INTERNATIONAL JOURNAL OF INNOVATIONS IN ENGINEERING TECHNOLOGY AND APPLIED SCIENCES (IJIETAS)

INTRUDER DETECTION SYSTEM USING DEEP LEARNING

[1]Nikhil Rajput, [2]Shubham kumar, [3]Tejaswi Pratap

[1]Sharda University,Greater Noida, India, 2021420883, nikhil@ug.sharda.ac.in
[2]Sharda University, Greater Noida, India, 2021403019, shubham@ug.sharda.ac.in
[3]Sharda University,Greater Noida, India, 2021422266, tejaswi@ug.sharda.ac.in

ABSTRACT

A higher standard of living is primarily dependent on the safety of people and their possessions. Much progress has been made in the areas of making homes automatic and more secured. The home environment has seen increased remote monitoring, home security, and appliance control because to technological advancements and the (IoT). In order to track activity inside the user's house and offer a report, a number of home automation systems have been developed. Modern home automation systems include cameras and motion detectors for home security. However, one of the biggest challenges still lies in the logical component of avoiding bogus or superfluous messages. Smart home automation is effective because of intelligent monitoring and response.Utilizing a deep learning model, a technique is developed to enhance the smart home automation system intruder detection and reduce the likelihood of false alarm. Based on his walk, a person seen on video is either categorized as an intruder or a resident of the home. A PIR motion sensor, an ESP8266 development board, a 5 V four-channel relay module, an ESP32 security camera, and the recommended method's prototype were all used in the construction of the system. A CNN model experiment using human motion patterns was carried out to assess the classification for the identification of people.

Keywords: CNN, CosMos, SMTP, SVM.

DOI:

1 Introduction

Attacks, theft, and burglaries committed at homes annually increasing. In South Africa, a number of homes were attacked and burglarized despite the lockdown and stay-at-home orders [1]. The research in [1] indicates that assaults on people and their property have advanced to include sexual assaults. This has demonstrated the necessity for higher and extra degrees of Residences, structures, and individuals are all secured. It seems inevitable that this discipline will be used to address security issues that arise in smart homes environment, given the rapid industrial expansion of smart home automation system and the number of researches being done in the area [4]. Tenants of a safe house will be able to live in peace and security wherever they choose to call home.

A System that alerts homeowner of potential danger without raising a false alarm is essential in today's world, remote control over the house, and an excellent inside view are all desirable. The internet of things (IoT) and information technology (IT) advancements have created platforms that are suited for controlling, monitoring, and securing houses and other properties [8]. The (IoT) is a system that links multiple physical objects, provides network connectivity, & uses a communication channel to transmit information from device and gadgets to people [2, 3]. Thanks to IoT technology, devices, sensors, and appliances can now interact, communicate, and share data. In (IoT) space, autonomous, independent systems are greatly benefited by artificial intelligence (AI). In order to live comfortably, smart home automation must include home security. AI models and certain IoT technologies may be used to remotely monitor, control, and check the conditions of a house. For better decision-making on behalf of humans and a higher quality of life, machine learning and deep learning model may be applied. Machine learning and deep learning models perform well in the category of object recognition and classification [7]. In a smart home environment, the use of AI to homes automation will help with intruder classification & detection. In artificial neural network for image processing and recognitions, a convolutional neural network model is a subclass. When utilizing a large number of convolutional layers, the CNN method is widely used to handle image-based problem [9]. A security measure that can be implemented in a domestic setting is based on motion detection and monitoring. Images taken by the security camera can be processed using the CNN model depending on the region of interest for detection. Before raising an alert by categorizing detected motions as belonging to residents or intruders, an improved smart home automation system can use deep learning model-based intelligent detection [9]. The quality of life is improved by such an effective design for a home automation system, which also lowers stress and waste of essential resources like water and power.

Suggested in order to increase a deep learning model (CNN) to categories human movements pattern [11] as a security mechanism for the identification and classifications of people, and make significant research contributions in the context of smart home automation is proposed[14]. The suggested categorization of movement patterns in the context of a smart house aims to distinguish between regular residents and invaders. The system has prerecorded the motion patterns of regular home occupants, so when a motion pattern is seen, it compares to the previously recorded motion patterns and, before raising the alarm, categorizes the item as either a typical house inhabitant or an invader.

An android smartphone is used to control the house, and the smart homes automation application's GUI allows users to observe all of the house's conditions [10]. The mobile application stores sensor data generated in real-time on a platform as a service and outputs a graphical representation of the ecological finding [5]. In addition, the prototype implementation's IoT hardware selection is based on affordable, expandable, and accessible devices.

Through ambient intelligence, the smart home automation system regulates the atmosphere, home appliances, entertainment systems, and lighting [6]. Wide-range connections for home management, intelligent decision-makings on the part of system, storage, accurate motion detection, and real-time data storage for future forecasting, analysis, and decision-making are still issues in the field of smart home automation. We suggest real-time, low-cost, cloud-based smart home automation systems based on an android mobile application to do this. It should be emphasised that when we use the term "low cost," we really mean that the prototype version of recommended systems was built utilising inexpensive IoT hardware [6], implying that it is both economical and easy to set up. Microcontroller boards, sensors, cameras, and other parts are included in this hardware.

Related Work

Impressive advancements, inventions, experiments, and implementations have been developed in the study of smart home automation. Due to improvements in Information & IoT technologies, smart home automation systems now provide services beyond the basic control of a home and environmental monitoring. Intelligent home automation systems have use machine learning and deep learning techniques systems to improve the intelligence responses. The use of control methods such as Bluetooth, GSM, SMS, Zigbee, and others, which have the drawback of having a limited range of coverage, is an illustration of a research fault. The use of SMS and web-based technology for controlling appliances are further study issues that have been noted. Last but not least, there aren't many study studies that have built intelligent decision-making into the systems but we tried to get as much as we could so that the updates made in this field can be displayed by just creating some comparison based on the features and advantages of previous technologies and the prototype we have designed. The cloud-based system is very efficient in manipulating the data stored so as to present a clear view of what the deep learning algorithm needs[13]. By giving it to the system we just need to store as much information we can and there is no worry about the space required to make such a big data available for the system. There are some relative models which can be used in our prototype but we go with CNN because it makes it easy to get the task done at several front in real time. It is very effective in such type of situation and easy to implement.

Detailed Methodology

A prototype configuration was used to put our technology into operation. Hardware and software components were used in the prototyping process. As far as software goes, we utilized the Arduino IDE and Blynk. Blynk was used in the designs & setup of the Android-based mobile applications. A platform called Blynk allows users to create and deploy mobile applications for Internet of Things systems. In addition to the GUI of our mobile application, the Blynk platform [3] was employed in the design of our hardware for homes administration and monitoring. On the cloud, real-time data collected by sensors may be shown and visualised. The setup was made using the Arduino IDE. Instructions for sensors, other household equipment, and the microcontroller board. The Arduino IDE is multi-platform software that runs on Windows, Linux, and Mac computers.

Additionally, using the Arduino IDE, commands for controlling home appliance were developed and uploaded to the device.

Hardware Component

The 5 V four-channel relay modules, an ESP7266 Wi-Fi module, an ESP32-CAM module, an Arduino UNO board, jumpar cables, breadboards, USB cables, LED, resistors, and sensor make up the hardware of our system. Jumper cables were used to connect two or more items of equipment from one gender to the other. The hardware components were connected using breadboards, and the current flow was reduced using resistors. The user may access their house from anywhere thanks to its ability to connect to the internet and connect to the developed system. The ESP7266 Wi-Fi module may be utilized with sensor and other homes item because of its storage capability. The Wi-Fi module and microcontroller board in our project were both constructed using the ESP7266.The prototype The LED indicator on each relay module channel lets you know whether the designated port is active. A motion-detecting sensor called an HC- SR501 is used to detect motion in Internet of Things situations. It is used in automatic light switching systems, garage doors, gates, and security systems. The PIR sensor is used by our system to detect movement, and it promptly notifies the user that an action has to be made. It is a reasonably priced, sensitive, user-friendly, and reliable sensor for motion detection. It features an internal voltage regulator and facilitates wide-range communication. When using video streaming to prototype our work, we employ the ESP32-CAM to ensure security. The user may monitor the situation at home after receiving a motion alert using the phone interface. Since the ESP32-CAM lacks a programmer chip, the camera board was programmed using an Arduino UNO board.

Datasets and CNN Architecture Training

The CNN was trained using a home training dataset that had roughly 4000 examples of various human stances. We identified four key postures that are indicative of what the CosMos architecture's camera module is anticipated record for image recognition. Sample was divided into groups representing an intruder and a resident of the residence for each pose. Eight classes were produced consequence, which were employed in the CNN architecture's multiclass classification [11].

Accuracy of Model can be calculated using the formula given below:

$$Accuracy = \frac{True_{positive} + True_{negative}}{True_{positive} + True_{negative} + False_{positive} + False_{negative}}$$

And Precision can be calculated as:

$$Precision = \frac{True_{positive}}{True_{positive} + False_{positive}}$$

Where

Truepositive = Correctly classification a right sample Truenegative = Correctly classification a wrong sample Falsepositive = Incorrectly classification a right sample Falsenegative = Incorrect classification of wrong sample

Comparison b/w CNN & SVM & Viola-Jones Algorithm [12]

Three distinct strategies for classifying images exist: CNN (Convolutional Neural Network), SVM (Support Vector Machine), and Viola-Jones Algorithm. There are benefits and drawbacks to every algorithm. An overview of these algorithms is provided below.

Convolutional Neural Networks (CNN):

CNNs are a sort of deep learning neural network that have been extensively employed in image categorization applications. As a result of its ability to automatically learn features from the input picture, CNNs are more resistant to changes in the input image. In order to be taught, CNNs often need a lot of training data and computer power, but once trained, they can perform image classification tasks with a high degree of accuracy. It have three main types of layers.

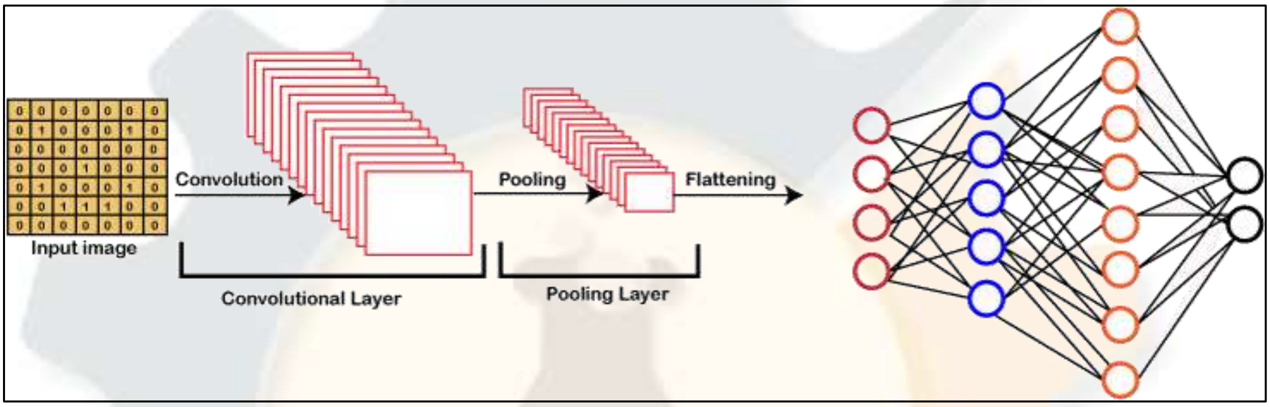

Figure 1: data flow diagram of CNN

We are using the Rectified Linear unit function which the output zero when the data is below zero, and when the input rises above a certain threshold. It has a linear relationship with the dependent variable.

Support Vector Machine (SVM):

Support Vector Machines (SVM) are machine learning algorithms that are employed in the categorization and regression of data. By locating a hyperplane that divides the input photos into distinct classes, SVMs for image classification carry out their function. When the number of features exceeds the number of samples, SVMs can still perform well since they are effective at handling high-dimensional data. To obtain best performance using SVMs, thorough hyperparameter tweaking and selection of the appropriate kernel function are required.

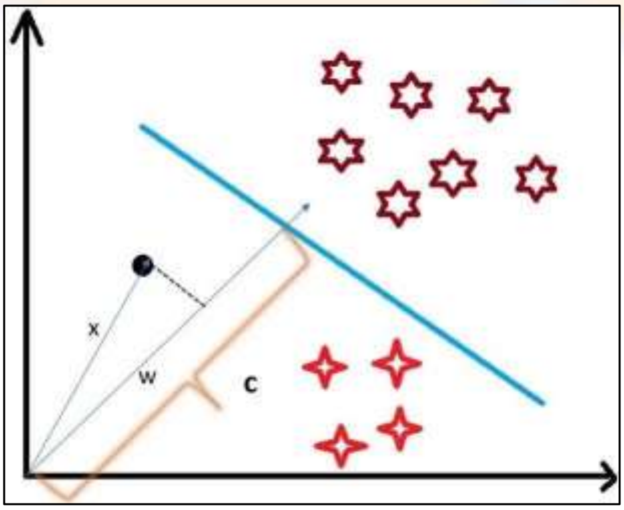

Figure 2: vector representation of datapoints

$$\vec{X}.\vec{w} = c \quad (the\ point\ lies\ on\ the\ decision\ boundary)$$

$$\vec{X}.\vec{w} > c \ (positive\ samples)$$

$$\vec{X}.\vec{w} < c \ (negative\ samples)$$

We can calculate the distance(d) in a way that no positive or negative points can cross the margin line. For correctly classified this condition must be true.

$$y_i(\vec{w}.\vec{X} + b) \geq 1$$

Viola-Jones Algorithm:

The widely used Viola-Jones method for face detection searches a picture for particular traits. A cascade of classifiers is used by the algorithm to successively identify several components of the item under detection. The algorithm is quick and capable of recognising faces in real- time applications with high accuracy.

Due to their ability to automatically learn features from the input picture, CNNs are currently the most advanced method for classifying images in a variety of applications. This makes them more resistant to changes in the input because of their ability to learn features from the input image. A common method for classifying images is the use of SVMs, particularly when there are more characteristics than samples. However, the Viola-Jones method performs better at object detection tasks, especially in real-time applications where speed is crucial. The final method of choice is determined by the qualities of the input data and the particular needs of the application.

The comparison between these algorithms in term of their accuracy is depicted in the bar graph shown in figure 1 below:

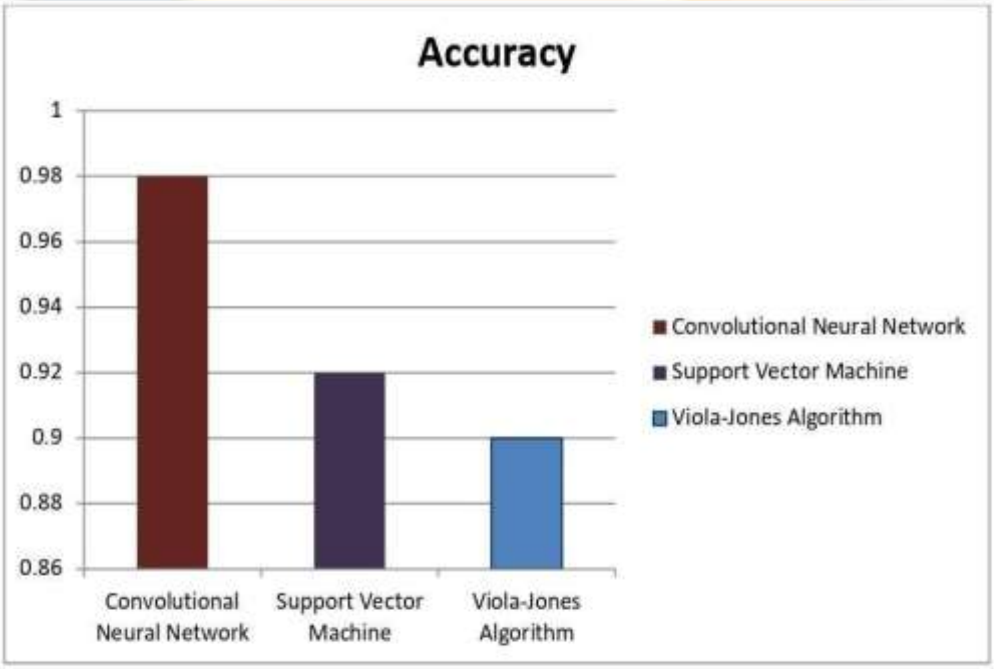

Figure 3: accuracy comparison

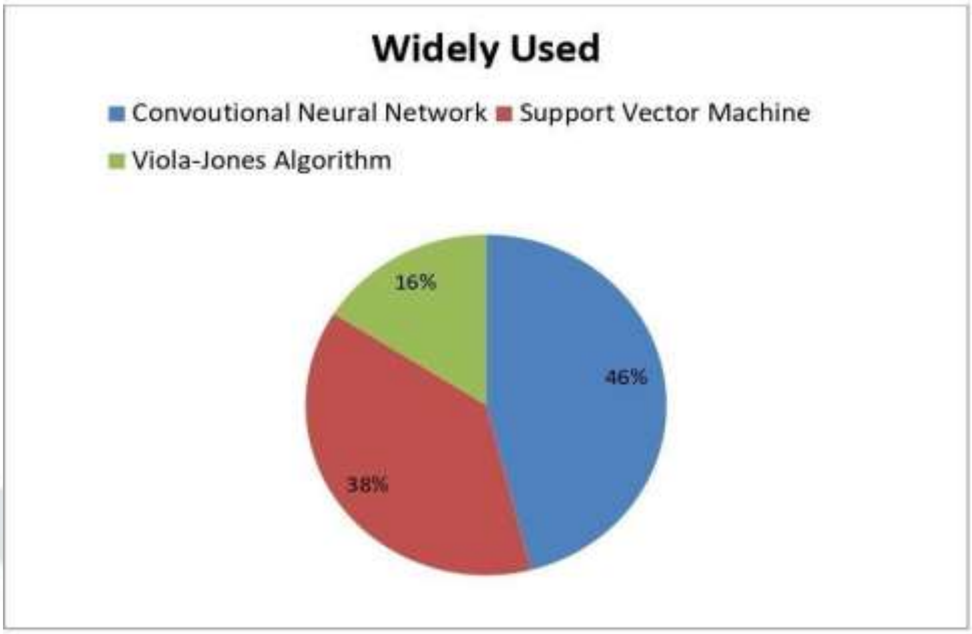

Figure 4: pie chart illustrating the most widely used algorithm and their popular application

figure 2.

C. Proposed Methodology

The components which we are using is explained above in various sections. Here we just tried to map things like how they going to work at every level of the system and with the brief depiction of what the work of each component with the help of the flowchart embedded with the abstract implementation of the prototype. The design made for the prototype is made with the best possible explanation so as to make the flowchart easy to understand at the first sight. Here we managed to get a brief functioning of our system with the best possible outcome.

Now the implication technique with the help of figure 3 is shown below:

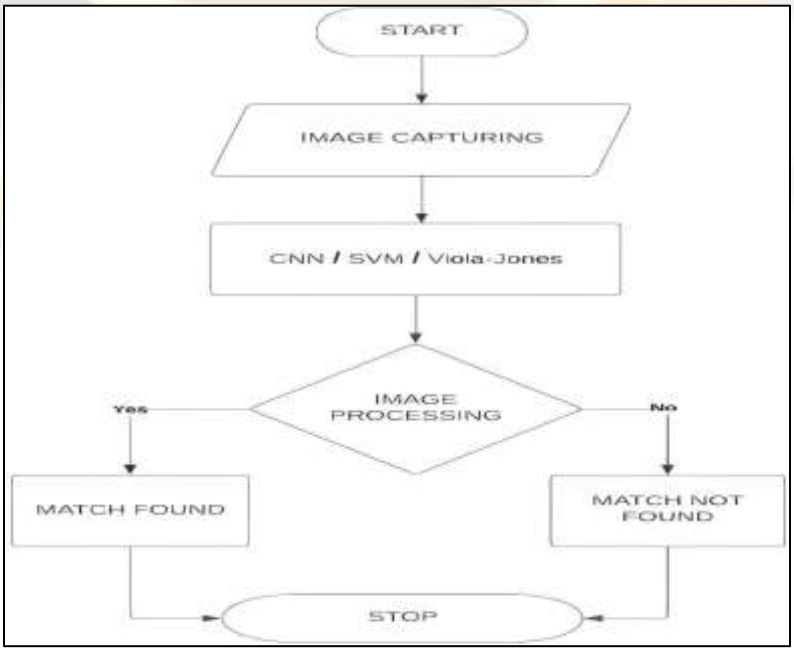

Figure 4: working of the model

The abstract implementation of the prototype CosMos is shown with the help of a diagram given below in **figure 4:**

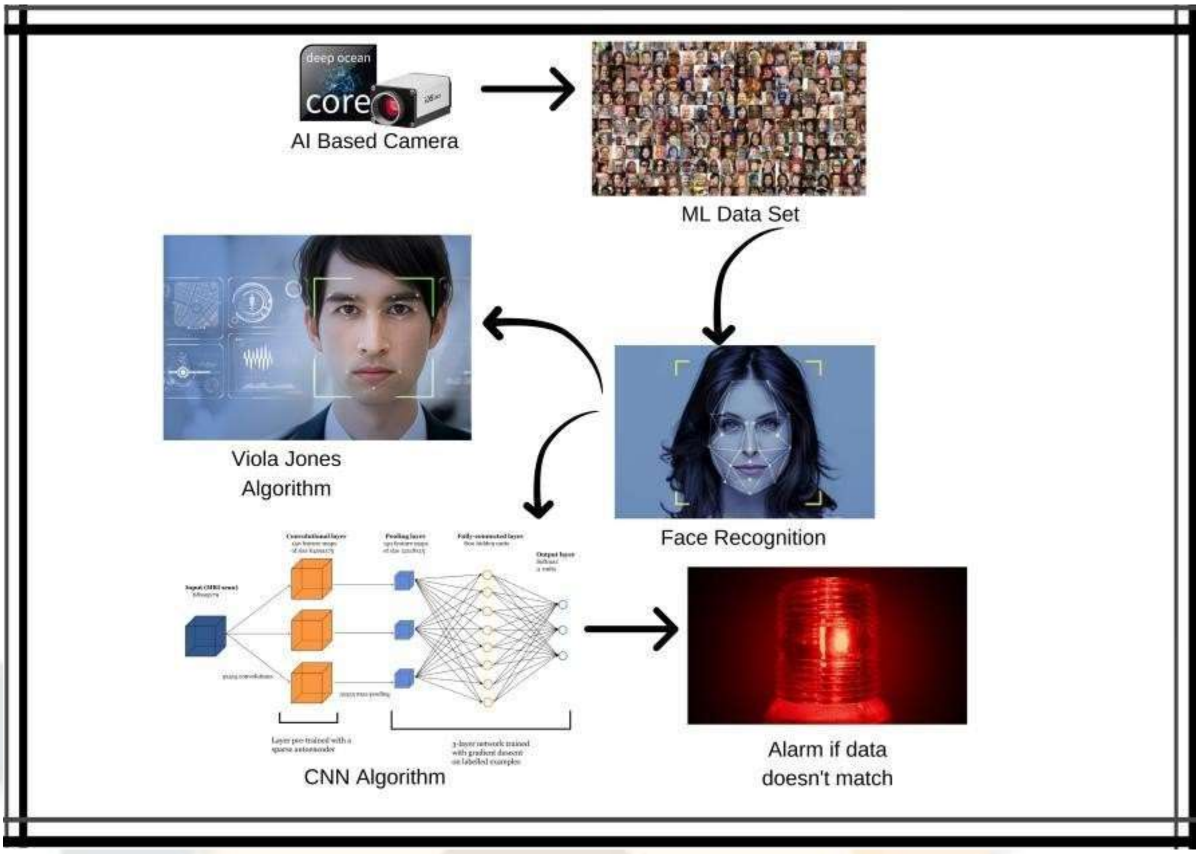

Figure 5: Abstract Implementation of the Prototype

Result and Discussion

In this section, we provide a comprehensive explanation of the design, configuration, and experimentation of our system. We evaluated the functionality of our system using a prototype implementation with IoT hardware.

A breadboard, an ESP32-CAM module, a DHT11 sensor, an ESP7266 Wi-Fi module, an Arduino UNO board, the 5 V four-channel relay modules, a PIR motion sensor, a smartphone running Android, and were all used to test the proposed CosMos. The Arduino IDE was used to programme the communication between the microcontroller board, cameras module, and mobile applications. The configuration's coding phase included the declaration of network credentials for connectivity. The Wi-Fi module a wireless connection was made. The prototype system analyses indoor temperature and humidity, detects movement, includes a video streaming module, and controls a lamp, fan, and plug. It also analyses indoor temperature and humidity. Scalability was not given a full point. The relay module was interfaced with the household appliances to guarantee correct current flow. The mobile applications get a signal from the system through the Wi-Fi modules.

In a normal CNN training procedure, the model's performance on the training and validation sets is tracked across several epochs. The term "epochs" refers to the quantity of times the full training dataset is run through the model. The accuracy trend on both the training and validation sets as the number of epochs rises is depicted in the accuracy vs. epochs graph.

The accuracy on both the training and validation sets begins low at the beginning of training and steadily rises as the model develops the ability to generate more accurate predictions. The accuracy on the training set keeps improving as the number of epochs rises, but it may plateau or even decline on the validation set. This is an example of overfitting, where the model is memorising the training data and is unable to generalise successfully to new, unknown data. The maximum accuracy on the validation set prior to a plateau or decline in accuracy is the optimum number of epochs for a CNN. The model can now accurately predict outcomes using fresh data and has learned to generalise successfully. It's important to keep in mind that the accuracy versus epoch graph might change based on the CNN's individual architecture, training method, and dataset complexity.

After performing 7 epochs on our dataset the accuracy pointed is shown below in Figure 5:

```
Epoch 1/20
215/215 [==============================] - 15s 69ms/step - loss: 1.1045 - accu
racy: 0.6530 - val_loss: 3.7369 - val_accuracy: 0.0752
Epoch 2/20
215/215 [==============================] - 9s 42ms/step - loss: 0.2220 - accur
acy: 0.9278 - val_loss: 1.5874 - val_accuracy: 0.5170
Epoch 3/20
215/215 [==============================] - 9s 44ms/step - loss: 0.1011 - accur
acy: 0.9678 - val_loss: 0.1524 - val_accuracy: 0.9452
Epoch 4/20
215/215 [==============================] - 9s 44ms/step - loss: 0.0647 - accur
acy: 0.9790 - val_loss: 0.0890 - val_accuracy: 0.9699
Epoch 5/20
215/215 [==============================] - 9s 42ms/step - loss: 0.0529 - accur
acy: 0.9831 - val_loss: 0.0279 - val_accuracy: 0.9912
Epoch 6/20
215/215 [==============================] - 9s 43ms/step - loss: 0.0329 - accur
acy: 0.9901 - val_loss: 0.1260 - val_accuracy: 0.9564
Epoch 7/20
215/215 [==============================] - 10s 47ms/step - loss: 0.0283 - accu
racy: 0.9911 - val_loss: 0.0286 - val_accuracy: 0.9911
```

Figure 5: Accuracy Stairs [11]

The training and validation accuracy graph of CosMos is shown in figure 6 given below:

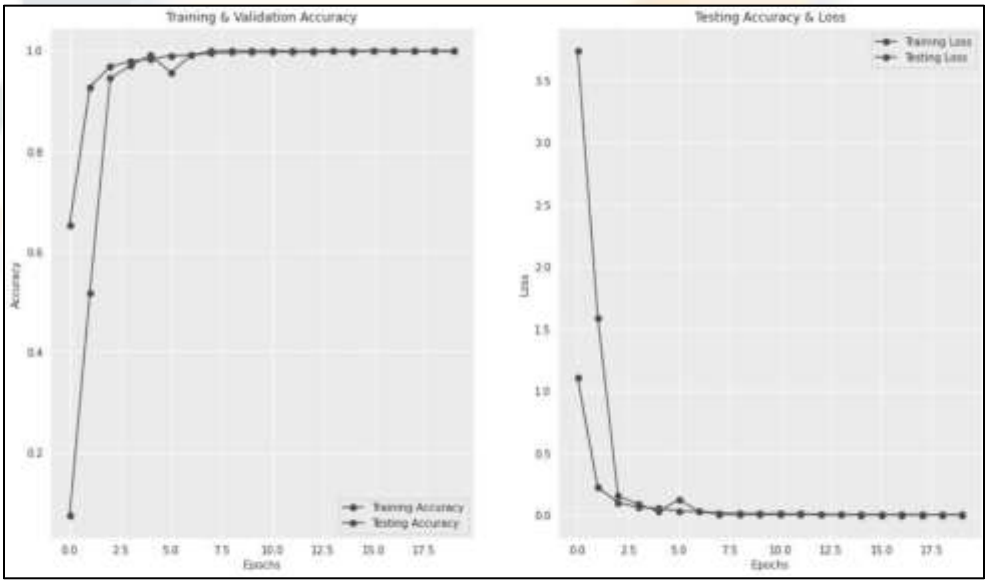

Figure 6: Accuracy Graph [11]

Conclusion

Areas of interest and development include more comfortable living, a healthier lifestyle, comfortability, and home security. Reduce daily tasks that can stress out the sick, old, and disabled and have a detrimental impact on their health. In order to do this, it was developed a smart home automation system that allows for local & remote control, monitoring, and home safety. This study contributes to the body of current creation of a versatile Android-based mobile applications for the smart home automation sector in order to conduct research in home automation. Use of CNN's deep learning model to categories & identify intruder is part of our strategy to improve home security. Identification of motion in the domestic environment serves as the basis for the detection. Using this strategy demonstrates that users will have improved home security while experiencing less disruption from notifications.

The suggested approach aims to get rid of frequent and erroneous messages in an automated smart home system. Our recommended approach’s disadvantage is that it can only identify one movement at a time. The movement of one individual at a time was the foundation for the training and classification models. There is more work to be done on the multiple motions in either people or animals are accurately predicted, classified, and feature extracted. As a result, the motion detection module can be improved to accurately classify and forecast the precise motions detected when the system detects several movements. The system's inability to take into account how emotions affect how people move is another drawback. An individual's walking pattern can change due to emotions, which could lead to erroneous categorisation. Additionally, analysis and prediction of sensor data can be used to improve user lifestyles and the functionality of home automation systems.

Future Scope

We intend to add the following phase in the research will be to add in order to get data from the home environment that can be analyzed, more sensors need be added to the system and it should be tested over a few weeks. Using our IoT devices that are linked to a cloud database, we want to collect real-time motion photographs in order to create a dataset for the suggested support vector machine method's training and classification. ESP32-CAM and a PIR motion sensor assign values to recognised objects. In the work we are doing now, video streaming is done through a web interface. The future effort will feature a mobile application module for streaming video, offering real-time, cloud-based media storage. Additionally, the mobile application will use the deep learning model for complete functionality. To prevent the system from classifying this group of people as intruder mechanism to categories friend and extended families will also be taken into consideration as future work. The full implementation of electrical appliances is also anticipated for the following stage. In our upcoming work, we'll employ deep learning algorithms to classify data and include sensors from the system incorporates smart home healthcare to aid in remote patient monitoring.

References

1. B. Celea, Quarter One Crimes Statistic, South Africa Government Journal, Pretorias, 2021.
2. K. Ashtons, That internet of things thing, RFiD Journals, vol. 21, no. 8, pp. 97–115, 2009.
3. O. Taiwoe, L. A. Gabrallas, and A. E. Ezugwus, Smart home automation: taxonomy, composition, challenge and future directions, International Conference on Computational Science and Its Applications, Cagliari, 2020.”
4. F. Mehmood, I. Ulloh, T. Ahmad, and D. Kims, Object detection mechanism based on deep learning algorithms using embedded IoT device for smart home appliance control in CoT, Journals of Ambient Intelligences and Humanized Computing, pp. 1–19, 2019.
5. D. Liciottie, M. Bernardinis, S. Romeo, and E. Frontonie, A sequential deep learning applications for recognizing human activities in smart home, Neurocomputing, vol. 398, pp. 501–517, 2020.”
6. J. Jaihari, N. Lingayate, P. S. Vijaybha, G. Ventakes, and K. P. Uple, Smart home automation using machine learning algorithm, International Conference for Emerging Technology (INCET), Belgaum, India, 2020.
7. D. Pop, F. Pope, C. Serbanescui, and A. Castiglion, Deep learning models for home automation and energy reduction in a smart homes environment platform, Neural

8. Computing and Applications, vol. 33, no. 7, pp. 1317–1339, 2019.
9. Z. Pengs, X. Lie, and F. Yans, An adaptive deep learning model for smart home autonomous systems, International Conference on Intelligent Transportations, Big Data and Smart City (ICITBS), Vientiane, Laos, 2020.
10. S. J. Shah, Z. Tariqe, and Y. Li, Audion IoT analytic for home automations safety, IEEE International Conferences on Big Data (Big Data), Seattle, WA, USA, 2018.
11. T. Khan, B. Seo, and K. Kim, Toward energy efficient home automation: a deep learning approaches, Sensors, vol. 21, no. 25, p. 7189, 2020.
12. Madhav Mathur, CNN using Keras(100% accuracy), Image Processing using Machine Learning, New Delhi, India, 2011.
13. https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm.
14. Sagar, Anil Kumar, Latha Banda, Subrata Sahana, Karan Singh, and Bhupesh Kumar Singh. "Optimizing quality of service for sensor enabled Internet of healthcare systems." Neuroscience Informatics 1, Vol no. 3 (2021): 100010.
15. Sagar, Anil Kumar, Shivangi Singh, and Avadhesh Kumar. "Energy-aware WBAN for health monitoring using critical data routing (CDR)." Wireless Personal Communications 112 (2020): 273-302.