Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 9307961, 22 pages
https://doi.org/10.1155/2022/9307961

Research Article

Enhanced Intelligent Smart Home Control and Security System Based on Deep Learning Model

Olutosin Taiwo ,[1] Absalom E. Ezugwu ,[2] Olaide N. Oyelade,[2] and Mubarak S. Almutairi [3]

[1]*School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Westville Campus, Private Bag X54001, Durban 4000, South Africa*
[2]*School of Computer Science, University of KwaZulu-Natal, King Edward Road, Pietermaritzburg Campus, Pietermaritzburg, KwaZulu-Natal 3201, South Africa*
[3]*College of Computer Science, University of Hafr Al Batin, Saudi Arabia*

Correspondence should be addressed to Absalom E. Ezugwu; ezugwua@ukzn.ac.za

Received 1 October 2021; Revised 15 November 2021; Accepted 27 December 2021; Published 15 January 2022

Academic Editor: Tarek Gaber

Security of lives and properties is highly important for enhanced quality living. Smart home automation and its application have received much progress towards convenience, comfort, safety, and home security. With the advances in technology and the Internet of Things (IoT), the home environment has witnessed an improved remote control of appliances, monitoring, and home security over the internet. Several home automation systems have been developed to monitor movements in the home and report to the user. Existing home automation systems detect motion and have surveillance for home security. However, the logical aspect of averting unnecessary or fake notifications is still a major area of challenge. Intelligent response and monitoring make smart home automation efficient. This work presents an intelligent home automation system for controlling home appliances, monitoring environmental factors, and detecting movement in the home and its surroundings. A deep learning model is proposed for motion recognition and classification based on the detected movement patterns. Using a deep learning model, an algorithm is developed to enhance the smart home automation system for intruder detection and forestall the occurrence of false alarms. A human detected by the surveillance camera is classified as an intruder or home occupant based on his walking pattern. The proposed method's prototype was implemented using an ESP32 camera for surveillance, a PIR motion sensor, an ESP8266 development board, a 5 V four-channel relay module, and a DHT11 temperature and humidity sensor. The environmental conditions measured were evaluated using a mathematical model for the response time to effectively show the accuracy of the DHT sensor for weather monitoring and future prediction. An experimental analysis of human motion patterns was performed using the CNN model to evaluate the classification for the detection of humans. The CNN classification model gave an accuracy of 99.8%.

1. Introduction

Home-based crime attacks, theft, and burglary are on the increase annually. Despite the lockdown and stay-at-home order, several homes were still attacked and burgled in South Africa [1]. According to the report in [1], attacks on homes and properties have aggravated into sexual offenses and assaults. This has shown a need for additional and improved levels of security for homes, properties, and individuals. The growth in the industrial development of smart home automation systems and the rate of research works carried out in the field give assurance of using the field to solve security issues arising in smart home environments. A secured home will make occupants live in peace and without fear, irrespective of their location. Having remote control over the home, a clear view of the situation within the home, and a system that notifies the home owner of impending danger without any false alarm is of high importance in this era. Advancements in information technology (IT) and the internet of things (IoT) have provided platforms suitable for the control, monitoring, safety, and security of homes and properties. The internet of things is the interconnection of several physical devices, network connectivity, and communication media to transmit information between devices and devices to humans [2, 3]. The IoT technologies have led to communication, interaction, and data exchange among devices, sensors, and appliances.

Moreover, the overall remote control of the home over the internet, Wi-Fi, Bluetooth, or other network modules via smartphones, laptops, tablets, and other related devices is now made possible, thus giving the desired comfortability to people in the applied area of IoT. Activities in the home and environmental conditions can be monitored and controlled through IoT technologies and are often referred to as smart home automation. According to [4], a smart home (SH) is a place of abode incorporated with a communication network that connects the major home appliances and services utilized for remote control, monitoring, and access from within or outside the premises. A smart home automation system is projected to deliver safety and home security services aside from regular home control. The smart home environment is embedded with intelligence to reduce human efforts in controlling and monitoring the house and its appliances. With ambient intelligent home control and security mechanism, the occupants can have a conducive environment to live in, excluding incessant notifications and alarms.

Artificial intelligence (AI) is contributing immensely to independent automated systems in the field of IoT. Home security is an essential aspect of smart home automation for convenient living. With the aid of AI models and specific IoT technologies, home premises and their conditions can be remotely monitored, controlled, and surveyed. With machine learning and deep learning models, systems can automatically make intelligent decisions on behalf of humans for enhanced quality living. Classification and detection of objects are the key strengths of machine learning and deep learning models. The application of AI in home automation will assist in the classification and detection of intruders in a smart home environment. Studies have shown that deep learning and machine learning models have been efficiently applied in smart home automation for object detection and recognition, human activity detection, facial recognition, intelligent control of appliances, energy efficiency, home monitoring, safety, and security [5–10]. A deep learning algorithm is a machine learning model that adopts the structure of a human brain to conclude data analysis with a given logical structure [11]. Although several AI models exist for the classification, detection, and prediction of objects, the deep learning model has been widely used and proved efficient for providing solutions to engineering, classification, and detection challenges [12].

A convolutional neural network (CNN) model is a subclass of artificial neural networks for image processing and recognition. The CNN model is widely used to solve image-based problems and is efficient with several convolution layers [13]. The security mechanism that can be adopted in a home environment is based on motion recognition and surveillance. With the CNN model, images captured by the surveillance camera can be processed based on the area of interest for detection. A deep learning model-based intelligent detection can enhance a smart home automation system to classify detected motions into home occupants or intruders before sending an alarm to the user. Such an efficient home automation system design reduces stress, wastage of basic amenities such as electricity and water, and enhances the life quality. The smart home automation system controls lights, entertainment systems, environmental conditions, and other home appliances through ambient intelligence [14]. However, there are still some challenges in the smart home automation domain, such as wide-range connection for the control of the home, intelligent decision making by the system, storage, precise motion detection, and real-time data storage for future prediction, analysis, and decision making. To this end, we propose a real-time cloud-based, low-cost smart home automation system based on an Android mobile application. Note that by low cost, we mean to say that the prototype implementation of the proposed system using IoT hardware such as microcontroller board, sensors, cameras, and other components was set up using low-cost IoT hardware, which by implication is affordable and easy to configure.

To enhance security in a smart home automation environment and make meaningful research contributions, we propose a deep learning model (CNN) to classify human movement patterns as a security mechanism for the identification and classification of humans. The proposed classification of the movement patterns is to distinguish between regular home occupants and intruders in a smart home environment. The patterns of regular home occupants are prerecorded in the system, and once a motion pattern is detected, the system compares it with the existing motion patterns on the system and classifies it as either a regular home occupant or an intruder before raising the alarm. The approach aims to overcome the challenges such as false detection due to the use of masks and distortion due to weather or lighting associated with the biometrics approach in the existing literature. The home is controlled via an Android smartphone, and all the conditions of the home can be viewed on the GUI of the smart home automation application. The mobile application uses a platform as a service for its real-time storage of data generated by sensors and displays a graphical output of the ecological readings. In addition, the IoT hardware choice for the prototype implementation is based on low-cost, extensible, and accessible devices. In summary, the technical contribution of this paper is highlighted as follows:

(i) A proposal for a deep learning (CNN) algorithm for intrusion detection in a secured smart home automation environment

(ii) A design and development of an Android-based smart home automation system for the control and monitoring of electrical home appliances and environmental conditions

(iii) A prototype implementation of an IoT, smart home system for home control, monitoring, and security

(iv) Experimentation of a CNN-based deep learning model for classifying human walking patterns to detect intruders in a smart home environment

The remainder of this paper is structured as follows: a review of related works is presented in Section 2; Section 3 gives the proposed system architecture and functionalities and discusses the CNN-based home security concept. In Section 4, we present the experimentation and prototype implementation. Finally, the conclusion and future work direction are given in Section 5.

2. Related Literature

Research in the smart home automation domain has witnessed impressive progress, innovations, experimentation, and implementation. With the advancement in IT and IoT technologies, smart home automation systems now offer services beyond the initial home control and environmental monitoring. Machine learning and deep learning algorithms have been deployed to improve the intelligence responses in smart home automation systems. Some research lapses observed are the use of Bluetooth, GSM, SMS, Zigbee, and other communication modes for control which has the drawback of limited range coverage. Other research problems noted are the use of web-based design and SMS as means of appliance control. Lastly, few existing research works incorporated intelligent decisions into the systems. This section presents a review of related and relevant literature works in smart home automation and artificial intelligence. In contrast, Table 1 presents a summary of related works in smart home automation, focusing on the functionality, use of cloud computing, and other features compared to our work.

Xiaodong and Jie [15] presented the design of embedded smart home control and monitoring system using an STM32 microprocessor. Their system was designed for monitoring indoor environmental conditions (temperature and humidity) and home control. The authors used a GSM and GPRS module to communicate remote control and Zigbee terminal communication between the home devices. The implementation of their system was achieved through coordinated remote control and feedback on household appliances and the μCOS-II as an embedded real-time operating system. Bimenyimana et al. [16] designed and implemented a web-based home control system to control and monitor electrical appliances. Their system allows the user to manage electricity usage in the home by remotely turning off an appliance when not in use. A prototype implementation was displayed using Arduino board, Node MCU, Light-Emitting Diodes (LED), relays, and other hardware. A motor DC used in the design controls the light and reduces the energy consumption. However, the control is web-based, taking longer control time than a mobile-based application.

Gunawan et al. [17] presented a smart home control system for controlling home appliances via a website. The system uses temperature, smoke, and gas sensors to monitor environmental conditions such as temperature and humidity, detect gas leakage, and ensure security using a PIR sensor for intrusion detection at the home gate. Communication between home appliances, sensors, and the Arduino board was established through an APC220 Wireless RF module. At the same time, an Ethernet Shield connects the board to the website interface to control the home. The wireless modules give a wider communication range, but a website is not as efficient for the control of the home as a mobile application in terms of time consumption. Similarly, a Wi-Fi-based smart home automation system was proposed by Singh and Ansari [18]. The system controls home appliances, monitors environmental parameters (humidity and temperature), and maintains home security through an alarm buzzer. A prototype implementation of the proposed method was presented using an Arduino microcontroller, ESP8266 Wi-Fi technology alongside sensors, and other IoT hardware. An application named Blynk provided the platform for control. In their system, home control access is granted by a designated administrator of the system.

Aadel and Ali [19] presented a design and prototype implementation of a low-cost smart home system. The system was designed to control windows, doors, lights, electricity, and temperature in the home. A prototype implementation of the system was displayed using an Arduino board, servo motors, LED lights, temperature, and motion sensors. The door and windows were controlled using servo motors, a PIR motion sensor for tracking movement in the house, the INA219 high-side DC sensor to monitor the drop and supply of voltage, and a DHT11 temperature and humidity sensor was used for measurement of the temperature and humidity in the home. The system used Bluetooth to communicate, which is a limitation in terms of communication range.

An optimal and automatic control system for the control of electrical home appliances was proposed by Parsa et al. [20]. The proposed system's design is to control power consumption by automatically turning on and off the smart plugs connected to specific devices in the home at the appropriate time. An optimization approach was used to determine the appropriate time of use before the automatic switch was implemented according to the predefined constraints. The designed system favors the supplier more than the electricity consumer in the home.

Classification of human activities in smart homes using different deep learning models was presented by Liciotti et al. [6]. The deep learning models were applied sequentially to determine a specific action carried out by the home occupant at a particular time. The long short-term memory (LSTM) was applied for modeling temporal sequences in long-term dependency situations. The LSTM is combined with other machine learning algorithms to learn, recognize, and predict human actions such as bathing, walking, eating, relaxing, and sleeping and in a smart home environment. Similarly, Manu et al. [21] proposed an LSTM deep learning algorithm for performing predetermined tasks based on human activity recognition. The LSTM algorithm is integrated with IoT technologies to predict human activity towards home control, safety, and comfort for users.

The support vector machine (SVM) algorithm was implemented by Majeed et al. [22], for intelligent decision

Table 1: Summary of related works in smart home automation.

	Wireless technology	Functionality	Scalability	Cost-effective	Cloud computing platform	Notification on the lighting condition of the home	Intelligent decision making
Taiwo et al. [29]	Bluetooth and Zigbee	Control of home appliances with Android-based smartphone	No	Yes	No	No	No
Taiwo and Ezugwu [30]	Not discussed	Smart home automation	No	Yes	No	No	No
Majeed et al. [22]	Not discussed	Control light and home temperature with the use of a sensor via an Android-based smartphone	Yes	Yes	Yes	No	Yes
Alam et al. [31]	No	Control of lights and sensors in the home through a computer interface	No	Yes	No	No	No
Naing and Hlaing [32]	No	Control and monitoring of light through sensors	No	Yes	No	Yes	No
Jena et al. [33]	No	Control of lights through voice recognition	No	Yes	No	No	No
Taiwo and Ezugwu [34]		Controls home appliances and detects motion through an Android-based app	Yes	Yes	Yes	Yes	Yes
Mahmud et al. [35]	Wi-Fi	Control of home appliances and electronic machines through a web interface	No	Yes	No	No	No
Liao et al. [36]	Wi-Fi	Control of home appliances, monitoring of environmental conditions in the home, security of the home, and measurement of gas level in the home	Yes	Yes	Yes	No	No
MufHAS	Wi-Fi	Controls home appliances, monitors environmental conditions, and detects movements	Yes	Yes	Yes	Yes	Yes

making in a smart home. The intelligent control and status of home appliances were based on the SVM algorithm. Secured communication is also ensured using blockchain technology. An Android application was developed for the remote control of home appliances. Brenon et al. [23] presented a voice-controlled, context-aware-based home automation system enhanced with a deep reinforcement learning model. The reinforcement learning model is used for context extraction from graphical representation. Their approach also used a CNN for decision making in the home environment. Jaihar et al. [7] presented an approach that combines three AI models for security, intelligent control, and decision making in a smart home automation system. The authors used a CNN model, fisher face classifier, and SVM approach for facial recognition and detection in a smart home environment. Also, Neverova et al. [24] presented a DL approach for detecting human identification from motion patterns. The approach presented compares several DL models to learn human motion patterns efficiently for multimodal authentication systems. Feature and latency extraction was used for the learning representations. The comparison of the CNN-based models presented shows that motion patterns are effective for the authentication and identification of humans.

The existing literature has presented home security and safety mechanisms, models, frameworks, and experimentation. For instance, Saravanan et al. [25] proposed a Bluetooth-based safety and security system for home automation. A smartphone application was designed to control home devices, doors, and overall monitoring. Their system automatically switches off the lights at night, detects gas leakage or smoke, and controls home appliances. The system also locks and unlocks the door in the home automatically via an authentication module. Communication with the user's smartphone and the hardware is via Bluetooth. The method used a secured mechanism. However, the coverage range of Bluetooth is lower compared to Wi-Fi-based technologies. Jabbar et al. [26] proposed a Wi-Fi-based intelligent home automation system for home control, monitoring, and security. The system measures ecological factors around the home and displays the value, controls electrical appliances, and ensures home security through motion detection. An alarm is raised by the system when a motion is detected to alert the user of an intruder. Also, the system can support a reduction in electricity bills through remote control of appliances. However, there is no intelligent module to distinguish between the detected motions.

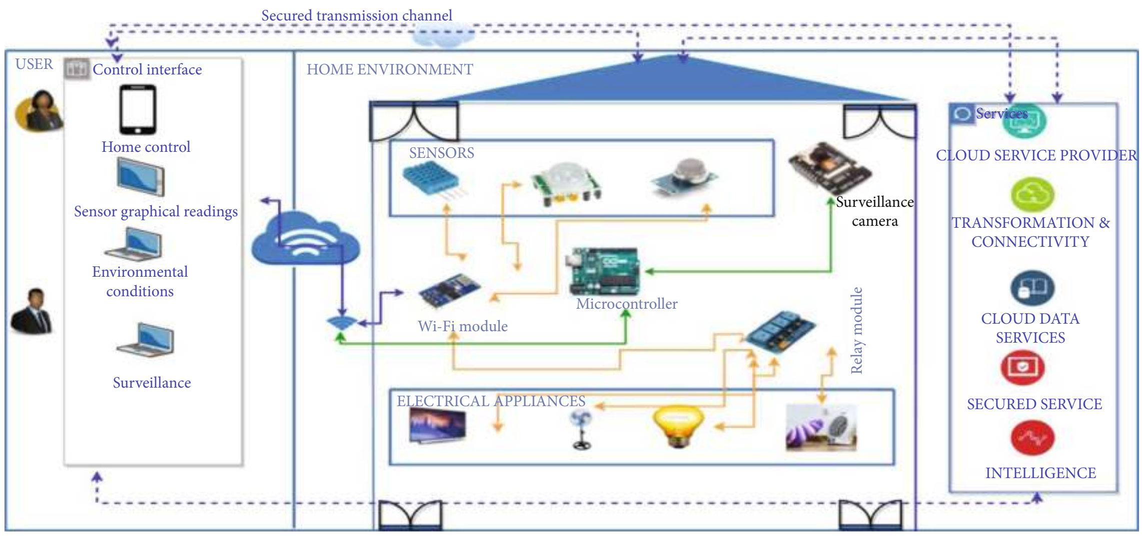


Figure 1: The MufHAS Architecture.

A security system for a smart home automation system was presented by Ajao et al. [27]. The system secures entry points of the home and grants access through authorization and authentication. An in-app message notifies the user of access or denial of access. Sensors are also embedded in the system for motion detection. The home security system is controlled through wireless IoT communication and Android mobile application. The presented approach is easy and flexible to use, but capturing an image of intruders is not included for surveillance. Singh et al. [28] proposed a smart home automation system for the safety and control of electrical appliances, doors, and movement detection in a house. The system could also send an alert to the user in response to sensors detecting low levels of gas in a cylinder or the presence of a human. A prototype implementation of the system was carried out using an Arduino Uno board, a Node MCU ESP8266, IR, and LDR sensor modules. A summary of the studies reviewed in this section is listed in Table 1. A comparative presentation of features, methods, and functionalities was included in the summary.

3. System Architecture

The reviewed literature has clearly shown that artificial intelligence greatly contributes to the enhancement of intelligent response, detection, and decision making in smart home automation systems. However, there are still associated with exact motion detection, aversion of false and frequent notifications by home automation systems, especially in the security domain of the field. Therefore, we have been motivated to design and develop an intelligent smart home automation system with various functions and a multifunctional smart home automation system (MufHAS). Also, it is expected that an efficient home automation system must be intelligent to make decisions on behalf of the user to reduce disturbance and human interference. Therefore, a CNN deep learning model is proposed to enhance our system's home security by detecting abnormal intruder movement. In this section, we describe the architectural design of our MufHAS to control home appliances, monitor and measure environmental conditions, and detect movement in the home. An effective smart home automation system is known to offer assistance to the disabled, sick, or elderly [30], control home appliances remotely [37, 38], manage energy [39, 40], and measure and monitor environmental conditions [41, 42]. It also provides home security to its user by detecting unauthorized movement or intrusion [43, 44]. One of the major objectives was to use the emerging IoT technologies to provide a suitable medium to achieve most of the vital roles a smart home automation system must offer its users. Smart home automation systems' major functional components are smart home technologies, sensors, appliances, microcontrollers, and control mechanisms. An IoT smart home automation and its components are networked together either wired or wirelessly for automated control via a specified medium. With the help of specific IoT sensors, convenience, the safety of lives and properties can now be ensured in a smart home automation environment. For example, motion sensors can alert the user via the system about movements in the home. The user can conveniently switch off lights in the house from any location with the aid of an intelligent notification from the system, thus conserving energy. In recent times, an effective smart home automation system has improved beyond the conventional control of heating, ventilation, and air conditioning appliances in the home to overall automation with the incorporation of home safety and security mechanisms. The overall architectural design of the home is depicted in Figure 1.

3.1. MufHAS Modules. The system has three major modules, the user's side, the home environment, and the backend that comprises the database module for storing data generated from devices in the home and the intelligent module for ensuring security in the home. The user communicates wirelessly with the home environment over the internet using Wi-Fi. The home environment comprises sensors, detectors,

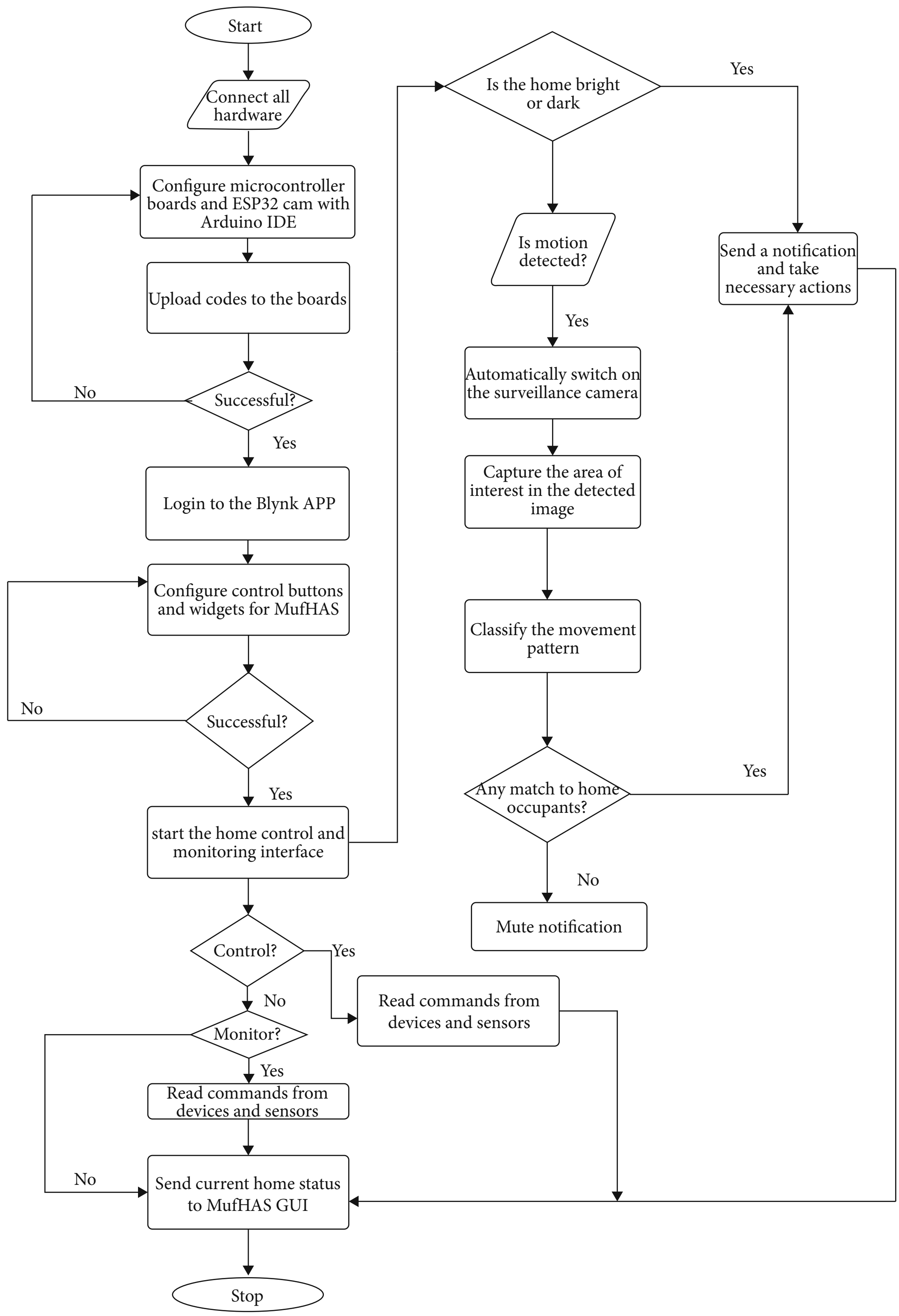


Figure 2: Flowchart of MufHAS.

home appliances, a surveillance camera, and a communication gateway. The architectural design is further explained with the prototyping in Section 4. The microcontroller board is responsible for delivering services issued through a command from the user's phone to the home environment. A real-time database hosted in the cloud is used to store data generated by devices and sensors in the home. The services rendered by cloud computing platforms complement the smart home automation system. Thus, data generated from an intelligent home environment should be stored to know the performance of the installed devices, especially sensors. Also, prediction and analysis can be carried out based on the data generated in a smart home to enhance systems to be developed and to know the changes in the home's environmental conditions. The process workflow of MufHAS is presented in Figure 2.

3.1.1. User's Module. On the user's side of the system, control and monitoring of the home take place via the GUI of an Android-based smartphone. The interaction between the user and the home takes place at this end. With the designed Android application, users can easily have a real-time view of environmental conditions (temperature and humidity), change the status of an appliance (on or off), live stream the home's surveillance, and perform other tasks as desired. With the cloud computing platform integrated into the mobile application, users can view a graphical chart of the sensors in the home environment. The data generated by the sensors are also stored in the cloud for future use. Communication between the user and the home environment is over the internet. Control of the home is done over the mobile application. However, the surveillance can be streamed from laptops, tablets, or desktop computers.

3.1.2. The Home Environment. The home environment comprises smart home appliances, sensors, a Wi-Fi module, and a surveillance camera. Bulbs, fans, television, and sockets are the primary home appliances considered in our design. The IoT devices incorporated into the design are the temperature and humidity sensor, motion sensor, and camera. The home appliances, devices, and sensors are connected to a microcontroller for communication with each other and the outside environment over a wireless network. The ESP8266 functions as the microcontroller and Wi-Fi module. The communication protocols used are Wi-Fi, one of the primary operating standards for home automation technology, TCP/IP, and HTTPS/IP. The ESP32 camera module is interfaced with an Arduino board as it requires its own microcontroller for proper functionality and power supply. The ESP32 camera board has an in-built Wi-Fi chip for wireless connectivity. The ESP32 and ESP8266 are connected to the same network for seamless communication.

3.1.3. The Backend Module. The backend comprises the intelligent module that oversees the security of the home and the cloud computing platform for storage. The home's security is enhanced with a deep learning model (CNN) for the detection, classification, and notification to the user about the presence of a human or other objects in the home environment. The deep learning model classifies the detected object as a regular home occupant or an intruder based on the movement pattern. The motive behind the classification is to avert frequent notifications and false alarms. Based on the intelligent classification of the detected object, a notification is sent to the user for necessary action. Steps involved in the connection, communication, operations, and services of the MufHAS are presented in Algorithm 1.

3.2. MufHAS' Functionality. This section explains the components of our home automation system. As previously stated, our system is made up of home appliances, sensors, and an Android smartphone to control and monitor the home. The user communicates and monitors the house over the internet via an Android smartphone. The major functions of our designed smart home automation system are control and monitoring of essential home appliances, measurement of environmental conditions, and motion detection. The proposed system is based on wireless network connectivity, and all its functionalities are further explained in the next subsections.

3.2.1. Control and Monitoring of Home Appliances. The designed MufHAS allows essential home appliances to be controlled within and without the home premises. The user controls the home over the internet outside the home and remotely while within the home. This design consideration makes the system function in a dual mode (remote and global control). Home appliances factored into our system's design are bulbs, air conditioners, sockets, fans, heater, television, fridges, and switches. Control of the home is through an Android mobile application. The user registers to be on the platform to generate login credentials for subsequent home communication. Control over the internet while away from home will reduce the consumption level of amenities and thus impact the economy. For instance, if the user forgets to switch off the light before leaving home, the light can be switched off from any location with global control. The remote control is factored into our consideration to assist the user in reducing the amount spent on internet data as our system is designed to be low cost.

3.2.2. Control and Measurement of Home Conditions. Sensors are installed in the home to measure environmental conditions within the home premises. A DHT11 is a sensor that covers a wider range of temperature and humidity. It is also accurate and more precise in reading. In our design consideration, if the home's temperature is higher than the desired level, the fan is automatically switched on, and if the temperature is lower than the desired level, the heater is switched on. For enhanced healthy living, smoke and fire detectors, carbon monoxide detectors, gas sensors, and air quality sensors are considered. Carbon monoxide is odorless and colorless; thus, it is termed a silent killer. The carbon monoxide detector measures the level of carbon monoxide in the home environment. If a higher concentration is detected, a warning is sent to the user via the mobile application to avert inhalation of poison that can harm the body system. The designed system not only caters to convenience but also for enhanced healthy living.

3.2.3. Security and Detection of Movement. The security of the home against intrusion is of high importance. With the installation of door and window sensors, users can have the home secured against break-ins. MufHAS has sensors installed to detect movement in the home environment. An IoT camera serves as a surveillance camera for capturing intruders before notifying the user to improve the home's security. The user is notified about leaving home, and the doors or windows are unlocked. The system performs an

```
1:  Begin
2:  Define N_c parameters
3:  Initialize EHA and HSD
4:  Establish and confirm the status of N_c
5:  IfN_c = 1
6:    Evaluate the initial state of Ha; ∀EHa ∈ N_C
7:    ifHa = n (where n = number of configured home appliances)
8:      Start MufHAS
9:    Else, go to step 4
10:  End if
11:  if not (N_c&&MufHAS = 1)
     go step 4
12:  Evaluate the initial state of Hs; ∀Hs ∈ N_C
13:  IfHs = n (where n = number of home sensors and detectors)
14:    Connect MufHAS to the internet
15:    Acquire sensor data
16:  Else, go to step 4
17:  If is_connected(MufHAS)
19:    Get the values for T, H, and motion
20:    Upload data to CS via MufHAS
21:    Update status of Hs in MufHAS
22:    Display graphical status of Hs in MufHAS
23:    Synchronize data to CS
24:  Else, go to step 12
     End if
25:  Case 1: (LDR)
26:  if(D=1),then
27:    Notify the user, "It's DARK, Turn on the LIGHTS."
28:  Else
29:    Notify the user "It's BRIGHT, Turn off the LIGHTS."
30:  break;
31:  Case 2: (Home security)
32:    Ensure the camera is ON
33:  IfM is detected,
       Notify via iHOCS and apply SVM
34:  IfM ∈ (HWp_1, HWp_2, HWp_3, ,⋯HWp_n)
35:    Mute alarm
36:  Else,
       Notify user via email "TOSIN: Motion detected"
37:    Raise alarm and send picture to email
38:  end if
38  User monitors Ha and Hs via MufHAS app
40:  Remotely control the home
41:  End
```

Algorithm 1: MufHAS (home control, monitoring, and security) algorithm

intelligent check with the aid of a CNN model by cross-checking the detected movement with the predefined movements in the database. If the movements are that of home occupants, then the system does not raise the alarm. Else, an alarm is sent to the user with the captured image. The PIR sensor has trigger sensitivity with an adjustable duration of the trigger signal, allowing the user to set the desired trigger sensitivity. Motion recognition and pattern could help identify suspicious and malicious activities [45], thus preventing intrusion.

3.3. Deep Learning-Based Home Security. Home security and safety are of great importance to the well-being of home occupants. Therefore, a smart home automation system should have a management scheme for home security via the control module with alarms or alerts [46]. With motion recognition and detection, the smart home environment can be secured against intruders while minimizing false alarm rates from the system. We propose a home security module for movement recognition, classification, and detection. Human movement patterns can uniquely identify a person as walking patterns differ from person to person [47, 48]. IoT sensors are widely used for the collection of data from the environment. Motion detectors, sensors, and cameras are applicable in gathering information about a human's motion pattern and activity. Thus, biometric verification of humans is possible through their movement patterns [49]. A smart home environment can therefore be secured using motion sensors and surveillance cameras. Our work proposes a home control and security system based on motion recognition. We used this concept because it is faster and discrete for identification as against other biometric means of identification [49]. Also, other human detection mechanisms are based on images that compare skin tones, skin color, eye color, and other facial attributes. Although facial detections prove effective, several factors can influence the accuracy of facial detection in a home security domain. Such factors are lighting, weather, brightness, use of facial masks, hoods, and so on [50]. With motion sensors, movement in an environment can be silently detected, captured, and verified. The proposed security module of our system performs operations by capturing movements in the home and comparing them with the predefined motion patterns (home occupants) before raising the alarm.

A CNN is a network of convolutional and pooling layers to extract main features from an input towards the desired output. CNNs are applicable in domains such as facial recognition [51], image classification, image and video recognition, motion recognition [52] recommender systems, medical image analysis [53], and other classification and decision-making domains. The mathematical definition [54] for the image classification of the CNN model represented as a tensor is given as follows:

$$\dim\left(\text{image}\right) = n_H n_W n_C, \tag{1}$$

where n_H is the size of the image height, n_W is the size of the image width, and n_C is the number of channels.

The filter is calculated as

$$\dim\left(\text{filter}\right) = \left(\mathfrak{f}, \mathfrak{f}, n_C\right). \tag{2}$$

$\mathfrak{f}$ denotes an odd dimension.

From equations (1) and (2), the filter of an image is

$$\text{conv}(\mathcal{I}, \mathcal{K})_{x,y} = \sum_{i=1}^{n_H} \sum_{j=1}^{n_W} \sum_{k=1}^{n_C} \mathcal{K}_{i,j,k} I_{x+i-1, y+j-1, k}. \tag{3}$$

The CNN is combined with a Softmax classifier for the classification of the extracted feature. A Softmax classifier is a linear classifier used in deep learning for the classification of vectors and to determine the probability of the extracted [55]. The Softmax is also known as multinomial regression and can be used for mutually exclusive multiclass classification. It has been widely used in deep learning, yielding excellent performance [56, 57]. The mathematical definition of Softmax is given as

$$\sigma\left(\vec{\mathcal{Z}}\right) = \frac{e^{\mathcal{Z}_i}}{\sum_{j=1}^{\mathcal{K}} e^{\mathcal{Z}_j}}, \tag{4}$$

where (Z) is the input vector to the Softmax function made up of $(z0 \cdots zk)$, Z_i values are the elements of the input vector to the Softmax function that can take any real value, e^{Z_i} is the standard exponential function applied to each element of the input vector, and $\mathcal{K}$ is the number of classes in the multiclass classifier

1: Begin
2: Define N_c parameters
3: Initialize *Ms* and *Cam*
4: Establish and confirm the status of N_c
5: ***If*** $N_c = 1$
6: Evaluate the initial state of the home
7: ***If*** the camera and motion sensor are active
8: Start MufHAS security
9: ***Else***, go to step 4
10: ***end if***
11: ***while****Ms* is ON
12: ***for*** each object detected do
13: ***if*** cam is on sleep mode;
14: trigger ON
15: ***else***
16: Continue streaming
17: ***end if***
18: Capture image
19: *Extract region of interest*
20: Apply the trained CNN model
21: ***if***$O_d \in (HWp_1, HWp_2, HWp_3, , \cdots HWp_n)$
22: Save to cloud
23: ***else,***
24: *Raise an alarm and save the captured picture*
25: **end if**
26: **End**

Algorithm 2: CNN-based home security algorithm

Algorithm 2 presents the steps involved in our movement-based object detection and classification using the CNN deep learning model. The workflow of the home security and image processing based on the CNN model is presented in Figure 3. The CNN model is used majorly to classify human and pet movements based on the motion pattern. The object in our context refers to human beings and pets as both occupy a house. Thus, either of these two can trigger an alert while moving. The motion patterns considered are walking, jumping, limping, and running. In our work, the surveillance camera captures and records the activities in the environment. However, the ESP32 camera used in the prototyping does enable a sleep mode if there are no records of activities. Hence, the combination with a PIR motion sensor. If a movement is noticed, the PIR sensor sends a signal to the camera, and it comes on. The camera, in turn, captures the region of interest (movement pattern) in the detected image, classifies and compares it with the existing and predefined set of movement patterns in the database. If the detected pattern matches the existing ones, an alarm is not triggered. Otherwise, the user is notified via the mobile application of the house's situation.

The CNN architecture applied for detecting intruders in the smart home framework is presented in Figure 4. Four (4) convolutional blocks are composed to build the CNN architecture, interleaved with zero padding and max pooling operations. This composition allows for a comprehensive and fine-grained feature extraction procedure, leading to

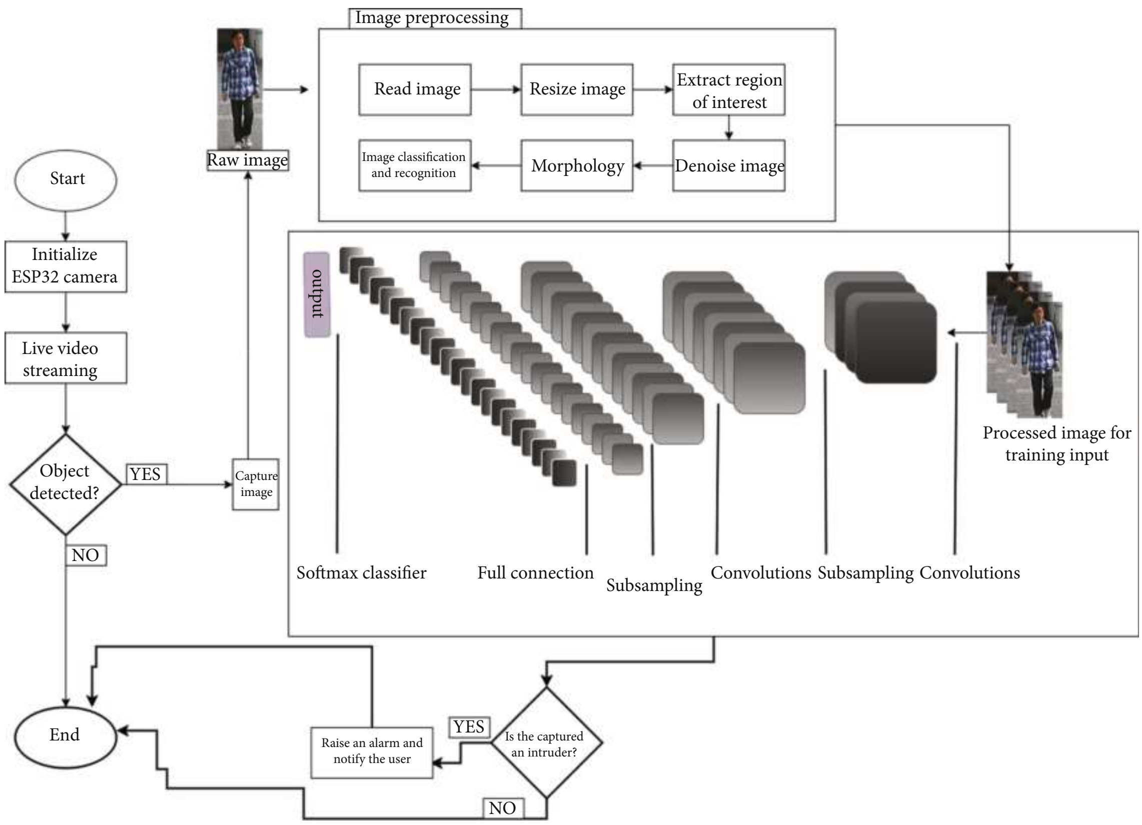


Figure 3: Flow process for CNN-based home security.

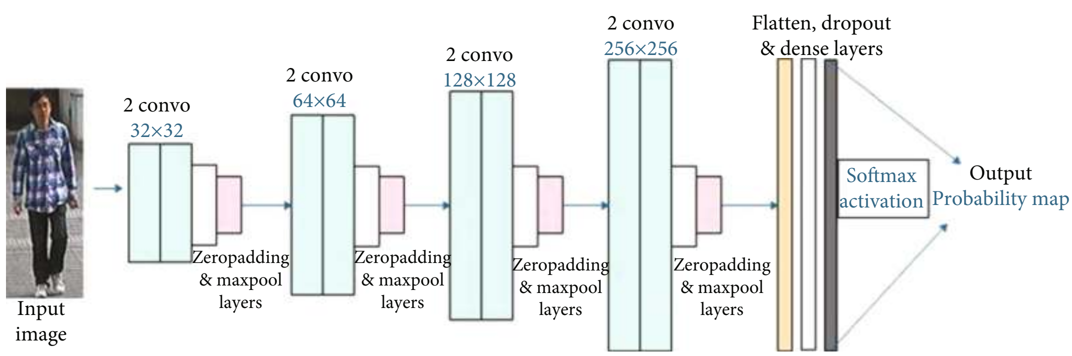
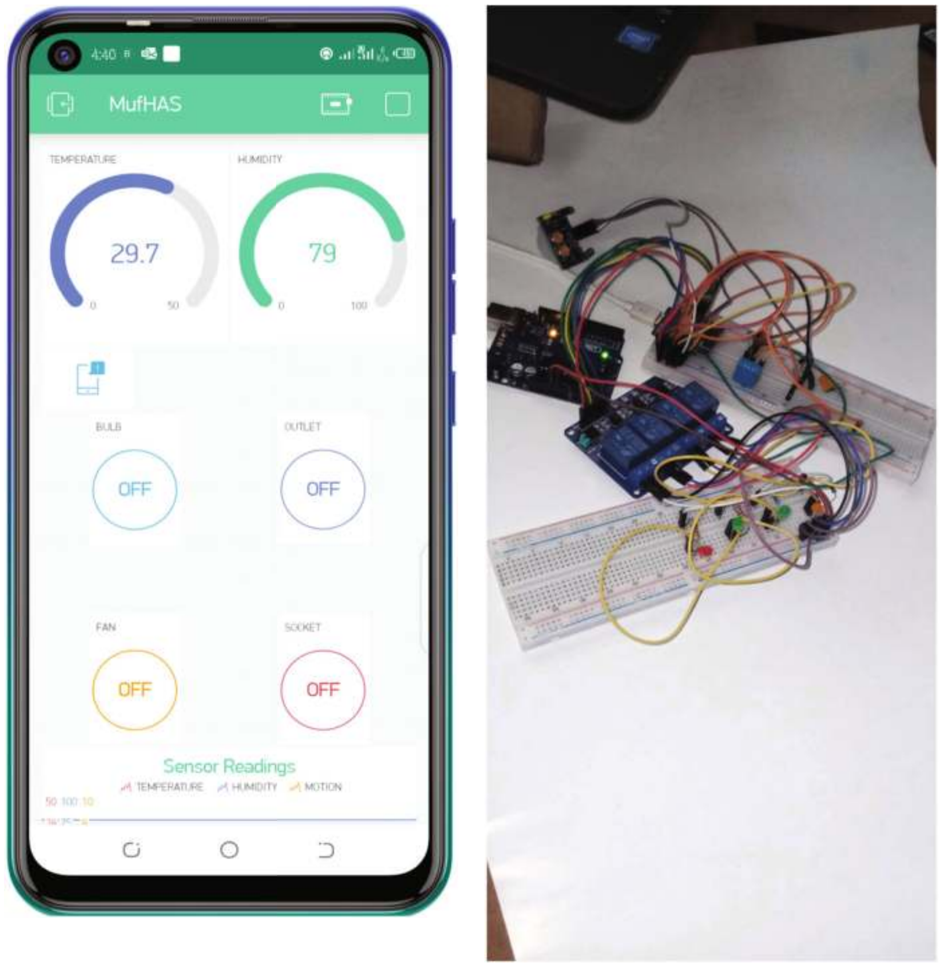


Figure 4: An illustration of the CNN architecture for intruder detection in the MufHAS architecture.

the Softmax activation classification process. The preprocessing of images is an essential aspect of making images fit for the modeling process. The initial process involves a range of steps applied to the input to achieve an acceptable form for the feature extraction. Preprocessing is very important because feature extraction might not yield desired results if the inputs are not properly preprocessed [58]. As a result, inputs were preprocessed using the denoising and CLAHE operations before being passed into the CNN architecture. The former preprocessing operation allows noise removal while the latter improves image quality through a contrast improvement strategy.

Filter counts of 32, 64 128, and 256 were used for the convolutional layers' first, second, third, and fourth blocks. Kernel sizes of 3×3 were used in each convolutional operation spanning the four blocks. After each block of the convolutional operation, we zero-padded the output before passing it to the max pooling operation. The peculiarity of the problem being addressed in this study showed that the max pooling operation demonstrated better performance than the average pooling operation. The padding size of 1×1 was used in the zero-padded layer, and the kernel size of 2×2, 3×3, 2×2, and 3×3 was used for the max pooling layers of first, second, third, and fourth convolutional blocks, respectively. The strides of 1×1 were applied to all convolutional operations while the size of 1×1 and 2×2 was alternated in the max pooling layers across the architecture. We applied some regularization techniques to the CNN model



(a) (b)

FIGURE 5: (a) Home control interface. (b) Initial state of the demo.

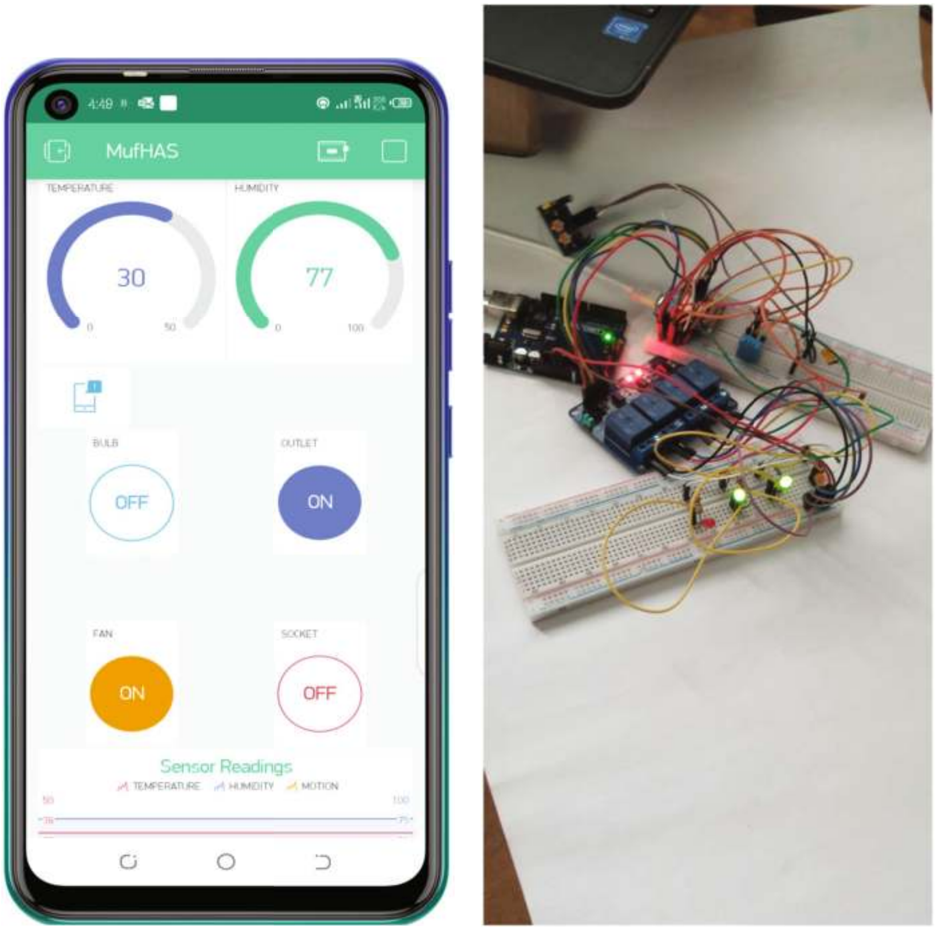
to eliminate overfitting the model. First, we used the L2 with a value of 0.0002 in each convolutional operation and dense layer and added a dropout layer of 0.5 rates after the flatten layer. Considering the multiclass nature of the dataset to which the CNN architecture was applied, we used the Softmax activation function to obtain the probability map for each class of image samples in the dataset.

4. Implementation Details

Our system was implemented using a prototype setup. The prototyping was carried out with the use of hardware and software components. The software components we used are the Blynk software and Arduino IDE. The Android-based mobile application was designed and configured with the Blynk platform. The Blynk platform is an IoT platform for designing and configuring mobile applications for IoT systems. We used the Blynk platform to design the GUI of our mobile application and configure the pins of our hardware for home control and monitoring. It supports cloud data storage, display, and visualization of real-time data generated by sensors. The instructions for the configuration of the microcontroller board, sensors, and other home appliances were developed using the Arduino IDE. The Arduino IDE is a cross-platform application for Windows, Linux, and MAC operating systems. Commands to control home appliances were also developed and uploaded to the board via the Arduino IDE.

4.1. Hardware Components. The hardware components of our system are an ESP8266 Wi-Fi module, an ESP32-CAM module, an Arduino UNO board, a 5 V four-channel relay module, jumper cables, breadboards, USB cables, LEDs, resistors, and sensors. Jumper cables (male to female and female to female) were used to connect two or more hardware to each other. The resistors were used to reduce current flow, while the breadboards were used to link the hardware components. The major hardware components are further explained.

4.1.1. ESP8266 Wi-Fi Module. The ESP8266 Wi-Fi module is a self-contained system on chip (SOC) with an integrated TCP/IP protocol stack. It is a low-cost Wi-Fi module that gives most microcontrollers access to Wi-Fi networks, thereby enabling the connectivity of several home automation technologies and devices. It allows the developed system to communicate with the internet and grants access to the



(a) (b)

FIGURE 6: (a) GUI's command to switch on two appliances. (b) System's response to power on two appliances.

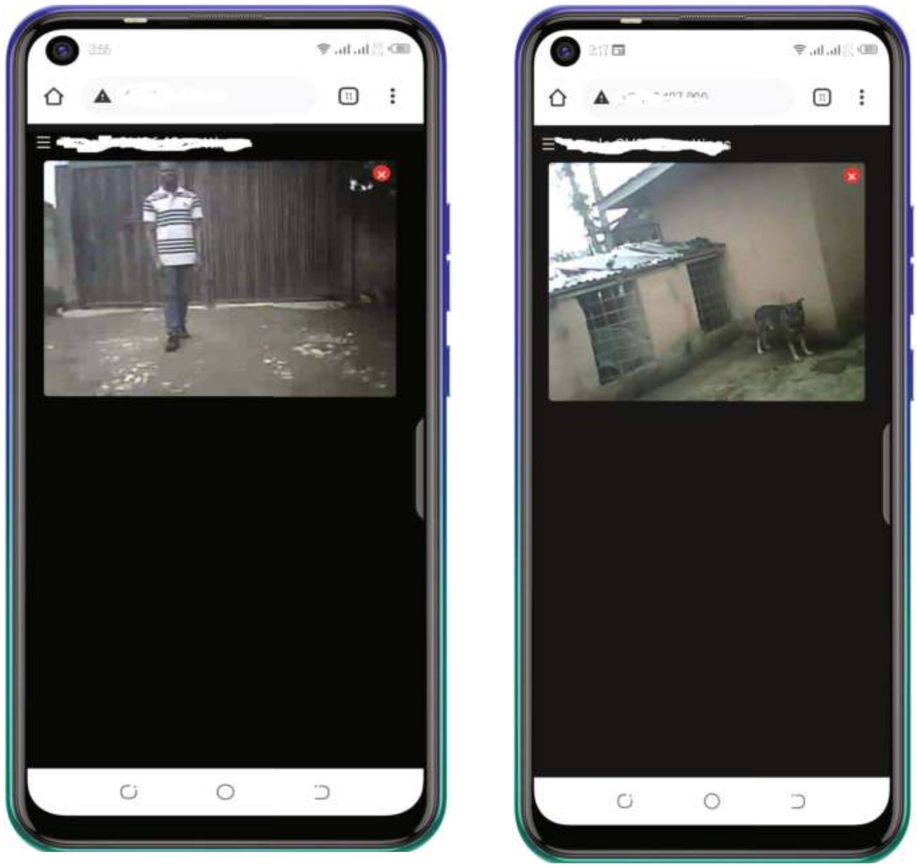
home to the user from anywhere. The storage capability of the ESP8266 Wi-Fi module allows it to be integrated with sensors and other home devices. The ESP8266 served a dual function of the microcontroller board and Wi-Fi module in our work.

4.1.2. Relay Module. A 5 V four-channel relay module was used in building the prototype MufHAS. A relay is an electrical device that controls devices and can switch much higher voltages and currents to normal microcontroller output using low voltage as input. The relay module is used to control a large current from the microcontroller. Each channel of the relay module has an LED indicator to show that the specified port is on.

4.1.3. DHT11 Temperature and Humidity Sensor. The DHT11 sensor was used to measure the home's temperature and humidity levels. The DHT11 has a fast response and interference ability. A DHT11 sensor is a low-cost, embedded sensor that serially provides temperature and humidity values via wire protocol. The humidity range for a DHT11 sensor is 20% to 80% relative humidity, but there are claims of up to 95% in some datasheets. The temperature range for the DHT11 sensor is 0 to 50°C [59].

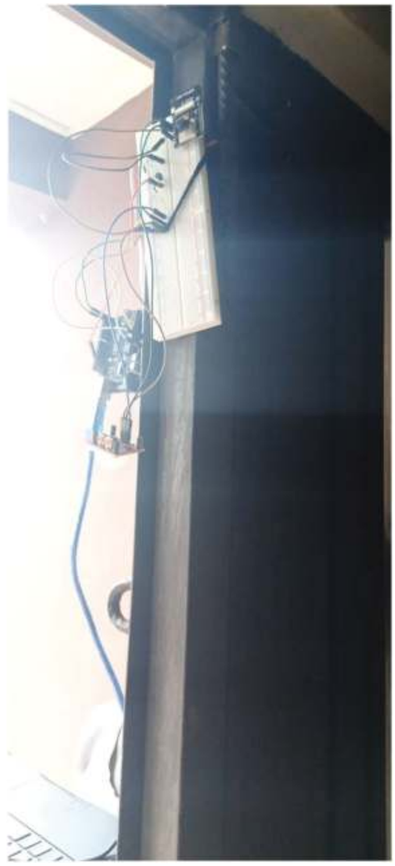
4.1.4. HC-SR501 PIR Motion Sensor. An HC-SR501 is a motion-detecting sensor used in an IoT environment for detection motion and used in security systems, garage doors, gates, and automatic switching of lights in some systems. We use the PIR sensor to detect movements in our system, and it sends a notification to the user immediately for an action to be taken. It is a low-cost, sensitive, easy-to-use, and accurate sensor for motion detection. It supports a wide-range communication that has an in-built voltage regulator.

4.1.5. ESP32-CAM. An ESP32 camera module is a low-cost, low-energy development board for video streaming and image capturing in IoT prototyping. It has a combined Wi-Fi and Bluetooth chip based on a 32-bit CPU. The camera module has a resolution of up to 1600 by 1200 and supports up to 4 GB storage with an SD card. We use the ESP32-CAM to ensure security in prototyping our work through video streaming. The user can view the situation of the home once an alert of motion is received via the phone interface. To program the camera board, an Arduino UNO board was



(a) Video streaming interface (human captured) (b) Video streaming interface with an animal captured

Figure 7: Continued.



(c) Surveillance setup

Figure 7

used for the USB-to-TTL module because the ESP32-CAM does not have a programmer chip.

4.2. Datasets and Training of the CNN Architecture. The home training dataset consisting of about 4000 samples of different human postures was applied to the CNN for the training process. We found four major postures suggestive of what the camera module in the MufHAS architecture is expected to capture for image recognition. For each posture, samples were drawn into groups demonstrating an intruder and the home occupant. This resulted in eight classes for the multiclass classification used in the CNN architecture. Using the ratio of 80 : 20, samples were split for training and testing the CNN model. Adam optimizer was then used to train the CNN model with the configuration of 0.001 learning rate, 0.9 for beta1, 0.999 for beta2, and epsilon of $1e-8$. The training of the CNN model was carried out for 100 epochs under the computational environment with the configuration of Intel (R) Core i5-7500 CPU 3.40 GHz, 3.41 GHz; RAM of 16 GB; 64-bit Windows 10 OS.

4.3. Results and Discussion. We present a detailed explanation of our system's setup, configuration, and experimentation in this section. A prototype implementation using IoT hardware was used to test the functionality of our system. The designed MufHAS was tested using an ESP8266 Wi-Fi module, a PIR motion sensor, an ESP32-CAM module, an Arduino UNO board, breadboards, a 5 V four-channel relay module, a DHT11 sensor, and an Android-based smartphone. The microcontroller board, camera module, and mobile application were programmed to communicate on the Arduino IDE. Network credentials for connectivity were declared in the coding phase of the configuration. A wireless connection was established through the Wi-Fi module. The prototype system detects movement, has a video streaming module, and measures the home environment's temperature, and humidity controls a bulb, fan, and socket, and an extra point was left out for scalability. The home appliances were interfaced with the relay module to have the appropriate current flow. The mobile application receives a signal from the system through the Wi-Fi module. The initial state of the system and mobile application are presented in Figures 5(a) and 5(b). All appliances were yet to be powered on. The system starts recording the measured environmental conditions once communication is established.

When the user issues a command on the mobile application, the corresponding home appliance gives an indicator on the LEDs and relays board to depict the switched device on or off. Figure 6(a) gives the GUI of the mobile application, indicating a command to switch on two of the

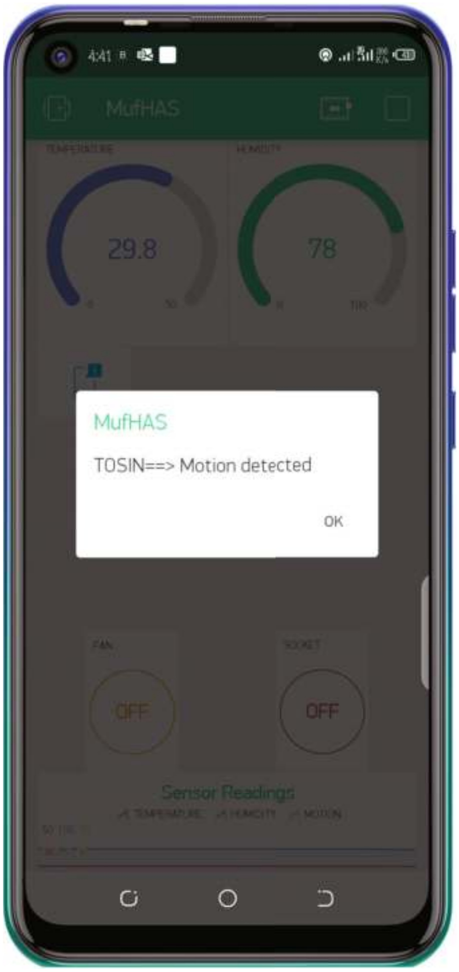


FIGURE 8: Movement notification.

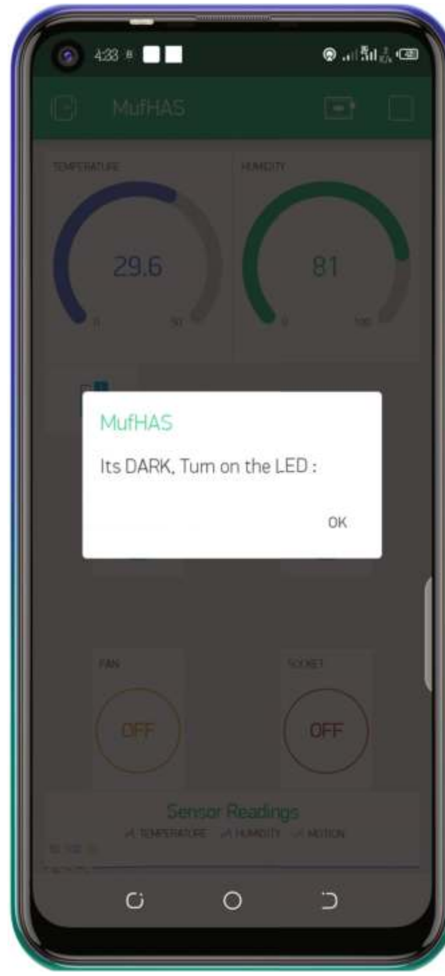


FIGURE 9: Intelligent notification about the light.

appliances, while Figure 6(b) shows the prototype's indicator. Time to establish communication between the mobile application and the smart home shows the system is effective in terms of connectivity even when not close to the system. Also, the humidity and temperature levels are measured at intervals by the system. The mobile application indicator reveals that the user can gather information about the home's environmental conditions and the level of the measured parameters. The user monitors the home environment over the internet via the video-streaming interface. The camera is placed at the entrance of the home to monitor the surroundings, record, and capture events. With the aid of the SD card module embedded in the camera module, recordings of the home environment are stored. Also, the user can capture images as they appear on the screen. The video streaming interface of the security module of our system, with a human and an animal captured, is shown in Figures 7(a) and 7(b), while Figure 7(c) shows the hardware setup. As explained earlier, movements are detected in the house, and a notification is sent to the user for necessary action. Figure 8 shows the GUI of the mobile application with the displayed notification about a movement detected in the house. Our system also gives an intelligent notification to the user to switch on the light when it is dark or switch off the light when it is bright. This is to ensure energy efficiency in the home. The system was not designed to automatically turn on or off the light because the user might be on the premises and want the light on at that instance. Figure 9 shows the GUI of the notification to switch on the light. The mobile application displays a graphical reading of the sensors installed in the home for the user to visualize the trend of the environmental conditions. These readings are displayed based on real-time data acquired from the sensors and stored in the cloud platform of the Blynk application. The data generated can be used for weather prediction and analysis of the environment in the future. The graphical display is presented in Figure 10.

4.4. CNN Image Classification Experimentation. The proposed CNN model is tested to evaluate the feasibility and efficiency of our proposed method for home security. The

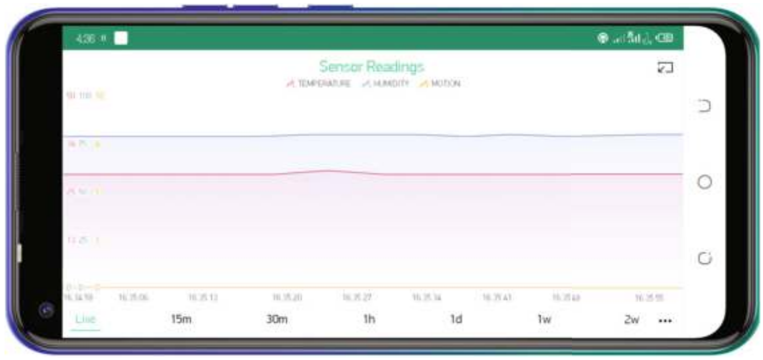


Figure 10: Graphical display of sensor readings.

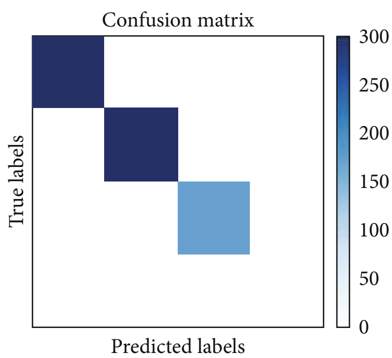


Figure 11: Confusion matrix of the trained CNN model when applied for prediction on some samples for the classification problem of detecting home occupants and intruders.

dataset for the training was downloaded from the CV image database [60], a dataset containing 3,884 images of the human motion of 972 pedestrians. Each person in the dataset was captured from four different angles to show different walking patterns. We divided the dataset into four groups: walking, jumping, limping, or running. Each group contained similar captured angles of different people in the same position. After the grouping, we separated the groups into two other subgroups that we termed home occupants and intruders for the classification process.

The CNN model yields an accuracy of 98%. A performance measurement required in DL classification is the confusion matrix. It presents an output table for the classification model, combining the predicted and actual values. The confusion matrix is shown in Figure 11. The accuracy of the CNN deep learning experiment shows that smart homes can be made more intelligent for the security of home occupants and their properties. The classification proves that motion patterns can be used to distinguish and identify entities in the home environment discretely and efficiently.

The performance of the CNN architecture during training is captured in Figures 12 and 13. The curve of the loss values obtained for the training and validation samples during the 100 training epochs is graphed in Figure 12. In the first 20 epochs, we see the unstable pattern of the validation samples as applied to the CNN model while the training dropped steadily. Interestingly, the result obtained in epochs 40–100 demonstrates that the loss values for the training and evaluation phases are in tune and that the CNN model effectively learns the classification problem. Similarly, the accuracy of both the training and evaluation phases rose consistently in alignment with the loss values sustained. The accuracy values for the two phases showed a stable output around 60-100 epochs of the training process.

Using the trained model, test samples were served to it, and the model's performance was evaluated using the accuracy, precision, recall and F1-score, specificity, and Cohen's kappa metrics. Table 2 presents the evaluation table using the precision, recall, F1-score, specificity, and Cohen's kappa values to compare the classifiers that were compared.

The precision evaluates the model's relevancy and quantifies the prediction about the number of positive values in the model. It is calculated as

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}. \quad (5)$$

The percentage of rightly classified actual positive predictions in the model is evaluated by calculating the recall as

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}. \quad (6)$$

The F1-score is a measure of accuracy based on the precision and recall values of a test. It is calculated as

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (7)$$

Specificity is used to ascertain the proportion of actual negative cases that the model rightly predicted. Specificity is calculated as follows:

$$\text{Specificity} = \frac{\text{True Negatives (TN)}}{\text{True Negatives (TN)} + \text{False Positives (FP)}}. \quad (8)$$

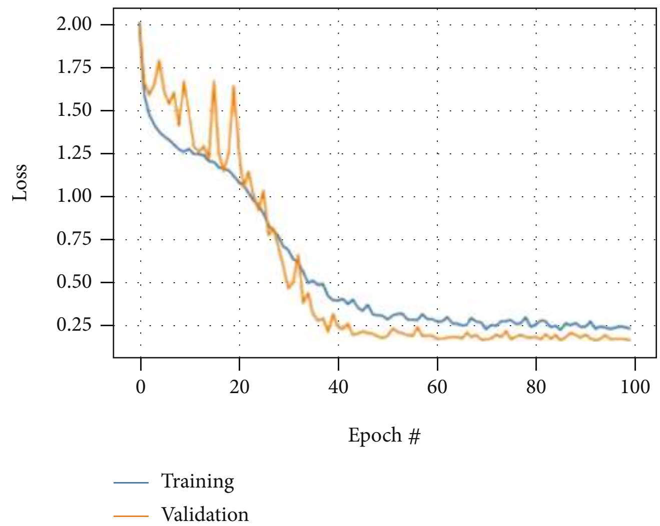


Figure 12: Training and validation loss obtained during training of the CNN architecture in the classification of walk postures of home occupants from intruders.

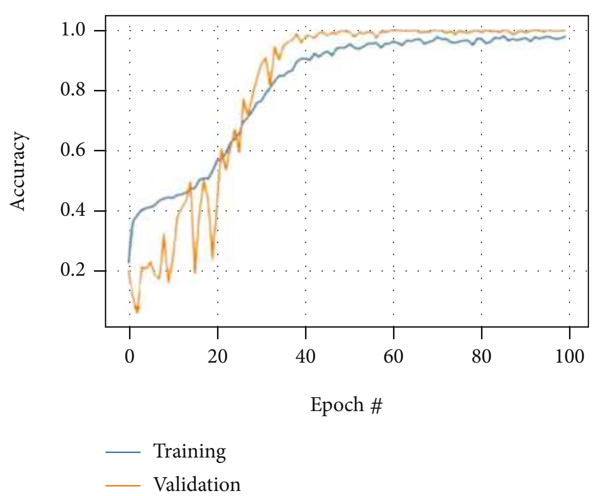


Figure 13: Training and validation accuracy obtained during training of the CNN architecture in the classification of walk postures of home occupants from intruders.

Cohen's kappa (K) is a measure of agreement between two raters in a test and is also used to assess a classification model's performance. It is calculated as follows:

$$K = \frac{\rho_0 - \rho_e}{1 - \rho_e}, \quad (9)$$

where ρ_0 is the overall accuracy of the model and ρ_e is the measure of the agreement between the model's predicition and the actual class values.

The prediction procedure on the trained CNN model was achieved using 775 samples, and the confusion matrix obtained is shown in Figure 12. The y-axis of the figure shows the true labels, while the x-axis is the predicted labels for the samples drawn from the test set. The diagonal of our confusion matrix confirms the CNN architecture's impressive performance, which is almost similar to what is obtained in a well-performing model where only the diagonals contain values. On row 1 of the matrix, we have 300, 0, 0, and 0, which means our CNN model successfully classified the 300 samples for the class represented by the row. The same follows for the second row, which has 0, 300, 0, and 0. The third row shows the values of 0, 0, 174, and 1, and the fourth output is 0, 0, 0, and 0 since no sample was passed for this class. The CNN prediction accuracy yielded a value of 0.998 as compared to the SVM, KNN, and complex decision tree yielding an accuracy of 0.900, 0.867, and 0.830, respectively, and demonstrating a very good performance of the CNN model in learning to classify home occupants from an intruder in the proposed MufHAS architecture. This performance also rippled into the values obtained for the precision, recall, F1-score, specificity, and Cohen's kappa, as listed in Table 2.

We compare our work with an existing deep learning-based smart mat monitoring system for identity information, stepping position, and activity status in a smart home or building environment [61]. The authors used a CNN model to predict valid users in a room, the authors tested their model using a dataset of 1000 samples, and the results of the CNN training model yielded an accuracy of 96%. MufHAS-CNN training yields an accuracy of 99.8%. However, it is noteworthy that the comparison with [60] was restricted to only one metric because the authors of the existing work focused their performance evaluation only on the prediction accuracy metric measure to evaluate their model. Also, the authors evaluated their implemented model based on triboelectric output signals and output voltages of the individual walking pattern, which they also applied in training their proposed CNN model, basically for recognition accuracy. The input values are different from the current implementation discussed in this study, hence the limitation in comparison. Moreover, the dataset used in [60] was not accessible to run or replicate any further analysis and comparison using other stated performance metrics.

Table 2: Results obtained from the prediction for the classification problem of house occupant and intruders using the proposed CNN architecture.

Metrics	Accuracy	Precision	Recall	F1-score	Specificity	Cohen's kappa
CNN	0.998	0.998	0.998	0.997	1.000	0.998
SVM	0.900	0.833	0.714	0.769	0.957	0.706
KNN	0.867	0.667	0.857	0.750	0.870	0.661
Complex decision tree	0.830	0.667	0.571	0.615	0.913	0.734

We also compared the proposed CNN model with the models presented in [24]. More so, similar to the above-highlighted limitation in metric comparisons with some related existing studies, the comparison with the model implementation in [25] is limited to accuracy because it is the only similar metric that was employed in the previous works. Table 3 compares our CNN model and previous works using the accuracy. This proves that the proposed CNN model can enhance home security in IoT smart home automation. With the proposed CNN models, smart home automation applications can be enhanced to detect intruders based on motion patterns. The users can also identify intruders based on the surveillance camera and models for detection, classification, and differentiation of motion patterns. The smart home applications' security notifications and alerts will also be based on detected motion patterns. Thus, home security is enhanced in the smart home environment.

Table 3: Comparison with previous works.

Model	Accuracy (%)
CNN [61]	96.00
CON-RNN [24]	91.79
Small Conv-LSTM [24]	91.50
Large Conv-LSTM [24]	91.50
Conv-CWRNN [24]	92.38
Conv-DCWRNN [24]	93.02
Proposed CNN-MufHAS	99.80

4.5. Home Temperature and Humidity Evaluation. The temperature and humidity sensor was connected to the system and placed in a strategic location in the home to measure both the humidity and temperature values and display them on the mobile application. The mobile application was refreshed hourly, and the output displayed on the application indicates that the sensor receives a signal from the environment. Also, the values displayed correlated with the weather forecast for the periods. This shows that communication between the home environment and the mobile application is effective. The readings were taken hourly for seven days, and the graph was plotted using MATLAB. Figure 14 presents the graph for the temperature readings of MufHAS, while Figure 15 shows the MufHAS' graph of the humidity readings. From the graphs presented, $n = 1, 2 \cdots, N$ is the nth hour collected, and the range for temperature readings is between 0°C and 30°C. The range of humidity is between 60 and 95%. Figures 16(a) and 16(b) show the output of readings from MufHAS plotted against online daily weather readings for temperature and humidity, respectively.

Based on the graphs in Figures 16(a) and 16(b), we calculated the mean percentage error to measure how close the designed MufHAS mobile application readings are to the daily weather forecast. The mean percentage error of the readings was calculated as follows:

$$\%\text{Error} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\left|T_i^M - T_i^W\right|}{T_i^W}\right), \qquad (10)$$

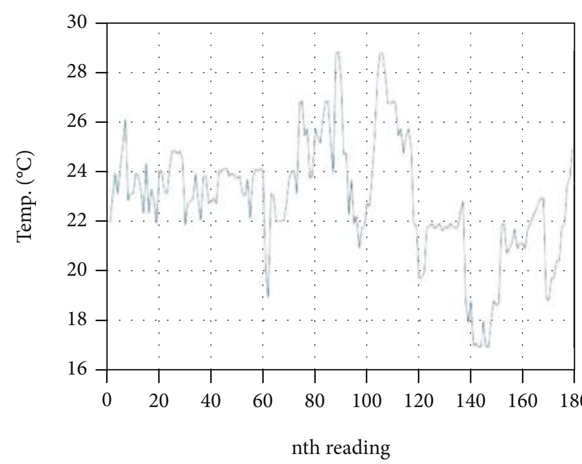


Figure 14: Graph of temperature readings.

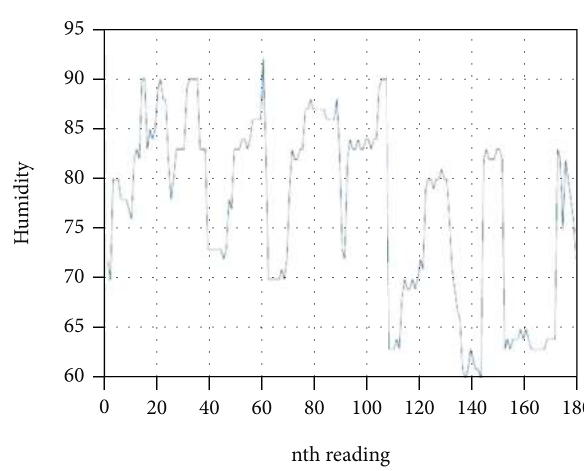
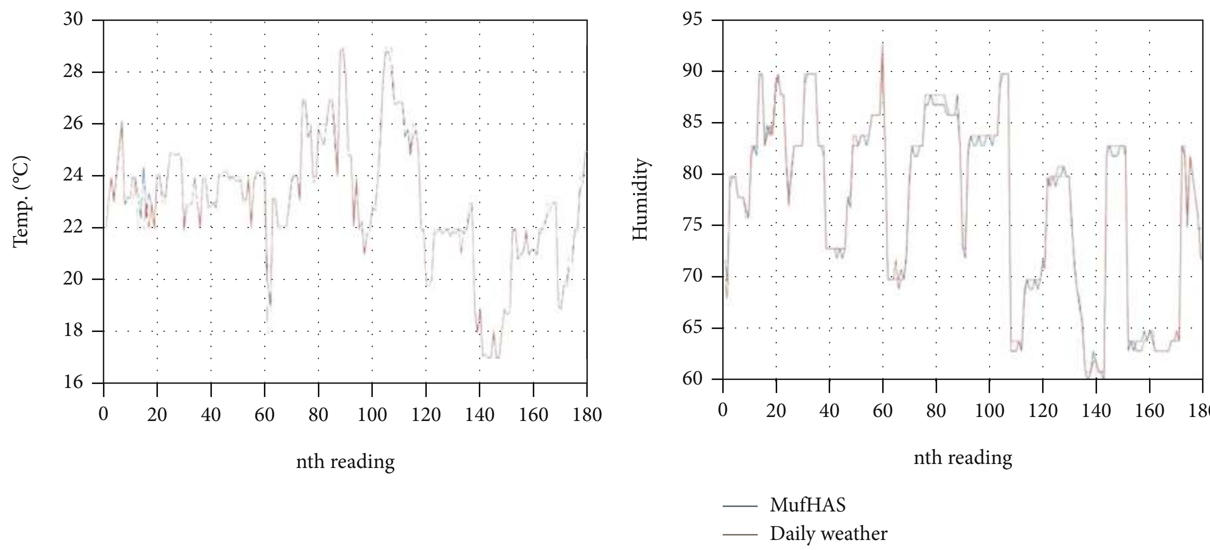


Figure 15: Graph of humidity readings.



(a) Comparison graph for temperature

(b) Comparison graph for humidity

Figure 16

where T_i^M is the MuShAS temperature readings in °C for day i and T_i^W is the online delay weather temperature readings in °C for day i.

The temperature calculated percentage error for $N = 225$ measurements is 0.9057%, and the humidity's calculated percentage error for $N = 180$ measurements is 0.6256%. These show that the readings of the MufHAS mobile application are of good accuracy and be relied on to measure environmental conditions in the home with good results.

5. Conclusion and Future Recommendation

Improved convenient living, a healthy lifestyle, comfortability, and home security are areas of interest and development. The elderly, handicapped, and sick need to reduce daily activities that can stress them and negatively impact their health. To this end, a smart home automation system that can facilitate local and global monitoring, control, and safety of the home was developed. This work contributes to the existing research in home automation with the design and development of a multifunctional Android-based mobile application for the smart home automation domain. We have proposed an approach to enhance home security using the CNN deep learning model to classify and detect intruders in the home. The detection is based on the identification of motion in the home environment. Using this method shows that users will have enhanced security of their houses while having minimal disturbance from notifications.

The proposed method intends to eliminate frequent notifications and false notifications in a smart home automation system. The drawback of our proposed method is the detection of multiple movements at a time. The training and classification models were based on the movement of a person at a time. The feature extraction, classification, and accurate prediction of multiple movements either from animals or humans are not yet established. Therefore, the motion detection module can be further developed for accurate classification and prediction of the exact motions detected when several movements are detected in the system. Another limitation of the system is not considering emotions as an influence on the movement of human beings. Emotions can make an individual's walking pattern differ from the regular movement, which may cause a false classification. Also, data generated from sensors can be used for analysis and prediction that can enhance the home automation system's functionality and improve users' lifestyles.

For the next phase of this research, we plan to implement more sensors in the system and test over a couple of weeks to generate data from the home environment to be used for analysis. A PIR motion sensor and ESP32-CAM attribute value to detected objects, and we intend to gather real-time motion images using our IoT devices via the cloud database to form a dataset for training and classification using the proposed support vector machine algorithm. A web interface is used for video streaming in our current work. As part of the future work, a mobile application module for video streaming will be incorporated, supporting real-time, cloud-based storage of media. Also, the deep learning model will be integrated into the mobile application for full functionality. As future work, a mechanism to classify friends and extended family members will also be considered for the system not to classify such a category of people as intruders. Electrical appliances are also considered to be fully implemented in the next phase. For future work, we intend to incorporate sensors in smart home healthcare into the system to assist in the remote monitoring of patients and use deep learning algorithms for our classification.

Abbreviations

Ha: Home appliances
Nc: Network connectivity
Hc: Home control

Hs: Home sensor
Ms: Motion sensor
Ho: Home occupant (where the Ho is determined by the walking, running, limping, or jumping patterns of humans or pets in the home)
Wp: Motion patterns (walking, jumping, limping, or running)
HWp: Home occupant's walking pattern
Hp: Home pet
In: Intruder
Cs: Cloud storage
M: Motion
D: Darkness
LDR: Light-dependent resistor
Cs: Cloud storage.

Data Availability

All data used in the study can be found in the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] B. Cele, *Quarter One Crime Statistics*, South Afrcian Governmentq, Pretoria, 2021.

[2] K. Ashton, "That 'internet of things' thing," *RFiD Journal*, vol. 22, no. 7, pp. 97–114, 2009.

[3] O. Taiwo, L. A. Gabralla, and A. E. Ezugwu, "Smart home automation: taxonomy, composition, challenges and future direction," in *International Conference on Computational Science and Its Applications*, Cagliari, 2020.

[4] N. King, "Smart home-a definition," 2003, http://www.housinglin.org.uk/_assets/Resources/Housing/Housing_advice/Smart_Home_-_A_definition_September_2003.pdf.

[5] F. Mehmood, I. Ullah, S. Ahmad, and D. Kim, "Object detection mechanism based on deep learning algorithm using embedded IoT devices for smart home appliances control in CoT," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–17, 2019.

[6] D. Liciotti, M. Bernardini, L. Romeo, and E. Frontoni, "A sequential deep learning application for recognising human activities in smart homes," *Neurocomputing*, vol. 396, pp. 501–513, 2020.

[7] J. Jaihar, N. Lingayat, P. S. Vijaybhai, G. Ventakesh, and K. P. Upla, "Smart home automation using machine learning algorithms," in *International Conference for Emerging Technology (INCET)*, Belgaum, India, 2020.

[8] D. Popa, F. Pop, C. Serbanescu, and A. Castiglione, "Deep learning model for home automation and energy reduction in a smart home environment platform," *Neural Computing and Applications*, vol. 31, no. 5, pp. 1317–1337, 2019.

[9] Z. Peng, X. Li, and F. Yan, "An adaptive deep learning model for smart home autonomous system," in *International Conference on Intelligent Transportation, Big Data and Smart City (ICITBS)*, Vientiane, Laos, 2020.

[10] S. K. Shah, Z. Tariq, and Y. Lee, "Audion IoT analytics for home automation safety," in *IEEE International Conference on Big Data (Big Data)*, Seattle, WA, USA, 2018.

[11] A. Oppermann, "Towards data science - what is deep learning and how does it work?," 2019, September 2021, https://towardsdatascience.com/what-is-deep-learning-and-how-does-it-work-2ce44bb69ac.

[12] M. Khan, J. Seo, and D. Kim, "Towards energy efficient home automation: a deep learning approach," *Sensors*, vol. 20, no. 24, p. 7187, 2020.

[13] M. Sahu and R. Dash, "A survey on deep learning: convolution neural network (CNN)," in *Intelligent and Cloud Computing*, Smart Innovation, Systems and Technologies, 2021.

[14] G. Lobaccaro, S. Carlucci, and E. Löfström, "A review of systems and technologies for smart homes and smart grids," *MPDI Energies*, vol. 9, no. 5, p. 348, 2016.

[15] Z. Xiaodong and Z. Jie, "Design and implementation of smart home control system based on STM32," in *30th Chinese Control and Decision Conference (2018 CCDC)*, Shenyang, 2018.

[16] S. Bimenyimana, A. Ishimwe, G. N. O. Asemota, C. M. Kemunto, and L. Li, "Web-based design and implementation of smart home appliances control system," *IOP Conference Series: Earth and Environmental Science*, vol. 168, article 012017, 2018.

[17] T. S. Gunawan, R. H. I. Yaldi, M. Kartiwi et al., "Prototype design of smart home system using internet of things," *Science*, vol. 7, no. 1, pp. 107–115, 2017.

[18] U. Singh and M. A. Ansari, "Smart home automation system using internet of things," in *2nd International Conference on Power Energy, Environment and Intelligent Control (PEEIC)*, Greater Noida, 2019.

[19] H. Aadel and J. Ali, "Design and implementation prototype of a smart house system at low cost and multi-functional," in *Future Technologies Conference (FTC)*, San Francisco, 2016.

[20] A. Parsa, T. A. Najafabadi, and F. R. Salmasi, "Implementation of smart optimal and automatic control of electrical home appliances (IOT)," in *IEEE 2017 Smart Grid Conference (SGC)*, Tehran, Iran, 2017.

[21] R. D. Manu, S. Kumar, S. Snehashish, and K. S. Rekha, "Smart home automation using IoT and deep learning," *International Research Journal of Engineering and Technology*, vol. 6, no. 4, pp. 1–4, 2019.

[22] R. Majeed, N. A. Abdullah, I. Ashraf, and Y. B. Zikria, "An intelligent, secure, and smart home automation system," *Scientific Programming*, vol. 2020, 14 pages, 2020.

[23] A. Brenon, F. Portet, and M. Vacher, "Arcades: a deep model for adaptive decision making in voice controlled smart-home," *Pervasive and Mobile Computing*, vol. 49, pp. 92–110, 2018.

[24] N. Neverova, C. Wolf, G. Lacey et al., "Learning human identity from motion patterns," *IEEE Access*, vol. 4, pp. 1810–1820, 2016.

[25] S. K. Saravanan, A. M. Nainar, and S. V. Marichamy, "Android based smart automation system using multiple authentications," *IRE Journal*, vol. 3, no. 6, pp. 60–65, 2019.

[26] W. A. Jabbar, M. H. Alsibai, N. S. Amran, and S. K. Mahayadin, "Design and implementation of IoT - based automation system for smart home," in *International Symposium on Networks, Computers and Communications*, Rome, Italy, 2018.

[27] L. A. Ajao, J. G. Kolo, E. A. Adedokun, O. M. Olaniyi, O. C. Inalegwu, and S. K. Abolade, "A smart door security-based home automation system: an internet of things," *SciFed Journal of Telecommunication*, vol. 2, no. 2, 2018.

[28] H. Singh, V. Pallagani, V. Khandelwal, and U. Venkanna, "IoT based smart home automation system using sensor node," in *Fourth International Conference on Recent Advances in Information Technology(RAIT-2018)*, Dhanbad, India, 2018.

[29] O. Taiwo, A. E. Ezugwu, N. Rana, and S. M. Abdulhamid, "Smart home automation system using Zigbee, Bluetooth, and Arduino technologies," in *Computational Science and Its Applications-ICCSA 2020*, pp. 587–597, Springer, Cham, 2020.

[30] O. Taiwo and A. E. Ezugwu, "Smart healthcare support for remote patient monitoring during Covid-19 quarantine," *Informatics in Medicine Unlocked*, vol. 20, no. 100428, pp. 1–12, 2020.

[31] M. U. Alam, R. Tiwari, and J. Sharma, "Design and implementation of smart home automation system using LabVIEW," *International Journal for Scientific Research and Development*, vol. 8, no. 5, pp. 515–520, 2020.

[32] M. Naing and N. S. Hlaing, "Arduino based smart home automation system," *International Journal of Trend in Scientific Research and Development*, vol. Volume-3, no. Issue-4, pp. 276–280, 2019.

[33] S. R. Jena, R. Shanmugam, A. Tyagi, and R. K. Patjoshi, "Implementation of voice recognized LED using Arduino Uno Micro," *International Journal of Future Generation Communication and Networking*, vol. 13, no. 3, pp. 2233–2241, 2020.

[34] O. Taiwo and A. E. Ezugwu, "Internet of things-based intelligent smart home control system," *Security and Communication Networks*, vol. 2021, 17 pages, 2021.

[35] S. Mahmud, S. Ahmed, and K. Shikder, "A smart home automation and metering system using internet of things (IoT)," in *2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, Dhaka, Bangladesh, 2019.

[36] L.-D. Liao, Y. Wang, Y.-C. Tsao et al., "Design and validation of a multifunctional Android-based smart home control and monitoring system," *IEEE Access*, vol. 7, pp. 163313–163322, 2019.

[37] L. Smirek, G. Zimmermann, and M. Beigl, "Just a smart home or your smart home –a framework for personalized user interfaces based on eclipse smart home and universal remote console," *Procedia Computer Science*, vol. 98, pp. 107–116, 2016.

[38] M. Wang, G. Zhang, C. Zhang, J. Zhang, and C. Li, "An IoT-based appliance control system for smart homes," in *Fourth International Conference on Intelligent Control and Information Processing (ICICIP)*, Beijing, China, 2013.

[39] J. Chhabra and P. Gupta, "IoT based smart home design using power and security management," in *International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH)*, Noida, 2016.

[40] Y. Li, "Design of a key establishment protocol for smart home energy management system," in *Fifth International Conference on Computational Intelligence, Communication Systems and Networks*, Madrid, 2013.

[41] H. Ghayvat, S. Mukhopadhyay, X. Gui, and N. Suryadevara, "WSN- and IOT-based smart homes and their extension to smart buildings," *Sensors*, vol. 15, no. 5, pp. 10350–10379, 2015.

[42] R. Piyare, "Internet of things: ubiquitous home control and monitoring system using Android based smart phone," *International Journal of Internet of Things*, vol. 2, no. 1, pp. 5–11, 2013.

[43] R. K. Kodali, V. Jain, S. Bose, and L. Boppana, "IoT based smart security and home automation system," in *International Conference on Computing, Communication and Automation (ICCCA)*, Noida, 2016.

[44] A. Anitha, "Home security system using internet of things," *IOP Conference Series: Materials Science and Engineering*, vol. 263, article 042026, 2017.

[45] M. Barnachon, S. Bouakaz, B. Boufama, and E. Guillou, "Ongoing human action recognition with motion capture," *Pattern Recognition*, vol. 47, no. 1, pp. 238–247, 2014.

[46] R. El-Azab, "Smart homes: potentials and challenges," *Clean Energy*, vol. 5, no. 2, pp. 302–315, 2021.

[47] R. J. Orr and G. D. Abowd, "The smart floor: a mechanism for natural user identification and tracking," in *CHI 00: extended abstracts on human factors in Computing Systems*, The Hague The Netherlands, 2000.

[48] J. Suutala and J. Roning, *Towards the Adaptive Identification of Walkers: Automated Feature Selection of Footsteps using Distinction Sensitive LVQ*, International Workshop on Processing Sensory Information for Proactive Systems (PSIPS), 2004.

[49] N. Rahman, K. Kalpoma, and T. Hasan, "Automated person identification system using walking pattern biometrics," *International Journal of Scientific and Engineering Research*, vol. 6, pp. 98–102, 2015.

[50] H. Sidenbladh, "Detecting human motion with support vector machines," in *17th IAPR International Conference on Pattern Recognition*, Cambridge, 2004.

[51] S. Khan, M. H. Javed, E. Ahmed, S. A. A. Shah, and S. U. Ali, "Facial recognition using convolutional neural networks and implementation on smart glasses," in *2019 International Conference on Information Science and Communication Technology (ICISCT)*, Karachi, Pakistan, 2019.

[52] H.-R. C. Cheng, G.-Z. Cao, C.-H. Li, A. Zhu, and X. Zhang, "A CNN-LSTM hybrid model for ankle joint motion recognition method based on sEMG," in *International Conference on Ubiquitous Robots (UR)*, Kyoto, Japan, 2020.

[53] O. N. Oyelade and A. E. Ezugwu, "Deep learning model for improving the characterization of coronavirus on chest X-ray images," medRxiv, 2020.

[54] M. Ismail, "Towards data science - convolutional neural networks' mathematics," 2020, September 2021, https://towardsdatascience.com/convolutional-neural-networks-mathematics-1beb3e6447c0.

[55] X. Qi and J. Liu, "Comparison of support vector machine and softmax classifiers in computer vision," in *Second International Conference on Mechanical Control and Computer Engineering*, Harbin, China, 2017.

[56] Y. Li, T. Wang, B. Kang et al., "Overcoming classifier imbalance for long-tail object detection with balanced group softmax," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020.

[57] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 39, pp. 1137–1149, 2016.

[58] S. Ahlawat and A. Choudhary, "Hybrid CNN-SVM classifier for handwritten digit recognition," *International Conference on Computational Intelligence and Data Science*, vol. 167, pp. 2554–2560, 2020.

[59] Chewett, "The Chewett blog, electronics,code and contraptions," 2017, November 2020, https://chewett.co.uk/blog/309/dht11-sensor-review-code-examples/.

[60] "CVonline: image databases," 2019, August 2021, https://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.htm#people.

[61] Q. Shi, Z. Zhang, T. He et al., "Deep learning enabled smart mats as a scalable floor monitoring system," *Nature Communications*, vol. 11, no. 4609, pp. 1–11, 2020.