can detect the presence of infrared radiation. Whenever the "Infrared radiation" is detected, the sensor produces an electrical signal that can be used to trigger actuators, in our case, the motors that drive the wheels.

The IR sensors in our robot are placed facing the ground and are purposed with keeping the robot on the black line that can guide it through the "Test Area". As we utilize 4 IR sensors placed parallel to each other on the front of the robot, we run them on the basis that the 2 inner sensors should always be detecting the black line, however if the outer 2 sensors do detect black, something is wrong with the path and the motors need correcting. If the sensor of the far left detects a black line, the left motor's output is increased slightly, and likewise for the right sensor and motor. These methods are utilized to ensure smooth movements along the line. The PID sensor [6] is an extra addition that we believed truly helped the robot move more smoothly and efficiently especially when going around corners. The PID or Proportional-Integral-Derivative, which is a control algorithm widely used in automation and control systems. It is used to maintain a process variable at a desired set-point by continuously adjusting a control variable, in our case, the speed of the motors. The "proportional term" is calculated based on the difference between the current process variable and the desired set point. The larger the difference the larger the proportional term, which causes the control variable to adjust more aggressively. The Integral term considers the history of the error between the process variables and set points. It calculates the integral of the error over time and adjusts the control variable based on accumulated error. This is useful for long term errors and reducing steady state errors. The derivative term calculates the rate of change of error between the process variables oscillations in the system. This term is used to predict future changes which helps to reduce overshoot and oscillations in the system.

This ability to provide an input that is variable and gradual leads to a much smoother operation than the ON or OFF inputs of an IR sensor. This is why we used a PID algorithm in our system to achieve much more stable movement. The following PID equation allows us to understand just why it is a better algorithm in a dynamic situation like ours.

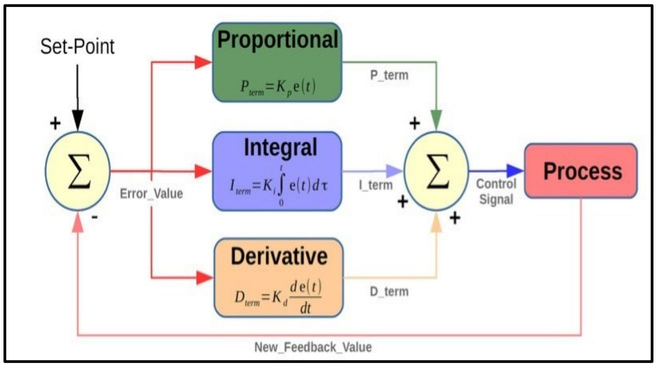


Fig. 5. PID Controller Implementation

IV. Conclusion

We would like to push forward with this system by expanding its usage and feature set, involving things like Self mapping for its own movement system, as well as increasing the number of "action points" to be monitored along with the involvement of Motion detectors placed near the sound sensors. We ran into trouble when integrating all the three different systems together but were able to create methods of communication that suited us best, as a best practice the programs should all be written on the same programming language and if possible, on the same computer. During our project we were limited in our processor's ability due to worldwide chip shortages, this meant we had to split the workload between a PI and an Arduino UNO.

In conclusion, this study was able to explore the application of AI in Home Security, specifically focusing on sound and facial recognition AI integrated into a home monitoring robot. The research demonstrated that AI technology can significantly enhance the accuracy, reliability and user-friendliness of home security systems compared to a traditional CCTV setup. By utilizing Sound Recognition AI as well as Facial Recognition AI the robot can detect dangerous situations and intruders providing homeowners with notifications and peace of mind with enhanced safety features. The integration of these AI components into a robotic movement system allows for a complete home monitoring setup with fast response capabilities. This research highlights the importance of accessible AI tech in everyday homes and its broader implications as AI becomes more and more relevant in all aspects of life.

References

[1] https://www.amazon.com/Introducing-Amazon-Astro/dp/B078NSDFSBJ (October 2023)

[2] https://www.tensorflow.org/tutorials/audio/transfer_learning_audio (November 2023)

[3] https://towardsdatascience.com/fast-fourier-transform-937926e591cb (November 2023)

[4] https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53#:~:text=A%20mel%20spectrogram%20is%20a,converted%20to%20the%20mel%20scale (November 2023)

[5] H. Meddeb, Z. Abdellaoui, F. Houaidi, "Development of surveillance robot based on face recognition using Raspberry- PI and IOT" , Microprocessors and Microsystems, Volume 96,2023,104728,ISSN 0141-9331

[6] X. J. Blake , N. Aphiratsakun, "Comparison between Regressive and Classifying Neural Networks for PID Controlled Path-Following", 19th International Conference on Electrical Engineering/ Electronics, Computer, Telecommunications and Information Technology (19th ECTI-CON 2022), May. 24 - 27, Prachuap Khiri Khan, Thailand