Design for Visitor Authentication Based on Face Recognition Technology Using CCTV

HYUNG-JIN MUN and MIN-HYE LEE

1Department of Information and Communication Engineering, Sungkyul University, Anyang, Gyeonggi 14097, South Korea
2Center for General Education, Wonkwang University, Iksan, Jeollabuk 54538, South Korea

Corresponding author: Min-Hye Lee (lmh3322@wku.ac.kr)

This work was supported by Wonkwang University in 2020.

ABSTRACT Recently, image recognition technology using deep learning has improved significantly, and security systems and home services that use biometric information such as fingerprints, iris scans, and face recognition are attracting attention. In particular, user authentication methods that utilize face recognition have been studied at length. This study presents a visitor authentication technology that uses CCTV with a Jetson Nano and webcam. In the preprocessing phase for face recognition, face data with 7 features that can be identified as a person are collected using CCTV. The collected dataset goes through the annotation process to classify the data, and facial features are detected using deep learning. If there are four or more detected features, the image data is determined to be a person, and the visitor’s face is matched with stored user data in detail using 81 feature vectors. Additionally, the security of the access control system was enhanced by implementing logging functions such as recording the face of the visitor, the number of visitors, and the time of the visit. This paper implements a visitor authentication system using a Jetson Nano and evaluates performance by analyzing the accuracy and detection speed of the system. The tiny-YOLOv3 in the Jetson Nano was effective in real-time verification for the real-time face authentication system with an average detection speed of 6.5 FPS and 86.3% accuracy. Through this study, we designed a system based on deep learning technology that recognizes and authenticates the face of a user during the visitor access process and controls user access.

INDEX TERMS Face recognition, deep learning, visitor authentication, security, YOLO, Jetson nano.

I. INTRODUCTION

As the need for security spreads around the world, many smart home services using CCTV (Closed-circuit Television) are being installed and operated [1]. According to a report by the global statistics site Statista (Fig. 1), the global smart home security market is expected to increase from $9,903 million in 2017 to $35,619 million in 2024, with its compound annual growth rate (CAGR) increasing by 20.1%.

CCTV can record the situation outside a building or in the absence of people and transmit this information to the user. So, it is often used for the purpose of crime prevention, facility safety, and fire detection. Currently, there has been researches about Fire Detection or weapon detection using CCTV applying deep learning technique such as CNN [2], [3], [4].

In particular, CCTV in smart home services has the additional functions of automatically applying user access controls, automatically searching the surroundings at a determined time, tracking a target, and increasing image resolution. As artificial intelligence (AI) has developed, systems that have introduced intelligent object recognition technology have been continuously studied. However, most commercial CCTV systems are focused on simple functions, so it is...
difficult to confirm whether a targeted object is a person or not, and there is a limit to responding to the situation even when a person is recognized.

Therefore, we propose a system that can identify visitors by applying AI technology to CCTV and provide services according to the situation. The proposed system uses You Only Look Once (YOLO) [5] and OpenCV with the image input from the CCTV camera in front of the door to determine whether or not a human face is recognized, whether or not a person is an intruder, and whether or not that person is trying to steal the door code by peeking over the registered user’s shoulder.

This system can provide an alarm notification to users by recognizing situations when there is an outsider beside or behind the registered person (such as a family member) who may be attempting to steal the door code. Recently, there have been a number of incidents of outsiders entering an apartment building along with a woman who is alone at night. In such a case, photographs and logs about the access situation are transmitted to the building manager to prevent intruder access and to be used as evidence in the case of an incident response.

The following are the requirements that the proposal system must satisfy:

1. It needs to identify an intruder or companion: If the companion is a family member, the alarm notification is displayed as a family member. If the companion is not a family member, a long password is required to prevent peeking.
2. It needs to check if someone is peeking at the password: The system detects the companion’s eyes and determines if he or she is peeking over the registered user’s shoulder. If there is a passer by, the alarm goes off and asks for a long password. When entering a short password, a notification message is printed informing the user that, “the password is too short.”
3. It needs to recognize the Registered person’s face: Family members register photos in advance. When a visitor is authenticated based on OpenCV, the door is opened through multi-factor authentication when entering a password at the front door.
4. Various services such as remote door opening and intruder detection needs to be available: If necessary, the system include the ability to be notified of the family’s return home on the system manager’s smartphone, check access information in real time, and analyze access patterns or provide additional services in the case of long-term absence.

In this study, we utilize a Jetson Nano and YOLO, which is an object detection deep learning model that can recognize visitors in real time, in order to develop a system that can check for intruders, authorized visitors, and peeking over a person’s shoulder.

There are various types of microprocessors (microcontrollers) that connect various sensors and are portable such as Arduino, Raspberry Pi, Panda, Jetson Nano, and so on. In order to determine the suitability of hardware for building and utilizing the proposed model, two types of microprocessors that are suitable for deep learning were analyzed in the experiment. First, the portable, inexpensive, and commonly used Raspberry Pi was used, and secondly, the recently released Jetson Nano with a built-in GPU was used.

We implemented a stand-alone CCTV mechanism by using Webcam and Jetson Nano. This proposed system is used to detect visitors’ faces and to identify whether faces are registered member or not.

The following are the contributions of this paper.

1. The system was developed at a low cost since it is stand alone and not server-based.
2. Experimenting Raspberry Pi and Jetson Nano in same environment, showed the different accuracy.
3. The purpose of visitor authentication was achieved by face detection and face recognition.
4. The system was proven safe from various attacks, provided convenience, and allowed visitor authentication.

The structure of the paper is as follows. Chapter 2 introduces related research including image processing technology and user authentication technology. Chapter 3 designs a visitor face authentication system using CCTV and presents the components necessary for deep learning, the type of learning required for the system. Chapter 4 introduces and presents the protocol of the proposed system. And in Chapter 5, the user face authentication technique and the proposed system are evaluated and analyzed, and conclusions are drawn.

II. RELATED WORKS
A. USER AUTHENTICATION
Identification and authentication are required to ensure that users have access and permissions to certain information. The user authentication approach is accomplished in four ways [6]. Authentication is divided into knowledge-based authentication (what you know, i.e. a password), ownership-based authentication (what you have, i.e. a credit card or mobile phone), and object characteristic-based authentication (what you are, i.e. biometric data). The fourth approach to user authentication includes a combination of the other three methods and is known as multi-factor authentication.
A simple and convenient user authentication method that is often used is ID/password-based authentication. Password authentication techniques, such as Fig. 2, pass the hash value of the password [7] to the server when the user enters their ID and password. It then verifies the hash value of the input password against the hash value of the original password provided at the time of registration.

The safety of password-based authentication depends on the management of the database (DB) in which passwords are stored. There have been numerous cases in which attackers linked to insiders have leaked users’ passwords by using a member information table containing member information. In general, the password registered by the user is not stored in the database. This is to protect the password from leakage in the member information table. But the resulting value of a hash function, which is a one-way function, is stored in the member information table instead of the password. Vulnerabilities, such as crashes in the MD5 hash function, which generates 128-bit hash values, have led to the MD5 hash function being widely replaced by stronger hash functions such as SHA-256 and SHA-512 [9]. However, whenever a user enters a password, it can still be recorded by someone peeking over the shoulder of the user or by recording the input when entering a password through Google Glass. In addition, social engineering attacks based on user information or password brute force attacks are also possible. If the member information table where the password is stored is leaked, the user’s password can be leaked through a Rainbow Table attack by using the hash value generated in advance [10], [11].

Ownership-based user authentication is an authentication technique that identifies a user’s possessions. Possessions used for authentication can include identification cards, encryption keys, smartphones, credit cards, and OTP (one-time passwords). After password-based authentication, additional authentication techniques such as SMS authentication or OTP authentication are required.

Object characteristic-based authentication is an authentication method which authenticates users based on user characteristics. Biometrics is a representative object characteristic-based authentication method. Biometrics distinguish physical and behavioral features. Physical features include face recognition using face shapes and thermal images of each person, iris recognition, vein recognition, fingerprint recognition, and other methods including retina and hand shape. Behavioral features include speech recognition, gait recognition, and signature recognition. Recently, companies have begun using behavioral features such as gait, signature, and voice recognition to enhance biometric accuracy [12], [13], [14]. Face recognition is more convenient than fingerprint, iris, voice, and vein recognition. Iris recognition is more accurate than other biometrics, and the installation cost is less than a speech recognition system. Recently, a permanent and low-cost fingerprint recognition method or Face ID method, based on the user’s face, has been implemented and used almost universally to identify the owner of the phone and allow smartphone use and access.

Multi-factor authentication is an effective authentication technique that allows users to further enhance security by authenticating with ID/Password and then using another additional authentication method. Additional methods of multi-factor authentication include ownership-based authentication or object characteristic-based authentication. In particular, smartphones are being used as a means of authentication with such features as fingerprint authentication, face authentication, OTP, or random number input through SMS.

B. IMAGE PROCESSING AND ARTIFICIAL INTELLIGENCE TECHNOLOGY

1) OBJECT DETECTION TECHNOLOGY

The traditional object detection methods such as Haar-like features, scale invariant features transform (SIFT), and histograms of oriented gradients (HOG) require the trouble of manually extracting and adjusting features of an image according to the detection target [15]. Object detection technology is growing rapidly due to the proposal of a deep learning method that can collectively process data at once from feature extraction to classification.

Object detection research using deep learning started with the proposal of a Convolutional Neural Network (CNN) model, followed by Regions with CNN features (R-CNN) [16], fast R-CNN [17], and faster R-CNN [18]. An R-CNN-based deep learning model is a model that detects objects by applying four techniques: region proposal creation, feature extraction, classification, and linear regression. However, the R-CNN-based model has a problem in that each process is executed separately, which requires more
computing power. But, the faster R-CNN model introduced another region proposal network (RPN) to solve the problem of inefficiency in learning and running time.

2) OBJECT DETECTION TECHNOLOGY IN A REAL-TIME ENVIRONMENT

The R-CNN-based deep learning model extracts and classifies features from the proposed region. Since the detection and recognition processes have to be individually trained, the problem exists that the computation speed is slow and optimization is difficult. YOLO is a model that integrates object detection and recognition into one system to improve the shortcomings of the R-CNN model. The input image is divided into a grid, and each grid cell calculates whether the object exists within it and the confidence score for the object’s existence. Then, the reliability of the object detection is calculated and the object in the area is recognized. Since this model is faster than the R-CNN model, it can be applied to the security and authentication system used in real-time environments to detect objects quickly and easily. Beginning with YOLOv1, the YOLO model was extended to YOLOv4 to compensate for the low object detection accuracy compared to the fast detection speed and to detect more classes. As the version number increased and performance improved, it was necessary to reduce the computational complexity by simplifying the network layer for real-time detection in a low-spec system environment. Accordingly, Tiny-YOLO was devised as a model for low-spec real-time object detection.

YOLOv3 uses Darknet-53 as the backbone network. By applying the Feature Pyramid Network (FPN) concept, each object is detected by feature maps at three scales created in the process of passing the image through each neural network. In order to detect small objects, an object is detected by increasing the size of the feature map by upsampling it. Three Anchor boxes will be generated for each scale, and a total of nine Anchor boxes will be selected. Fig. 3 shows the structure of the YOLOv3 model [19], [20], [21]. Tiny-YOLOv3 is a model with the YOLOv3 structure that reduces computing resources and increases speed through the simplification of the Convolutional and Pooling processes. Tiny-YOLOv3 is less accurate than the YOLOv3 model, but it is faster and can be used in a limited detection environment [22], [23], [24].

In order to be applied to an actual industrial field, even if the accuracy of its performance is somewhat low, it is necessary to reduce the cost of processing the data with higher speeds and multiprocessing. Therefore, for real-time application in the proposed system, it is appropriate to use Tiny-YOLO, which has a high processing speed, even if the accuracy is somewhat low [25].

3) SECURITY SERVICE WITH FACE RECOGNITION TECHNOLOGY APPLIED

Face recognition technology is a technology that scans and recognizes face shapes and thermograms with thermal infrared rays. After taking a picture of the face with a camera, either 68 or 81 key points on areas such as the eyes and eyebrows, nose, mouth, and chin can be analyzed to extract characteristic information. The extracted feature information is compared with the facial feature information stored in the database in order to recognize the face and confirm the identity of the user.

The video security market was able to detect objects with motion analysis and rule-based pattern analysis by early 2010, but false positives were frequent with an error percentage of 28.2%. But with the development of deep learning technology, a breakthrough improvement was made, reducing the error rate from 18.4% in 2012 to 3.57% in 2015. Additionally, the accuracy of face recognition increased by about 95%.

In general, CNN-based deep learning models are used to detect faces and apply them to face recognition systems. Included among these are face mask detection systems for preventing the spread of infectious diseases and restricting access [26], age estimation studies on face images, with various neural network algorithms applied, for use in intelligent security, financial transactions, and so on [27], criminal identification using MTCNN algorithm research [28], and face recognition in smart door lock systems [29]. All of these are being researched to apply to various intelligent security systems.

C. ACCESS CONTROL SYSTEM

An access control system, a type of physical security, is used in many places to block unauthorized users from entering, such as buildings or offices.

A representative example of this is shown in the method of controlling the entrance and exit of a car to a parking lot. A camera is used to recognize a car’s license plate number and then check whether it is registered in the system, or an RFID-based smart card system is used. This RFID-based contactless smart card is attached to the windshield of the car and used to determine whether car access is permitted. Convenience is guaranteed and for easy authentication, the same contactless smart card access is possible even inside the building when entering the office. In the case of contactless smart cards, a security algorithm is applied, so it is safe from illegal copying or hacking without the encryption algorithm, special key, and mutual authentication. Smart door locks are widely used to control access by entering passwords in apartments or houses. But with the development of biometric authentication technology, authentication with a user’s fingerprint, or iris recognition with a camera, the service for controlling access by confirming that the user is registered has become commonplace. Recently, various methods have also been developed to control access to prevent the spread of infectious diseases (Fig. 4).

Additional research is now being conducted to confirm the approach of a user with the PIR proximity sensor, or to recognize the user’s voice with a microphone, and identify the user with additional authentication methods such as questions and answers [30]. Face recognition technology using deep learning is being used for user authentication by recognizing
people in access management systems. In general, access control has previously been performed through security staff or visitor tags, but problems may arise due to the reduction of security staff or loss of visitor tags. Accordingly, face recognition technology is emerging as an alternative technology for effective access control. Recently, due to COVID-19, it has been partially applied for purposes such as creating access lists, wearing masks, and measuring body temperature. Object detection technology using such deep learning can also be applied to security management systems. In addition, when entering a number in front of the front door, face recognition is emerging as an alternative to the act of password peeking by outsiders. When entering a password, it can be used to develop a system that sounds an alarm by determining whether there is a person behind the user and whether that person is a companion.

### III. DESIGN FOR PROPOSED SYSTEM & DEEP LEARNING

#### A. SYSTEM STRUCTURE
The overall structure of the system proposed in this paper is shown in a block diagram as shown in Fig. 5. The composition of the system is largely divided into parts in charge of registration and processing. In detail, it consists of (1) deep learning model training, (2) visitor face detection, (3) face recognition and access verification, and (4) monitoring.

In the deep learning model training portion, a learning model is obtained by using Tiny-YOLOv3 for face features data sets that are labeled with 7 classes. In the visitor face detection portion, we create the face recognition security system. This includes a webcam to act as the CCTV, a servo motor to monitor the surroundings, and an infrared sensor for human body detection. Each of these is connected to the Jetson Nano board. First, a human face is detected from the real-time image received from the webcam using the learning model. When it is confirmed that a human face is present, the detailed features of the face are detected using the landmark algorithm in the recognition and security portion. The detected detailed features are compared with the family of face data stored in the database of the monitoring system, and security is canceled in case of a member, while an alarm notification is displayed to the user in case of an outsider. Lastly, log information such as the face and number of people in question is stored in the database through the wireless LAN. The results can be checked later.

1) DESCRIPTION OF EACH COMPONENT

1. **Deep learning model training component (derivation of a deep learning training model):** Desktop - Save images of family and acquaintances in a dataset in advance and learn them using the Tiny-YOLOv3 deep learning model.
2. **Video data collection component:** Collects the visitor’s video using the webcam CCTV.
3. **Control and processing components:** A database is built on the Jetson Nano board where the collected video images are stored and processed.
4. **Monitoring and post processing:** The processing result is transmitted to the manager’s smartphone and processed.
2) HARDWARE CONFIGURATION DIAGRAM
The operation of the entire system for face detection, recognition, and log storage uses NVIDIA’s Jetson Nano board as the main system. The Jetson Nano board is a development kit optimized for artificial intelligence learning with Ubuntu 18.04 on a Quad-core ARM A57 CPU @1.43 GHz, with a 128-core Maxwell GPU, and 4GB of 64-bit LPDDR4.

- NVIDIA’s Jetson Nano Board
- Logitech C270 HD Webcam
- HDMI LCD Touch Screen Monitor
- Servo motor
- Human body detection infrared sensor
- Logitech Speaker

Fig. 6 is an example of the hardware implementation form of the proposed system. Logitech’s C270 HD Webcam is connected to the Jetson Nano board to be used as the CCTV camera to scan for visitors, and an infrared sensor is added for visitor detection. The system begins operation when a visitor is detected through the infrared sensor. The image received from the CCTV installed in front of the door is detected using a deep learning model that has been pre-trained. By using the face detection system, key points that can identify a visitor’s face are searched and compared with the face data stored in the system database to determine whether the visitor is a member or not. MariaDB is installed in the system to store video images and access information collected from CCTV. An Internet connection is required for notification of visitors and remote monitoring of users. A wireless LAN card has been added to the Jetson Nano board to connect to the smart home. Additional authentication is required to reduce the false acceptance rate (FAR) of the visitor’s face recognition. Finally, the touch screen outputs a message to the user on the screen, and if necessary, a keypad that can be input on the screen is displayed, where the password can be input with a touch.

B. DEEP LEARNING-BASED TRAINING MODEL
1) DATASET COLLECTION AND PREPROCESSING
To create a dataset, we had Google crawl over 500 face images with various features on the web. Considering that the place where the system is installed is a specific environment, 500 additional images of people entering and exiting the building were collected. It obtained face images which have various looks (e.g., wearing eye glasses, mask, cap) in addition to gender, age, and lateral angle of the face in order to decrease false positive ratio in face detection and to increase accuracy.
For smooth learning in the Jetson Nano, the image was resized to a standard size of 400 × 320. In addition, the images collected in consideration of the influence of the environment were augmented through various angles of rotation, inversion, and brightness adjustment. Out of a total of 2,000 images, 1,400 were used as a training dataset, 400 were classified as a validation dataset, and 200 were classified as a test dataset.

2) ANNOTATION PROCESS (CLASSIFICATION)
We classify facial features using YOLO Marker, an open source tool for YOLO labeling distributed for free on AlexeyAB GitHub. In order to more accurately determine that it is a human face, 7 features of the face region are selected and classified as shown in Table 1. The data set is labeled with the index, class number, and bounding box coordinate values and stored in a txt file for each image. Then, a list file for the entire image is also created. Fig. 7 shows the matching structure for the result of labeling using YOLO Marker. There is a class number that was initially set for labeling, and then the square coordinate values for each labeling class are shown. For example, in the case of the first value [4 0.5 0.1 0.2 0.3], the class name is 4, the second is the x value, the third is the y value at the center point of the rectangle (0.5, 0.1). The fourth value is the width at 0.2, and the last value is the height at 0.3. (The width of the entire screen is set to 1.0, and the height is set to 1.0.) Class number 2 indicates a mask, and class number 4 indicates a face or head.

3) FACE DETECTION APPLYING TRAINING MODEL
Training of the Tiny-YOLOv3 model for face features was carried out on the Jetson Nano. We shown in Fig. 8 the architecture of deep learning model for face detection obtained from learning of dataset. We outputted results through input, Convolution step 13, MaxPooling step 6, Up-sampling step 1, step 2 of detection and output. It can recognize human faces through training model using 7 features. It detects faces which is the most important part that is critical to recognize human among extracted features and make attributes like eyes, nose, mouth as the main

| Class name     | Class number |
|---------------|--------------|
| Eye_area      | 01           |
| Mask          | 02           |
| Eyebrow       | 03           |
| Face(head)    | 04           |
| R_eye         | 05           |
| L_eye         | 06           |
| mouth         | 07           |
priorities. If there are four features or more, including face feature, it transfers extracted face area data to the authentication system. Fig. 9 shows the inference result using the training model.

IV. STRUCTURE AND PROTOCOL OF THE PROPOSED SYSTEM

A. USER AUTHENTICATION SYSTEM

A MariaDB is built on the Jetson Nano and a database for visitor access control is created. A table for storing access-related information is created like Table 2. The “pictureTBL” records the image of the visitor, their identity, the number of simultaneous visitors, and the time of the visit. In addition, for future post-processing, visit records are registered in the “visitorLogTBL” and managed separately. If there are two or more visitors, the name and information of the front-most visitor is recorded.

B. PROTOCOL

Fig. 10 is a flow chart that shows the process of the proposed system in action. Algorithm 1 is the pseudo-code suggested for the flow chart of the proposal system in Fig. 10.

Algorithm 1 is a pseudo-code that recognizes visitors with the PIR sensor and the CCTV camera installed on the main door. If it does not recognize a visitor after a long distance or while wearing a mask, it will send a message to the user so that the visitor’s face can be checked. In this process, it photographs the visitor, creating a visit history. It unlocks the door when it recognizes a visitor’s face and recognizes the visitor as a family member. When the initial identification fails, it requests additional identification.

[Step of visitor recognition]

1. As shown in Fig. 10, the visitor may be a family member or a guest for the purpose of visiting, a passerby at the front

| Table’s Name   | Field’s Name   | Data type     |
|----------------|----------------|---------------|
| pictureTBL     | No             | INT           |
|                | VisitorTime    | TIMESTAMP     |
|                | VisitorCount   | INT           |
|                | VisitorName    | VARCHAR2      |
|                | VisitorImage   | BLOB          |
| visitorLogTBL  | No             | INT           |
|                | VisitorName    | VARCHAR2      |
|                | VisitorTime    | TIMESTAMP     |
|                | VisitorLog     | TEXT          |

**TABLE 2. Definition of tables in the database.**

**FIGURE 9.** Inference result using face feature learning model.

**FIGURE 10.** Flow chart of the proposed system.
The system checks whether the visitor is a member of the family, and if the visitor is a family member, the door lock password is pressed.

2. If the visitor is a guest or family member, the system checks the visitor. In order to check the visitor’s face, the model finds a region of the face to recognize the face. However, if it cannot detect that face region, the face will not be detected, and the system will output the message “Please come closer” or deliver that message through the speaker.

3. If the region of the face of the visitor is found, then the face image of the visitor is stored.

[Step of face recognition]
1. Recognize a face in the CCTV image.
2. If a face is not recognized, check whether a mask is worn, and if the face is not recognized due to a mask, output the message “Please pull down your mask” or deliver it through the speaker.
3. If a mask is not worn, move to the step of finding the face region again and take a picture of it again.
4. If the face is now recognized, check whether it is a family member stored in the database.

[Steps for post-authentication process]
1. If the visitor is a registered family member, authenticate them through simple authentication and automatically open the front door.
2. In the case of a guest, the face image of the visitor is delivered to the manager. The manager may then open the front door after confirmation. If the system fails to recognize a face registered in advance, the door lock password may be entered to authenticate the visitor and then open the front door. In the proposed system, the door lock is a device that inputs the password through a touchpad and opens the front door when the correct password, which was stored in advance, is input. The front door lock system authenticates visitors when the password entered as a number string on the touchpad matches the password stored in advance.

Fig. 11 is a chart that shows the process of checking how many visitors are at the front door. In this way, it is possible to confirm how many visitors there are. This function is necessary to ensure that when a visitor enters a password into the door lock, no one is peeking from behind. The proposed system can identify the number of visitors through CCTV and protect against peeking, which is a social engineering attack [31].

Algorithm 2 is the pseudo-code that figures out the number of visitors in front of the main door in Fig. 11.

When face detection fails to authenticate a family member, the proposed method requires an additional authentication. This is not a problem if there is only one visitor, but if there are multiple visitors or passing people, the password may be exposed. In other words, determining the number of visitors helps to avoid shoulder surfing. If the number of visitors detected by video through CCTV is two or more, it senses shoulder surfing from behind. It determines whether or not a visitor is shoulder surfing based on his or her position, face size, and the focus of an eye from behind. If the shoulder surfing attack is detected,
the system requests the visitor enter a lengthy or complex password.

Steps for CCTV image capturing
1. Using CCTV, faces are found in the image using a model that has been trained in advance with deep learning.
2. The number of faces in the recorded image is determined.
   - In cases of at least two visitors
     - If the number of visitors is two or more, the system checks whether one visitor is peeking.
     - Shoulder surfing attack? If the person at the back comes near the front door or looks at the door lock over the front visitor’s shoulder, it is suspected that they are peeking. Then, the door lock display will display the alert message “Please enter the password carefully so that no one else can see.” The user then protects against this social engineering attack by entering the password carefully.
   - In the case of a single visitor, or not peeking
     - 3. If there are more than two visitors but no peeking is detected, or if there is only one visitor, the message “Please, input the password” is displayed.

Steps for password input
1. The visitor inputs the password on the touchpad.
2. If the password entered matches the password stored in the database, the front door opens.

Because this is a system that determines whether or not to open the door lock through face recognition, various attacks are possible. For example, users can be authenticated by using photographs of family members. Therefore, the proposed system identifies the changing of feature points on the face and confirms whether or not the face is a photo, in which case user authentication is blocked.

Fig. 12 shows the process of checking whether a visitor is a family member using CCTV at the front door. The proposed model detects the face of a visitor in the image taken through CCTV with a model trained by deep learning. Authentication is completed when the face image taken with CCTV is compared to the face information of family members registered in advance, and the comparison value is higher than the set threshold. Then, the information of visitors is registered in the database access log table and the next step is performed. If additional authentication is required, the door lock password may be entered to open the front door lock and authentication is completed.

V. ANALYSIS AND EXPERIMENTAL RESULTS OF THE PROPOSED SYSTEM
A. ANALYSIS OF USER AUTHENTICATION SYSTEM
When the visitor’s face is recognized with a CCTV image, 81 feature points are detected using the model that was trained on the file shape_predictor_81_face_landmarks.dat. Using the HoG feature as a face detector, a linear classifier is used to find the landmarks of the face such as the eyes, nose, mouth,
For user authentication, the faces of registered members or family members are stored in the database in advance. The vector values of the feature points in the family member’s face images are stored in the familyTbl table in the database. This method extracts the vector values of feature points in the visitor’s face image and compares those with the vector values of the feature points of family members stored in familyTbl. When the threshold value is within the authentication range, authentication is performed.

So, if a visitor’s face is very similar to a registered family member’s face, it may be recognized as a family member. That is, there is a False Acceptance Rate (FAR). In addition, despite being a family member, a False Rejection Rate (FRR) may occur when a face is obscured by a mask, glasses, or other facial covering and some of the vector values of feature points are distorted. Table 3 shows the evaluation indicators, FAR, FRR, and Precision according to the threshold of similarity in the HoG algorithm using Dlib [31].

Thus, because there is FRR and FAR, additional authentication methods are required.

Algorithm 3 is part of Python code that extracts 81 facial features. It uses the Dlib library to extract and save features from the family photo. It is the source code that extracts the feature in the video image collected by the webcam when a visitor is waiting at the main door and confirms if the image is the same person by comparing the vector value of the previously saved feature of the family to the extracted vector value.
In the proposed model, a pass number can be additionally requested or the administrator can perform an additional verification procedure.

The 81 features from the family are saved in the system as vector previously. The two members of family is registered in Fig 13. A face is detected with a webcam (CCTV), and 81 features of Person 0 and Person 1 are found by facial recognition technology and similarity with the vector value of the family stored in the system is calculated and recognizes as family member.

In case of Fig. 14, The method is the way that finds 81 features and authorizes they are family like Fig.13. In Fig. 14, the name of a family member is displayed on the face in screen.

According to Fig.14, When two visitors are in front of the CCTV, their faces are detected and they are asked to lower their masks in order to extract accurate face features from the proposed model. In addition, this is a screen that captures the feature points of the face when the mask is lowered.

**B. DATASET-BASED LEARNING RESULTS AND EXPERIMENTS**

1) LEARNING RESULT AND EXPERIMENTS USING DATASET

Fig. 15 shows a graph of the loss function of the trained images by applying 7 face feature classification data sets to both YOLOv3 and Tiny-YOLOv3 models, respectively.

The x-axis displays the number of repetitions, and the y-axis displays the average loss rate. The number of iterations of the model trained with the YOLOv3 algorithm was 3,400 and the average loss was 0.9384. The number of iterations of the model trained with the Tiny-YOLOv3 algorithm was 182,100 and the average loss was 1.6490.

When the average loss rate has a relatively linear shape, the result of a good training model is obtained. As shown in Fig. 17-(a), YOLOv3 derives a training model that is meaningful even when the number of repetitions is 3,400 times. However, in the case of Fig. 17-(b), the number of repetitions was increased because Tiny-YOLO did not come up with a meaningful training model. Fig. 17-(b) shows that 180,000 times were performed, but there was no difference in the loss rate.

**C. EXPERIMENT RESULTS**

The proposed system enables real-time processing of visitor face recognition with the Jetson Nano as shown in Fig. 16. It detects faces, eyebrows, masks, and eyes. It collects face images which have various looks such as wearing eye glasses, mask, cap and make up. Moreover, it can identify gender, age, and lateral angle of the face in order to decrease false positive ratio of face detection and to increase accuracy.

The Raspberry Pi, which is cheaper than the Jetson Nano, was also tested in the same environment. The model applied to the Raspberry Pi was trained with Tiny-YOLOv3. Three faces are detected, 4 face features are detected for the left person, 3 face features are detected for the middle person, and 3 face features are detected for the right person. Figure 17 shows the authentication by finding the 7 features and determining whether they are the same as the registered person.

The left face of Fig. 17 is detected with Raspberry Pi and shows the probability of whether a feature exists.

Fig. 18 shows the actual inference result and FPS output of the two algorithms. Fig 18-(a) shows the YOLOv3 execution result screen and Fig 18-(b) shows the Tiny-YOLOv3 execution result screen.

**VI. COMPARATIVE ANALYSIS AND DISCUSSION OF THE PROPOSED SYSTEM**

**A. COMPARATIVE EVALUATION OF SYSTEM**

Table 4 shows the average speed (FPS) and average accuracy (%) when running YOLOv3 and Tiny-YOLOv3 on the Jetson Nano board and detecting face features. Although the
The performance is lower than that of the Jetson Nano, a cheap Raspberry Pi was also used and compared together. The accuracy of face feature detection was calculated by taking the average of the face classes that were recognized in all faces. As shown in Fig 18, when YOLOv3 was run with the Jetson Nano, the average detection speed was 2.4 FPS and the accuracy was 90.3%. When Tiny-YOLOv3 was run with the Jetson Nano, the average detection speed was 6.5 FPS and an accuracy of 86.3%. Although this was slightly less accurate, it was found to be about three times faster in terms of speed. As a result of executing Tiny-YOLOv3 on the Raspberry Pi, it was confirmed that the calculation result of frames per second was less than 1 in real-time video, but the detection result was derived and the accuracy was 76.3% (Table 4).

FPS means the number of frames performed per second. As a test result, the average processing speed of Tiny-YOLOv3 is 6.5FPS, which means the speed performing average 6.5 frames per second. Therefore, it was found that the time taken for face recognition in the access system could be shortened compared to other algorithms and hardware by using the Jetson Nano board and Tiny-YOLOv3.

We measured the face recognition speed and stated the situation about detecting Single user and two or more users in Table 5.

As the result of comparison, there is a little difference between 1 and 2 or more, but no difference in average FPS.

Table 6 is the result of comparing processing speed of YOLO which is the core algorithm and Faster R-CNN which is the representative object extraction algorithm.

We compared YOLO and Faster R-CNN algorithm in GTX 1080Ti desktop environment and in Jetson system board environment. In inference time in GTX1080Ti desktop environment, YOLO is 3 times faster than Faster R-CNN. The processing speed of Tiny-YOLOv3 is 2.7 times faster than

### Table 4. The comparison and analysis of processing speed and accuracy about each algorithm on each system board.

| System board | Algorithm | Inference speed (FPS) | Accuracy (%) |
|--------------|-----------|-----------------------|--------------|
| Jetson Nano  | YOLOv3    | 2.4                   | 90.3         |
|              | Tiny-YOLOv3 | 6.5                  | 86.3         |
| Raspberry Pi | Tiny-YOLOv3 | Not measurable | 76.3         |

### Table 5. Recognition Speed of a single user.

| Object       | FPS          |
|--------------|--------------|
| Single user  | 2.2–2.4      |
| 2 or more    | 2.2–2.4      |

### Table 6. Performance Comparison.

| Algorithms  | Faster-RNN | YOLOv3 | YOLOv3 | Tiny-YOLOv3 |
|-------------|------------|--------|--------|-------------|
| Execution environment | GTX 1080Ti | GTX 1080Ti | Jetson Nano | Jetson Nano |
| Inference speed | 7.5 | 22.6 | 2.4 | 6.5 |
that of YOLOv3 in the Jetson Nano system board, which can stand alone due to the characteristics of the suggested system.

B. EVALUATION OF ANALYSIS OF SECURITY FIELD

The requirements mentioned in the introduction are satisfied as follows.

First, the proposed model can be used to identify the number of visitors based on faces found with CCTV. If the number of people decreases after a certain period of time, it is judged to be a passerby. Otherwise, it is judged to be a visitor or an intruder.

Second, if the visitor at the front touches the password for user authentication, the person behind the visitor can peek at it. So, it is necessary to check whether the visitor standing behind the front person is a family member or a companion.

If the companion’s face is perceived to be approaching the door, or if the CCTV recognizes that the eyes of the visitor are trying to steal the password, it outputs a message to the front visitor on the display warning them of this. In other words, if the visitor standing behind is judged to be peeking, the visitor standing in front receives a “Your password is too short” message when entering a registered password.

Third, family members are registered in advance. It learns from the registered photos, determines whether a face has been registered in advance from the video image entered into the CCTV, and authenticates the visitor.

When authentication is performed, a probabilistically calculated result is derived by comparing the CCTV image with a previously registered face. If the calculated result is higher than the threshold, primary authentication is performed, but if it is not 1, secondary authentication is required because incorrect authentication may occur. There are various secondary authentication methods, but in this study, secondary authentication can be performed by entering a short password. When this authentication is completed, the door is opened.

Fourth, the image taken by CCTV in the proposed model can be transmitted to the manager’s phone through the smart home service. The manager can detect intruders by checking the image. If password authentication fails or a family member’s visit is not registered, the visitor cannot enter the house with authentication.

In this case, it checks the visitor with a smartphone and opens the door remotely. The result of the proposed method is that the important image is stored on the server and the authenticated visitor’s name is stored in a log file. This log file can also provide additional services when family members return home, access information, access pattern analysis, or have a long absence.

VII. CONCLUSION

Deep learning, a major technology of artificial intelligence, is growing rapidly as it is applied to voice recognition and image recognition. In particular, deep learning technology in the field of image recognition is being applied as a core technology to autonomous driving and crime prevention monitoring systems, which are emerging as future industries. As for deep learning models in the field of image recognition, various algorithms that have improved and developed CNNs capable of image processing have been proposed.

In this paper, we introduce various object detection algorithms including CNN. CCTV detects the face of a visitor and then recognizes 81 feature points in the face to create a set of vector values based on the features. Members of the family are registered in advance with face images. In this study, we designed and implemented a system that opens the front door after recognizing a new visitor as a member of the family when the difference in recognized facial feature vector values between the CCTV image and the image stored in the database is smaller than the threshold.

The proposed model was applied to a microprocessor that is portable and operates various sensors and the system was tested. Although the system was applied to two representative microprocessors, the Raspberry Pi and the Jetson Nano, it was difficult to apply in real life because the operation was very slow and the desired result was not obtained in the cheap and widely used Raspberry Pi. However, the desired result was derived from the Jackson Nano with a built-in GPU.

Specifically, in the proposed model, YOLOv3’s inference time was 2.4 FPS with an accuracy of 90.3%, and TinyYOLOv3’s inference time was 6.5 FPS with an accuracy of 86.3%.

Future studies include the development and research of a system that recognizes faces when controlling access to shops or restaurants, automatically registers frequent visitors, identifies the number of visitors, stores visiting records, and controls access according to visitors. In addition, it is necessary to develop an access authentication system that can enable easy access with face recognition rather than fingerprints or passwords, and has a low false detection rate.

REFERENCES

[1] T.-H. Tsai, C.-C. Huang, C.-H. Chang, and M. A. Hussain, “Design of wireless vision sensor network for smart home,” IEEE Access, vol. 8, pp. 60455–60467, 2020, doi: 10.1109/ACCESS.2020.2982438.
[2] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, “Convolutional neural networks based fire detection in surveillance videos,” IEEE Access, vol. 6, pp. 18174–18183, 2018, doi: 10.1109/ACCESS.2018.2812835.
[3] M. T. Bhatti, M. G. Khan, M. Aslam, and M. J. Fiaz, “Weapon detection in real-time CCTV videos using deep learning,” IEEE Access, vol. 9, pp. 34366–34382, 2021, doi: 10.1109/ACCESS.2021.3059170.
[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.
[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), May 2016, pp. 779–788. [Online]. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Redmon_You_Only_Look_CVPR_2016_paper.pdf
[6] S. W. Shah and S. S. Kanhere, “Recent trends in user authentication—A survey,” IEEE Access, vol. 7, pp. 112505–112519, 2019, doi: 10.1109/ACCESS.2019.2932400.
[7] K. M. Renuka, S. Kumari, D. Zhao, and L. Li, “Design of a secure password-based authentication scheme for M2M networks in IoT enabled cyber-physical systems,” IEEE Access, vol. 7, pp. 51014–51027, 2019, doi: 10.1109/ACCESS.2019.2908499.
[8] H.-J. Mun, “Biometric Information and OTP based on authentication mechanism using blockchain,” J. Converg. Inf. Technol., vol. 8, no. 3, pp. 85–90, 2018, doi: 10.22156/CS4SMB.2018.8.3.085.
[9] A. Mohammed Ali and A. Kadhim Farhan, “A novel improvement with an effective expansion to enhance the MD5 hash function for verification of a secure E-Document,” IEEE Access, vol. 8, pp. 80290–80304, 2020, doi: 10.1109/ACCESS.2020.2989050.

[10] H.-J. Mun, S. Hong, and J. Shin, “A novel secure and efficient hash function with extra padding against rainbow table attacks,” Cluster Comput., vol. 21, no. 1, pp. 1161–1173, Mar. 2018, doi: 10.1007/s10586-017-0886-4.

[11] J. Shin, Z. Liu, C. M. Kim, and H.-J. Mun, “Writer identification using intra-stroke and inter-stroke information for security enhancements in P2P systems,” Peer-to-Peer Netw. Appl., vol. 11, no. 6, pp. 1166–1175, Nov. 2018, doi: 10.1007/s12083-017-0606-0.

[12] Science, ICT & Technology Trends. (Aug. 2019). Spreading the Introduction of Facial Recognition and Seeking Ways to Revitalize it in Korea, [Online]. Available: http://www.itfind.or.kr/admin/getFile.htm?identifier=02-001-190725-000007

[13] H. Li, J. Hu, J. Yu, Y. Bu, and Q. Wu, “UFaCeNet: Research on multi-task face recognition algorithm based on CNN,” Algorithms, vol. 14, no. 9, p. 268, Sep. 2021, doi: 10.3390/a14090268.

[14] L. L. Chambino, J. S. Silva, and A. Bernardino, “Multispectral facial recognition: A review,” IEEE Access, vol. 8, pp. 207871–207883, 2020, doi: 10.1109/ACCESS.2020.3037451.

[15] K. Kim, B. Lee, and J. W. Kim, “Feasibility of deep learning algorithms for binary classification problems,” J. Intell. Inf. Syst., vol. 23, no. 1, pp. 95–108, Mar. 2017, doi: 10.1007/s10844-017-0309-5.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.

[17] R. Girshick, “Fast R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 1440–1448.

[18] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2016.

[19] M. Lee, and H. Mun, “Comparison analysis and case study for deep learning-based object detection algorithm,” Int. J. Adv. Sci. Converg., vol. 2, no. 4, pp. 7–16, Dec. 2020, doi: 10.22662/ijasc.2020.2.4.007.

[20] H. Chen, Z. He, B. Shi, and T. Zhong, “Research on recognition method of electrical components based on YOLO v3,” IEEE Access, vol. 7, pp. 157818–157829, 2019, doi: 10.1109/ACCESS.2019.2950053.

[21] X. Jiang, T. Gao, Z. Zhu, and Y. Zhao, “Real-time face mask detection method based on YOLOv3,” Electronics, vol. 10, no. 7, p. 837, Jan. 2021, doi: 10.3390/electronics10070837.

[22] M.-H. Lee, J.-Y. Kang, and S.-J. Lim, “Design of YOLO-based removable system for pet monitoring,” J. Korea Inst. Inf. Commun. Eng., vol. 24, no. 1, pp. 22–27, Jan. 2020, doi: 10.6109/jkie.2020.24.1.22.

[23] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” 2018, arXiv:1804.02767.

[24] W. Fang, L. Wang, and P. Ren, “Timier-YOLO: A real-time object detection method for constrained environments,” IEEE Access, vol. 8, pp. 1935–1944, 2020, doi: 10.1109/ACCESS.2019.2961959.

[25] K. M. Lee, H. Song, J. W. Kim, and C. H. Lin, “Balanced performance for efficient small object detection YOLOv3-tiny,” in Proc. Korean Soc. Broadcast Eng. Conf. Anseong, South Korea: The Korean Institute of Broadcasting and Media Engineers, Nov. 2018, pp. 117–118.