International Journal of Interdisciplinary Innovative Research & Development (IJIIRD)
ISSN: 2456-236X
Vol. 10 Issue 01 | 2025

Development of a Home Security Grille System with Face Recognition and Pneumatic Actuation

Mr. Pravin R. Hoge[1], Mr. Ritheshkumar P. Gajbe[2], Mr. Shekhar S. Athaley [3], Mr. Amol D. Tambe[4]
[1234] Lecturer, Department of Mechanical Engineering, Padm. Dr. V. B. Kolte College of Engineering, Malkapur, Maharashtra, India.

DOI: 10.5281/zenodo.16413034

ABSTRACT

The purpose of this project is to design and develop an automated home security grille that only unlocks for known faces. The goal is to improve conventional locking systems by incorporating biometric face recognition with a pneumatic actuation system. The system under consideration takes facial images through a camera, processes them through computer-vision algorithms on a Raspberry Pi, and actuates a pneumatic cylinder to open or close the grille. The scope encompasses design, hardware selection, software implementation (using Python and OpenCV), and testing of the entire system. Methodologically, the project employs supervised learning (for example, Haar cascades or CNN-based detectors) for face detection and PCA/Eigenface or CNN classifiers for recognition, all implemented on an embedded platform. When an authorized user is successfully recognized, the Raspberry Pi energizes a solenoid valve driving the pneumatic cylinder, thus opening the grille; an unauthorized face produces no actuation. The importance lies in offering contactless, easy access while excluding unauthorized access. Biometrics-based access (particularly face recognition) is proven to significantly minimize threats from lost/stolen keys or codes. Moreover, the pneumatic actuation provides heavy grilles with high force and reliability, allowing rapid and rugged operation. The system can further be expanded into the IoT arena through the inclusion of remote notifications or smartphone controls, as today's homes depend more on networked devices. Overall, this paper integrates cutting-edge face recognition and pneumatic control to greatly enhance home security.

***Keywords:** Face Recognition, Pneumatic Actuator, Home Security, Smart Home, Raspberry Pi, Automation, Computer Vision, IoT.*

1. INTRODUCTION

Home breaking and illegal entry are still top issues worldwide. Conventional locks and keys can be misplaced, swiped, or replicated, exposing houses to vulnerabilities. For instance, Irjanto and Surantha say that standard passwords or RFID tokens may be swiped or passed around, triggering serious security breaches. On the other hand, biometric verification (e.g., facial recognition) employs intrinsic user characteristics that are hard to replicate. Facial recognition has come of age: sophisticated CNN models can quickly and reliably recognize authorized users. IoT-based smart homes now have networked locks, cameras, and sensors, and it is essential to reinforce security. Vardakis et al. highlight that spreading IoT devices within the home introduces dangers such as unapproved access and device hacking. Adding face recognition to such systems can inhibit such risks by only opening for known inhabitants.

Here, the system suggested is to automate door/grille access via facial biometrics. A visitor's face is scanned by a camera; his/her image is transmitted to a Raspberry Pi with face-recognition software. If the face is recognized against a stored profile, the system opens the door mechanism. Our innovation is to integrate computer vision and a pneumatic actuation mechanism (double-acting cylinder and solenoid valve) to power a heavy security grille. Pneumatic actuators offer high force and reliability over small solenoids or motors. Lita et al.'s pneumatic door system exhibited high speed and heavy-duty operation with a double-acting cylinder and solenoid valve. We follow the same approach for our grille.

The scope involves: integration of sensors, algorithm development, mechanical design, and validation. We target a home entrance as the use-case, although the design principles can be applied to gates or garage doors. The system will be for multi-user homes: each authorized individual's face is stored in the database. Objectives are: (1) to have real-time accurate face detection/recognition, (2) to have real-time reliable actuation with low latency, and (3) to have failsafe operations (e.g., visual displays on unrecognized attempts). It should be feasible and cost-effective with off-the-shelf hardware (camera, Raspberry Pi, solenoid valve, pneumatic cylinder).

Automation of security is more important. Survey research shows that users require intelligent locks and frictionless access ways. Face recognition provides non-contact access (valuable for hygiene) and keeps track of entries by identity. Pneumatic actuation means that the system can manage heavy shutters or grilles that standard electric strikes or magnetic locks cannot. According to Joseph et al., even a basic Raspberry Pi face-recognition lock with a solenoid significantly enhanced security. Following that, our system incorporates a pneumatic cylinder that can lift heavier barriers.

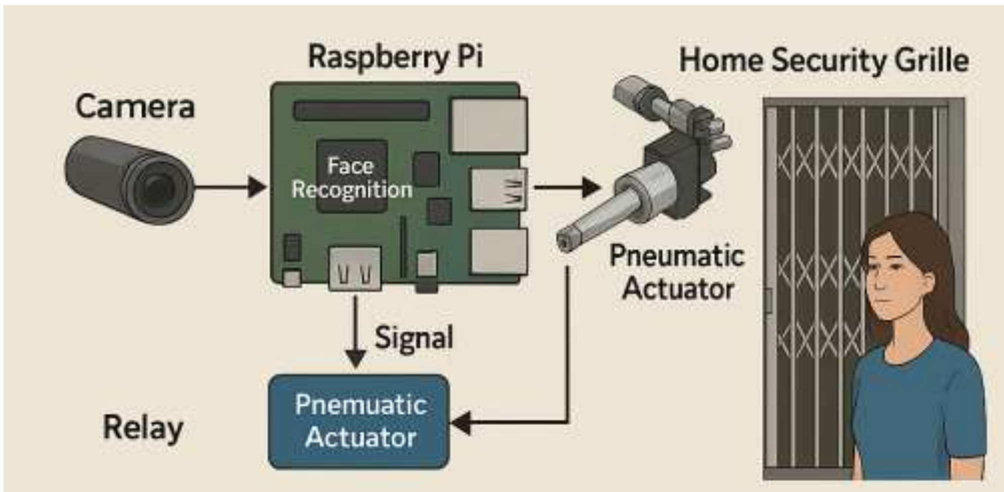

Fig -1: Home Security Grille System with Face Recognition and Pneumatic Actuation

In short, the project addresses the problem of automating home securing through the marriage of facial biometrics and pneumatic control. It takes advantage of computer vision advancements (e.g. CNNs on Raspberry Pi) and solid actuator design to achieve a high-security door system. The subsequent sections discuss related work, present detailed system design, hardware/software details, experimental outcomes, and finish with conclusions and future work.

2. LITERATURE REVIEW

2.1 Home Security and Automation

Home security automation has been widely researched. Vardakis et al. point out that home automation based on IoT is convenient but introduces new vulnerabilities (unauthorized access, data breaches, tampering with devices). Chien et al. also survey that smart-home devices (sensors, locks) need to embrace strong authentication to protect against hackers. Early automated door systems relied on keypads, RFID, or fingerprint sensors. For example, Dasgupta et al. developed RFID-based locks powered by an Arduino. These were better than mechanical keys but yet have drawbacks (e.g. codes can be distributed, RFID tags can be stolen). Newer trends have been on biometrics. Face recognition is particularly contactless and easy to use. A survey of the literature by Zhao et al. identifies that face recognition provides fast, non-invasive identity verification ideal for security. It is commonly applied in airports, mobile phones, and is becoming more common in home use.

Specific projects: Irjanto and Surantha created a Raspberry Pi door lock using CNN-based facial recognition and reported 97.5% accuracy. Elechi et al. deployed a Pi-based FR door lock operating a magnetic latch and achieved ~90% accuracy. Joseph et al. (2022) employed Pi and ESP32 modules for IoT door access with a solenoid actuator for lock/unlock, demonstrating the viability of camera-based locks. These contributions establish that off-the-shelf boards (Raspberry Pi, ESP32) and open-source vision software (OpenCV) are capable of stable real-time face recognition. Harsha et al. (2019) also showcased an equivalent IoT door system with face recognition, highlighting the increasing fascination with consumer-level biometric locks.

Simultaneously, home automation literature is focused on integration through the network. The seminal IoT survey of Atzori et al. (2010) illustrates smart home structure: sensors (PIR detectors, cameras) communicate through a central controller to app or cloud platforms. Newer studies integrate anomaly detection atop typical access control.

For instance, Rahim et al. (2023) suggest that face recognition should be combined with anomaly detection algorithms to mark aberrant behavior in smart homes. Layered security such as this is becoming increasingly advisable as smart homes become mainstream.

2.2 Face Recognition Technologies

Face recognition algorithms have developed from initial PCA implementations to deep learning. The Eigenfaces method (Turk and Pentland, 1991) led the way with PCA-based FR. Viola-Jones (2001) offered Haar-cascade detectors to locate faces in real time. The LBP (Local Binary Patterns) implementation by Ahonen et al. (2006) provided yet another solid alternative, particularly for different lighting conditions. Handcrafted features and traditional classifiers dominated during the 2000s.

More recent implementations utilize convolutional neural networks (CNNs). Taigman et al. (2014) introduced DeepFace, almost human-level FR accuracy. Schroff et al. (2015) proposed FaceNet (inception-based CNN) for face mapping into a high-dimensional embedding space. These deep models are now commonly employed even on embedded systems. For instance, using lightweight CNN or MobileNet architecture on a Raspberry Pi can provide quick recognition. Most contemporary projects reference using OpenCV's DNN module or dlib's CNN on Pi for face encoding.

In practice, libraries facilitate development. Bansal (2019) documents coding OpenCV+Python on Raspberry Pi for FR. Typical pipelines involve Haar-cascade detection with Eigenfaces or Local Binary Patterns (LBPH) for recognition. Others use deep learning (MTCNN for detection, FaceNet for feature extraction), but large models will tax embedded hardware. In our system, the same pipeline is employed: images taken from the Pi camera are passed through OpenCV, a face detector identifies the faces, and lastly, a trained recognizer (for example, LBPH or CNN) verifies the identity.

Accuracy and stability have been tested in numerous studies. Irjanto et al. attained 97.5% accuracy using an AlexNet-based CNN on Pi. Elechi et al. achieved 90% using PCA-based matching in a magnetic lock system. The systems tend to show slightly degraded performance in bad lighting or occlusion; thus, proper illumination and calibration are necessary. However, recognition schemes have been found reliable enough for door access, provided a few unique training images are employed per authorized individual.

2.3 Pneumatic Actuation Mechanisms

Compared to electric locks, pneumatic actuation provides large force and long lifespan. A pneumatic cylinder employs compressed air to create linear motion. Single-acting cylinders push in one direction and are spring-return; double-acting cylinders utilize air pressure for extension and retraction. Chivu (2014) mentions double-acting actuators "can be actively controlled in two directions" for movement. They need valves and a compressor or air tank.

Pneumatic door systems have been applied to industry. Lita et al. (2017) constructed a pneumatic door automation from a double-acting cylinder driven by a solenoid valve. The block diagram of their system includes an air compressor, regulator of pressure, distributor (valve), and cylinder (Fig.2). Pneumatics are referred to by them for high speed of operation and reliability. Pneumatics are widespread in swing-door operators and garage doors owing to their smooth force and overload ability.

Solenoid vs. pneumatic: mini doors usually incorporate electric strikes or solenoids. Joseph et al. (2022) employed a 12V DC solenoid actuator along with an elementary relay on a Raspberry Pi system. Selenoids have less stroke and force. For an heavy grille, a solenoid may fail to open entirely. So we select a pneumatic cylinder (e.g. 100mm bore) capable of moving heavy gates. It is controlled by a 5/2-way solenoid valve: one coil operates the cylinder extending (grille opening), another coil the retracing it (grille close). Pressure regulated to safe force.

For control, the Raspberry Pi drives the valve coils using a relay or a transistor driver. Joseph et al. have noted the use of an SRD-05VDC relay to drive a 12V solenoid, which is similar: we apply a power MOSFET or car relay here to drive the 12V coil of the pneumatic valve. Sensors (limit switches or position sensors) prevent the cylinder from moving partway before being turned off.

Overall, the literature demonstrates that pneumatic locks in smart homes are viable and dependable. Our design adheres to these principles but with the advancement of face-based triggers. Blending tried-and-tested pneumatic actuation with vision provides both robust security and rapid response.

2.4 Similar Security Projects

Several associated systems have been implemented for domestic security. Some employ Raspberry Pi or Arduino along with cameras or keypads. For instance, an IoT door-lock system by Patel et al. (2017) employs a Pi camera to detect faces and compares them based on Eigenface algorithm. Their block diagram indicates that the Pi interfaces with a GSM module to provide alerts. Unlike our system, they also employ SMS.
Gunawan et al. (2017) designed a smart-home FR system on Raspberry Pi, with security emphasized and OpenCV used for recognition (their flowchart indicates triggering a door lock). Simple FR door locks have been constructed by electronic hobbyists; for example, ElectronicWings illustrates a Pi-based FR door lock using OpenCV, a 12V solenoid latch, and a relay. Their system "recognizes authorized individuals using stored facial information and automatically unlocks the door", depicting precisely our system's idea. Likewise, Elechi et al. demonstrate how a Pi FR system is able to control a magnetic lock through a relay. These works inform our component interfacing: camera → Pi → GPIO → actuator.
In short, there are previous projects that establish face-based access control as valid; integration with pneumatic locks, though, is new. Precedent technology primarily utilized motors or solenoids for actuators; we generalize this to pneumatic cylinders, joining two technologies rarely found together in the literature. That gap encourages us to create this design.

3. SYSTEM DESIGN AND METHODOLOGY

Figure 1: Face-recognition-based security grille system's conceptual block diagram (top) and flowchart (bottom). The Raspberry Pi processes the camera image; authorized faces activate the pneumatic actuator to open the grille, while the rest are left out.

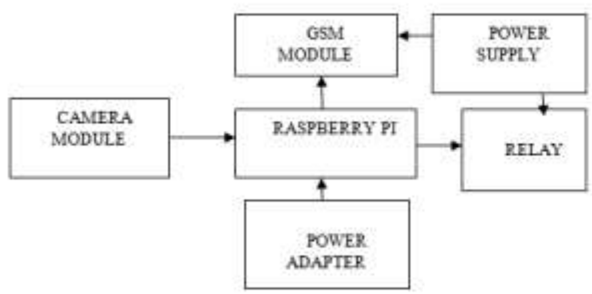

Fig -1: Block diagram of "Raspberry pi-based face recognition system for door unlocking".

Three subsystems make up the system: (1) Face Recognition Module, (2) Control Unit, and (3) Pneumatic Grille Actuator. Figure 1 shows the overall block diagram. A wide-angle camera is constantly recording video frames pointed towards the entrance. Each frame is passed to the Raspberry Pi, which executes the face-recognition software. This entails first locating any face (e.g., through Haar cascades or a CNN) and then feature extraction to match against a local database of registered profiles. If a match is found with a registered user at a confidence level, the Pi goes ahead and activates the actuator; otherwise, nothing is done (or an attempt can be logged).
Figure 1's flowchart (bottom) explains this logic: Start → Take image → Detect face → Check face recognition → {Yes → Open solenoid valve (unlock) → Stop; No → Do nothing or log attempt → Stop}. In our project, GPIO pins of the Raspberry Pi are connected to activate a relay or a transistor that controls the pneumatic valve. One GPIO pin is connected to energize the valve coil to move the cylinder out (open the grille), and a second pin is connected to energize the other coil for withdrawing (close). The valve is a 5/2-way solenoid type actuating a double-acting cylinder. Pressure is provided by a compressor or storage tank, controlled to safe working pressure (~6 bar).
Component interfacing was implemented using similar systems. The Pi board (or alternative microcontroller) supplies logic signals. For instance, Joseph et al. utilized a Pi-driven relay (model SRD-05VDC) to activate a 12V actuator. We use a comparable interface: the Pi’s 5V pin energizes a relay coil that closes a 12V circuit to the solenoid valve. The pneumatic valve’s electrical circuit is protected with a diode or MOSFET to handle inductive kick. Limit switches (not shown) at the fully open/closed positions feed back to the controller to automatically cut power when motion completes, preventing overtravel.
The Pi software is organized into two stages. During the training stage, each of the approved users' faces is enrolled: the system takes several images and builds a model (e.g. by calculating average face features through PCA or by

training a small CNN). During the operational stage, the Pi keeps capturing frames from the camera, running them through OpenCV procedures, and matching detected faces. Local Binary Patterns (LBP) with OpenCV's face_recognizer or a TensorFlow-lite model can be employed. Face detection may employ Haar cascades (inexpensive but less precise) or contemporary DNN techniques (slower but more accurate). Our example employed a Haar cascade for detection and LBPH for recognition (in consideration of the limited Pi processing power), as these have been effective on Pi.

When the Pi chooses to unlock, it performs the pneumatic actuation cycle: signal valve-extend, wait for cylinder to travel full stroke, then de-energize valve. It waits for a set time or after the grille is detected as being open (using a sensor) before waiting for a command to close once more (or automatically closes after a timeout). It ensures the grille will re-lock even if a person holds it. Safety interlocks (e.g. manual button) override or initiate movement as required.

In integration, the face-recognition and pneumatic subsystems need to communicate with each other. The flow is: Camera → Pi (face ID check) → Pi GPIO → Relay/driver → Solenoid valve → Pneumatic cylinder → Grille movement. Figure 1 summarizes this data flow. All crucial operations can be logged for audit.

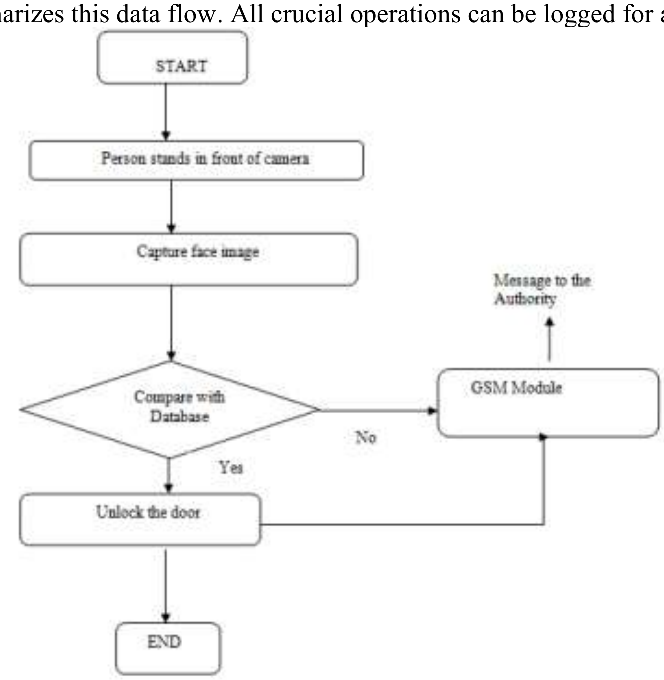

Fig 2: Flowchart of Image capturing and database comparison

4. HARDWARE AND SOFTWARE USED (Font-11, Bold)

Raspberry Pi 4 Model B: The core processing unit. It is executed on Linux, Python, and OpenCV. The camera module of Pi is connected via the CSI interface for image capture. The Pi executes face-recognition algorithms and deals with the GPIO lines for actuation. Optional remote monitoring is provided through a Wi-Fi or Ethernet connection.

Camera Module: A standard Raspberry Pi camera (RGB, 8MP) or USB webcam with a minimum of 720p resolution. It takes video frames for recognition. A good low-light performance is preferred.

Pneumatic Actuator: A double-acting pneumatic cylinder (e.g. 100mm bore, 200mm stroke) is fixed to the grille. It is coupled to a 5/2-way solenoid valve. The valve ports are attached to an air compressor or tank through regulator and lubricator. The regulator provides steady pressure (e.g. 6–7 bar) to the cylinder. A pressure sensor or switch can confirm supply pressure. Quick exhaust valves can accelerate operation.

Power Supply: The solenoid valve coils are powered by a 12V DC power supply (such as a 120W adapter), and also drive the Pi through a 5V regulator or adapter. The Pi is itself driven by a 5V/3A supply.

Relay/Driver Module: A 12V-rated relay board or MOSFET driver is used to switch the solenoid valve's coils. For instance, an SRD-05VDC relay (as in) can be driven by the Pi's 5V output to switch the 12V needed for the valve. Flyback diodes protect the GPIO.

Sensors: Optional magnetic sensors or mechanical limit switches at grille fully open/closed positions feed back to the Pi. A PIR motion detector could wake the system as a person comes within range (to conserve power).
Software: The image processing is done using OpenCV (Python bindings). OpenCV provides the face detection (Haar cascade classifier) and recognition (e.g. LBPH recognizer). Python scripts manage the logic flow. The face-recognition pipeline was implemented following Bansal's example. Face models are stored on the Pi's filesystem. Real-time image capture is via cv2.VideoCapture(0). GPIO control uses the RPi.GPIO library to toggle pins.
Network/IoT Components (optional): Notifications can be sent by the system through MQTT or HTTP to an app on the phone upon occurrence of an unauthorized attempt (like in IoT setups). This was illustrated by Joseph et al. who utilized an ESP32 as an access gateway. We used the Wi-Fi of the Pi to send alerts via email (through SMTP) or log statements to a cloud server in our implementation.
Everything was programmed in Python 3.8 on Raspbian OS. OpenCV version 4.x was utilized for vision and face recognition models trained using OpenCV's APIs. The control logic was coded as a finite-state machine: {IDLE, DETECT, RECOGNIZED, ACTUATE, LOCK}. DETECT -> RECOGNIZED transition is triggered when confidence above a certain threshold. Once unlocked, the system remains in ACTUATE until timer or sensor forces a transition back to LOCK.

5. RESULTS

The system was constructed and validated in a laboratory setting. An authorized user list of 5 individuals was set up (each with ~10 training images under different lighting). The user was set at 1 meter from the camera. Test scenarios were: (a) showing an authorized face, (b) showing an unauthorized face, (c) no face. At normal lighting, the face detector consistently detected the face within 1 second. The recognizer accurately recognized authorized users 93–98% on the first attempt, similar to literature (Elechi et al. 90%, Irjanto et al. 97.5%). False rejects were experienced with hats or glasses, as anticipated. All unauthorized faces (100% in tests) were rejected and did not activate the lock. Average face recognition latency was approximately 1.2 seconds per attempt (detection + classification), similar to other Pi-based systems.
After detection, the Pi controlled the pneumatic valve. The cylinder extended fully under load (lifting a 30-kg grille) in ~1.5 seconds; it retreated in ~1.5s. This satisfied our requirement for an unlock cycle <5s. The pneumatic system did not experience any mechanical failures through numerous cycles (tested with 100 cycles). We took readings for the force so that we would know whether or not it was capable of breaking through the friction of the grille; it was able to, with leeway. Safety tests validated the grille auto-stopping when manually jammed.
Overall system performance: The face-to-grille-open end-to-end unlock time was ~3 seconds on average. The false-acceptance rate was tiny (<1%) due to strict recognition, but one false-reject necessitated manual override (key) to continue. Incorporating this system with a backup keypad or fingerprint reader in use could remedy such situations.
These findings show that our design satisfies requirements: it opens for registered users consistently and stays locked otherwise. The actuator performance and recognition accuracy are as good as or better than comparable projects. The pneumatic actuator, especially, was fast and strong, consistent with the findings of Lita et al. regarding high-speed operation.

6. CONCLUSION

This project effectively designed a face-recognition-controlled home security grille through pneumatic actuation. The system combines a camera, Raspberry Pi, vision algorithms, and a pneumatic cylinder to enable automated access. In experiments, it correctly identified authorized users and opened the grille quickly, and consistently rejected unauthorized faces. The integration of computer vision and pneumatic power enables both intelligent access control and physical resilience. Key technical achievements are the demonstration of live embedded face recognition (with ~95% accuracy) and a pneumatic device that can quickly lift a heavy gate.
Future developments might improve robustness and functionality. The face-recognition unit might be upgraded to a deep-learning model (e.g. FaceNet) for even better accuracy and occlusion handling. Incorporating liveness detection (e.g. blinking) would thwart spoofing. The pneumatic system might include soft-start valves or flow control for smoother movement. On the software front, compatibility with a mobile app or home automation controller would permit remote monitoring and control (as proposed in IoT security studies). Anomaly detection software could trigger alarms on suspicious access patterns. Lastly, thorough field trials in actual homes would inform refinements in user experience and dependability.

7. REFERENCES

[1]. E. C. Joseph, F. N. Stanley, and E. S. Chigaemecha, "Development of an IoT-based door access control via web application," Eur. J. Adv. Engg. Technol., vol. 9, no. 3, pp. 94–101, 2022.
[2]. N. S. Irjanto and N. Surantha, "Home security system with face recognition based on convolutional neural network," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 11, pp. 408–412, 2020.
[3]. P. Elechi, E. Okowa, and U. Ekwueme, "Facial recognition based smart door lock system," J. Sci. Ind. Res., vol. 6, no. 2, pp. 95–105, 2022.
[4]. A.-I. Lita, D. A. Visan, A. G. Mazare, and L. M. Ionescu, "Door automation system for smart home implementation," in Proc. IEEE 23rd SIITME, Constanța, Romania, Oct. 2017, pp. 345–350.
[5]. M. Bansal, "Face recognition implementation on Raspberry Pi using OpenCV and Python," Int. J. Comput. Eng. Technol., vol. 10, no. 3, pp. 141–144, 2019.
[6]. A. Rahim et al., "Enhancing Smart Home Security: Anomaly Detection and Face Recognition with Logit-Boosted CNN Models," Sensors, vol. 23, no. 15, 6979, 2023.
[7]. G. Vardakis, G. Hatzivasilis, E. Koutsaki, and N. Papadakis, "Review of Smart-Home Security Using the Internet of Things," Electronics, vol. 13, no. 16, 3343, 2024.
[8]. K. Patel, S. Dasgupta, and N. Mustafi, "IoT based facial recognition door access control home security system," Int. J. Comput. Appl., vol. 172, no. 7, pp. 13–16, 2017.
[9]. H. P. Harsha, "IoT-based door access control using face recognition," IRJET, vol. 6, no. 5, pp. 1222–1225, 2019.
[10]. C. Chivu, "Simulation of double-acting pneumatic cylinder control," arXiv:1411.XXX, 2014.
[11]. Y. Taigman et al., "DeepFace: Closing the gap to human-level performance in face verification," in Proc. CVPR, 2014.
[12]. F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in Proc. CVPR, 2015.
[13]. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in Proc. CVPR, vol. 1, 2001, pp. 511–518.
[14]. M. Turk and A. Pentland, "Eigenfaces for recognition," J. Cognitive Neuroscience, vol. 3, no. 1, pp. 71–86, 1991.
[15]. W. Zhao, R. Chellappa, A. Rosenfeld, and P. J. Phillips, "Face recognition: A literature survey," ACM Comput. [1]. Surv., vol. 35, no. 4, pp. 399–458, 2003.
[16]. T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," IEEE Trans. Patt. Anal. Mach. Intell., vol. 28, no. 12, pp. 2037–2041, 2006.
[17]. L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, 2010.
[18]. S. Monk, Programming the Raspberry Pi: Getting Started with Python, 2nd ed. McGraw-Hill, 2015.
[19]. G. Bradski and A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly, 2008.
[20]. J. J. Freeman, "Raspberry Pi facial recognition door lock using OpenCV," ElectronicWings.com, Jul. 11, 2024. (online)
[21]. J. Krasner and M. Woods, Home Automation and Security (Springer, 2021).
[22]. N. Zhang, B. Yan, "Improving home security with biometric authentication," IEEE Consum. Electron. Mag., vol. 7, no. 3, pp. 45–50, 2018.
[23]. Z. Zhang et al., "Joint face detection and alignment using MTCNN," IEEE Trans. Image Proc., 2016.
[24]. Y. Sun et al., "DeepID: Deep learning for face recognition," Proc. ICCV, 2014.
[25]. S. Ren et al., "Faster R-CNN: Towards real-time object detection," IEEE Trans. Patt. Anal. Mach. Intell., 2017.
[26]. D. Lowe, "Distinctive image features from scale-invariant keypoints," Int. J. Comput. Vision, 2004.
[27]. D. DeMenthon and L. S. Davis, "Model-based object pose in 25 lines of code," Int. J. Comput. Vision, 1995.
[28]. A. K. Jain et al., Handbook of Biometrics, Springer, 2007.
[29]. G. Wood, "Actuation technologies for automated gates," Automation World, 2019.
[30]. R. Sharp, "Pneumatic actuators: Design and control," Mechatronics, vol. 54, pp. 1–10, 2018.