INTERNATIONAL JOURNAL OF INTEGRATED ENGINEERING
ISSN: 2229-838X e-ISSN: 2600-7916

IJIE

Vol. 17 No. 2 (2025) 246-256
https://publisher.uthm.edu.my/ojs/index.php/ijie

Haar-VGG: Face Attendance System

Quah Xuan Ying[1], Raja Abdullah Raja Ahmad[1]*, Muhammad Imran Ahmad[1], Zamri Zahir Ahmad[1], Ahmad Ashraf Abdul Halim[1], Mohd Wafi Nasrudin[1]

[1] *Faculty of Electronic Engineering & Technology, Universiti Malaysia Perlis, 02600 Arau, Perlis, MALAYSIA*

*Corresponding Author: rajaabdullah@unimap.edu.my
DOI: https://doi.org/10.30880/ijie.2025.17.02.020

Article Info

Received: 23 October 2024
Accepted: 1 July 2025
Available online: 28 July 2025

Keywords

attendance system, face recognition, Viola-Jones algorithm, VGGFace, transfer learning, Python

Abstract

Attendance taking is a crucial practice in educational institutions in Malaysia, but the traditional manual method is time-consuming and risky, particularly in the post-Covid era. To address this, a face recognition attendance system using Python is developed. The Viola-Jones algorithm known as Haar is utilized for face detection, and transfer learning on VGGFace is applied for model training, using 195 images from FLW dataset and volunteers among students. The system achieves a validation and testing accuracy of 1.0 through image preprocessing and augmentation. The attendance system includes a user-friendly graphical interface and live webcam feed, enabling instant recognition and recording of attendance. Integration with a MySQL database allows easy access to attendance records for teachers. This advanced system saves time, reduces the risk of virus transmission, and simplifies attendance management, offering a convenient and efficient solution for educational institutions.

1. Introduction

In Malaysia, there is a daily ritual for keeping track of attendance in almost every class at every level of school, college, and institution. Regular class attendance by students is important for performance evaluation and quality control in the current academic system. Most institutions continue to rely on insecure, time-consuming methods such as calling people by name or having them sign documents. However, the traditional attendance taking indeed bare a risk when it comes to the post-Covid situation.

During the pandemic, students must adapt to a new way of learning by attending classes or lectures via an online platform such as Google Meet, Zoom, Webex, and so on. The method of taking attendance could be traditional or by utilizing the function provided by those platforms to record students' attendance. After the pandemic, everything begins to return to its pre-pandemic state. The attendance-taking procedure is no different. Thus, in such a case, using an automatic attendance system might be beneficial.

As we know, the human body is unique, with physical characteristics that are relatively fixed and individualized. The characteristics can be used to identify individuals through forms of biometric technology such as facial recognition, retina scans, fingerprint mapping and etc. According to Kaspersky [1], biometrics is widely used nowadays as it is convenient to use because the characteristics are always with us, cannot be lost and are difficult to impersonate. The utilization of biometric systems can greatly improve overall efficiency and reduce the time consumed during the attendance-taking process.

Other than biometric systems, the use of Quick Response (QR) code-based attendance systems and Radio Frequency Identification (RFID) card systems as shown in Fig. 1, are some of the digital attendance systems that can replace the traditional method. At any rate, a further issue with the QR code-based attendance method is that it requires online access and takes between 5 and 10 minutes to acquire class attendance, which can occasionally interfere with learning when the internet connection is slow. In short, the implementation of biometric on the attendance system is more secure and not easy to be cheated as compared to the other methods. Therefore, this project focuses on the implementation of facial recognition to develop an attendance system via Python.

(a) **(b)**

Fig. 1 *Example of attendance system (a) QR code-based; (b) RFID card*

1.1 Face Recognition Algorithm

Gupta et al. [2] claimed in their paper, "Student Attendance System Based on the Face Recognition of Webcam's Image of the Classroom," that skin colour and facial features are utilized to identify, and extract faces in the acquired images. Furthermore, using RGB colour space to set skin colour values and segmentation reduces the time required to search for face images. They also claimed that implementing the Laplacian of Gaussian (LoG) filter performed well in extracting facial components under various lighting situations. Lastly, an artificial neural network known as the Feedforward Neural Network (FFNN) is performed in the classification to solve the pattern recognition issue.

In the study "Real-Time Smart Attendance System using Face Recognition Techniques" by Sawane et al. [3], a good camera is required to capture high-quality images for image processing. In the pre-processing stage, images are cleaned, and noise is eliminated to improve clarity. Followed by face detection, the Viola-Jones algorithm 12 is used to identify 68 facial landmarks on a person's face. Next, the face tracking and face landmark search are performed using face bounding box detection and a constrained local model. Finally, deep learning is implemented for face recognition. The enrolled images in the database will be compared to the captured images to determine attendance. Additional features on body temperature checking and hand sanitization are available in their system to cope with the Covid-19 situation.

Ali et al. [4] proposed using the HOG approach for face detection and the SVM classifier. Based on the research, they trained the classifier to recognize the individual who is the closest match after analyzing the measurements from a new test image. A person's name will be generated and used to record attendance. However, they also stated that the proposed approaches failed to identify two identical twins.

According to Sai et al. [5], they acquired their images either from computer images or video streams. The student database is connected to the recognized faces, and an Excel spreadsheet is used to track attendance. They used the Haar cascade classifier for face detection, which Viola and Jones invented. By identifying a face's negative and positive features in an image, the classifier trained machine learning to detect objects. To determine whether a face is present in the image, it typically performs about 20 levels of measurements, checking every possible location in the image and every possible face size. Similar to this paper, the algorithm is trained to recognize the distinctive characteristics of each student. Additionally, the implementation of the Local Binary Pattern Histogram (LBPH) is used for face recognition. The first step in the LBPH is to produce an intermediate image that best captures and highlights the facial features of the 13 original images. The final histogram depicts the characteristics of the original image. The algorithm is fed with the training dataset, represented by various histograms.

Bhattacharya et al. [6] employed face-tracking technology in their system. The Viola-Jones algorithm is used to detect faces first, and then the correlation tracker from the dlib library is used to track faces from frame to frame. A face log was created by transforming the face into a new frame in the real-time video sequence. The significant parameters in their system included pose estimation, image sharpness, size, and brightness. These parameters ensure that the faces in the face log are high quality. Even though the pre-processed images are in high-dimensional spaces, CNN extracted a low dimensional distinguishing characteristic from the face images. FFNN is a metric to assess how well a neural network classifies an image.

2. Experimental Analysis

Traditional attendance systems frequently relied on time-consuming and error-prone manual techniques, such as sign-in and sign-out. However, technological improvements have made facial recognition a commonly accepted approach for automating the attendance process.

Fig. 2 shows the complete flowchart of the proposed attendance system based on face recognition with the use of the Viola-Jones algorithm and transfer learning model (trained model). The process began with data collection, where both training and testing data were obtained from either a database or input images captured via a webcam under various lighting conditions and angles. The images in the dataset were pre-processed beforehand. After capturing images from the webcam, the system proceeded to prepare the images for further processing. This involved transforming the images into grayscale, adjusting their size to a standardized format, and normalizing them to ensure consistent and reliable results. In the training phase, image augmentation was performed to increase the overall dataset.

Next, the transfer learning model was utilized to extract features from the pre-processed images and train them on the provided dataset. This allowed the model to learn discriminative features that could be used for individual recognition. Recognition was performed by comparing known faces against predicted probabilities with the threshold in consideration. The accuracy of the model was evaluated. Finally, the results were analyzed, and attendance was marked based on the outcomes, with the records being stored in the database.

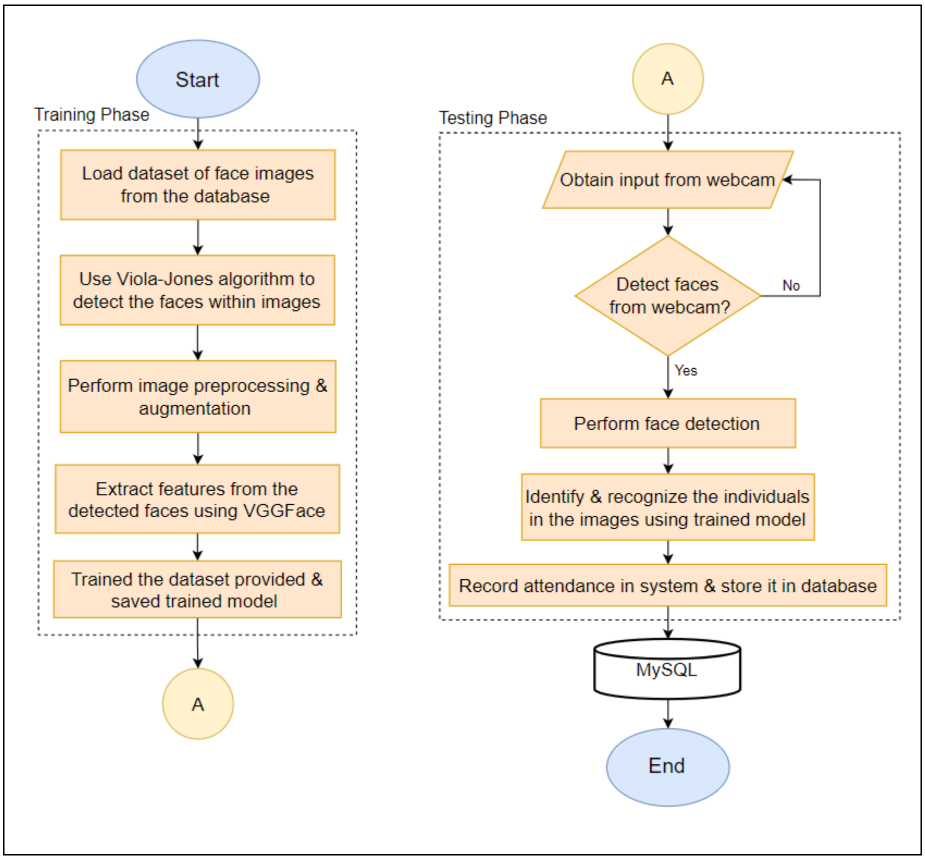

Fig. 2 *Flowchart of proposed attendance system based on face recognition*

2.1 Image Acquisition

Image acquisition is the retrieval of photos from an external source for training purposes, where they are recorded under varying lighting situations, angles, and facial expressions to strengthen the face recognition algorithm. A sizable collection of photographs with the faces of the people who are most likely to be recognized should be included in the images. The LFW database (see Fig. 3) and the student photos gathered in Universiti Malaysia Perlis (UniMAP) (via Google Form) were both utilized in this study. Personal information such as name, matric number and current academic year of study were required to be provided along with the attachment of personal images, which can accept up to a maximum of 10 images per submission.

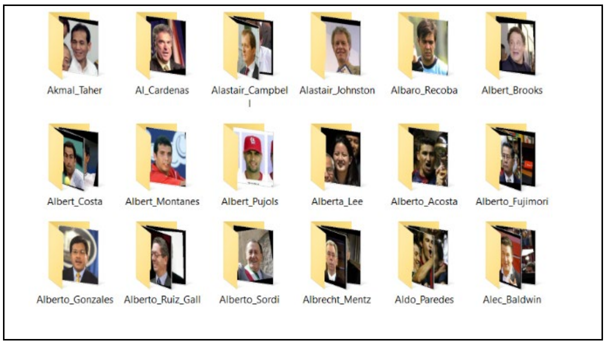

Fig. 3 *Samples face images from the LFW database [7]*

2.2 Face Detection

The Viola-Jones algorithm is a widely used method for object detection, particularly for face detection. It is based on a cascade of classifiers, where each classifier is trained to identify a specific feature of the object of interest. The key component of this algorithm is the Haar cascade classifier, which uses a series of simple geometric features called Haar-like features to identify objects within an image.

The Viola-Jones algorithm is particularly well-suited for face detection because of the way it extracts features. Faces have many distinctive and unique features that can be easily distinguished from non-faces, such as the eyes, nose, and mouth. The Haar-like features used by the algorithm are able to capture these features effectively and the cascade of classifiers allows the algorithm to quickly reject non-faces (see Fig. 4).

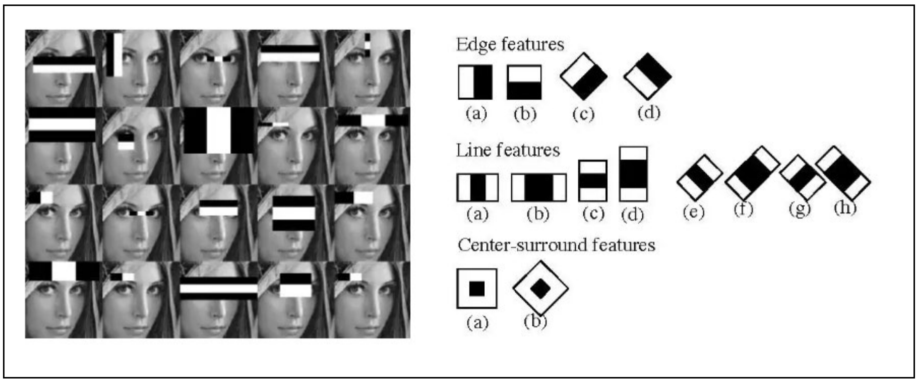

Fig. 4 *Haar-like features on face detection [8]*

2.3 Image Pre-processing & Image Augmentation

The pre-processing step is essential to ensure that the images are of the same size, brightness and contrast when they are passed to the feature extraction and face recognition steps. First, the images are converted to grayscale (see Fig. 5). This is an optional step, but it can improve the performance of the face detection step if the images are captured in poor lighting conditions. Next, the Haar Cascade classifier passed through every image again to detect the faces and set the region of interest (ROI) to be further resized. Finally, the images are resized to a consistent size (224 x 224 pixels) based on the ROI and saved in a new folder.

Fig. 5 *Convert RGB image to grayscale [8]*

Following by image augmentation, it is a method that enhances the training dataset by applying various transformations to the images. These transformations replicate real-world variations like rotation, shifting, shearing, zooming, flipping, and pixel filling. By introducing these variations, image augmentation allows the model to learn from a broader range of scenarios and enhances its capability to handle new and unseen data. Since the size of pre-processed images is small, the use of image augmentation is highly encouraged to create a more diverse and robust training dataset.

2.4 Face Recognition: VGGFace (VGG16 Model)

In many applications of face recognition and detection over the past few years, the use of CNNs has led to improvements in accuracy as well as speed. They have demonstrated extraordinary performance in terms of accuracy and robustness compared to other existing approaches. CNNs operate on the principles of Multilayer Perceptron (MLP), which have one input layer, one output layer and several hidden layers as in Fig. 6. The basic structure of every CNN consists of four components: convolutional layers, pooling layers, activation functions, and a fully connected layer. In order to introduce nonlinearity into the network, the Rectified Linear Unit (ReLU) is an activation function used in conjunction with convolutional layers to extract features from the input data and pooling layers to reduce the size of the feature map and control overfitting. The fully connected layers take in all the features from the previous layers and generate outputs using weighted connections.

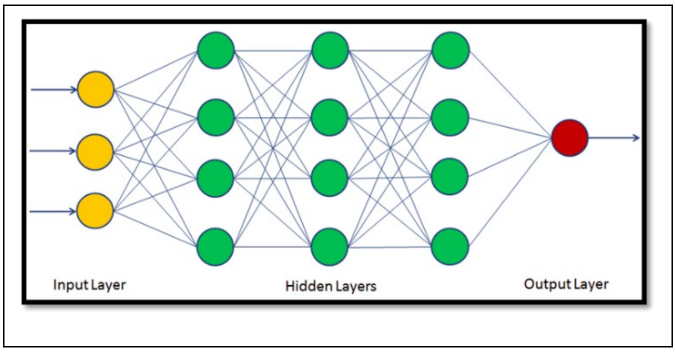

Fig. 6 *Example of MLP concept [9]*

CNNs are known for their ability to learn hierarchical representations of data. Early layers of the network learn to detect simple features like edges and textures, while later layers learn to detect more complex features like shapes and objects. This hierarchical structure allows CNNs to automatically learn features that are useful for a specific task, such as face recognition, without requiring manual feature engineering.

In this study, the VGGFace model is used, which is a pre-trained deep convolutional neural network that has been trained on a large dataset of faces and has been shown to be effective for feature extraction. The input image is processed by a series of convolutional layers, each of which extracts various features from the image using a set of filters. The filters are trained to recognize textures, edges, and shapes. These characteristics are subsequently processed through a number of pooling layers, which minimize the spatial dimensions of the features while preserving the most essential data. Fig. 7 shows the CNN architecture of VGGFace, in which it is the model used in this study. This is done several times, with each iteration extracting more sophisticated features.

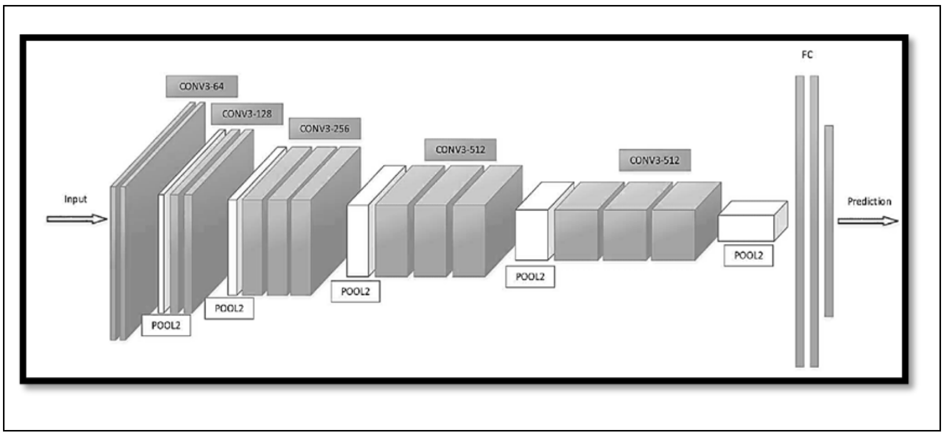

Fig. 7 *Example of MLP concept [9]*

The VGGFace is then taught to identify faces. During training, the model learned to correlate the retrieved features from each face with their respective identification labels. Once the model has been trained, it is finally ready to recognize faces. Face recognition requires locating the face in an unknown image, extracting its features, and comparing them to those of the training dataset's known faces. The system records the attendance of the identified person if the gap between the feature vectors is less than a specified threshold.

Transfer learning is then utilized with the VGGFace model to train the system in recognizing the specific images from the dataset. To accomplish this, the initial 19 layers responsible for pre-trained weights are kept frozen, while additional layers are added and trained specifically for classification and recognition on the dataset.

2.5 Attendance System

Fig. 8 illustrates the layout of the face recognition attendance system. The attendance system was implemented with a GUI designed using Tkinter. The GUI window, created using PyCharm, displayed the live webcam stream and the attendance records stored in a MySQL database. The window featured a "Date | Time" section to show the current date and time during system execution.

Furthermore, the students' attendance records, along with their information, were stored in MySQL Workbench. The system file established a connection with the MySQL database using the MySQL connector in PyCharm, allowing the retrieval and display of the stored data.

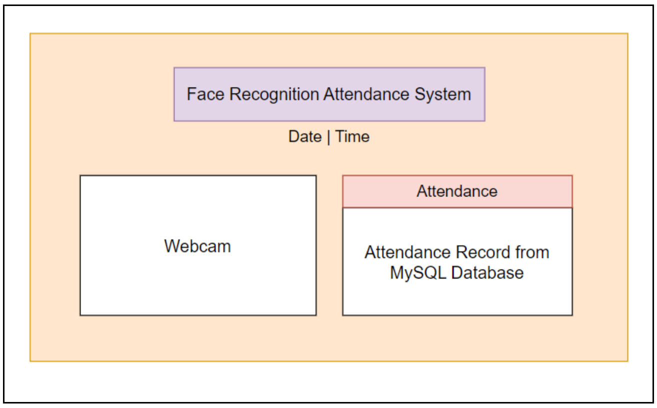

Fig. 8 *Illustration of face recognition attendance system*

3. Result and Discussion

This chapter covers the outcomes of the Face Recognition Attendance System implemented in accordance with the proposed methodology in the preceding chapter. It covers a comparison of the parameters used in face detection and an analysis of the use of data augmentation techniques.

3.1 Image Acquisition

The data was collected from UniMAP students (8 subjects) and combined with the datasets chosen from the LFW database (17 subjects). There were 25 subjects in all, making up 195 photographs, in the database. The photographs were saved in a folder called "dataset", where each subfolder was labelled with its corresponding name. The folders in the left image of Fig. 9 were chosen at random from the LFW database, while the dataset on the right was obtained from UniMAP students.

Fig. 9 *Image acquisition*

Int. Journal of Integrated Engineering Vol. 17 No. 2 (2025) p. 246-256

3.2 Face Detection & Image Pre-processing

Through the use of the Viola-Jones algorithm, a Haar Cascade classifier, the collected images underwent a face detection phase to identify the faces seen in the photos. The images were converted to grayscale for better feature detection. The detected faces were surrounded by white boxes and displayed as seen in Fig. 10.

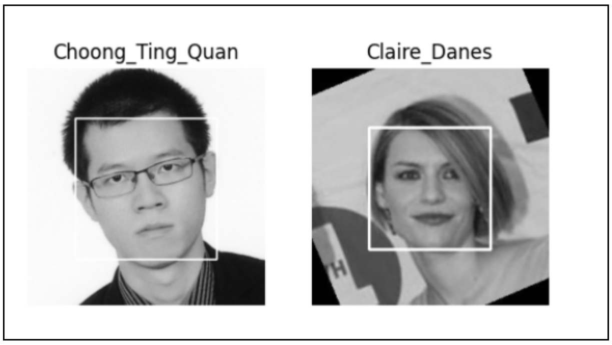

Fig. 10 *Example faces detected on datasets*

Another approach available for the system to capture images with a predetermined size, where face detection was guaranteed for the dataset collection, was to use the “face_collection.py” script created in PyCharm. A live webcam was turned on during execution and a green bounding box was utilized to display the face detected in front of the webcam, as shown in Fig. 11. After clicking the “Capture” button, the image will be captured, and the user will be required to provide their name to be saved in the “dataset” folder.

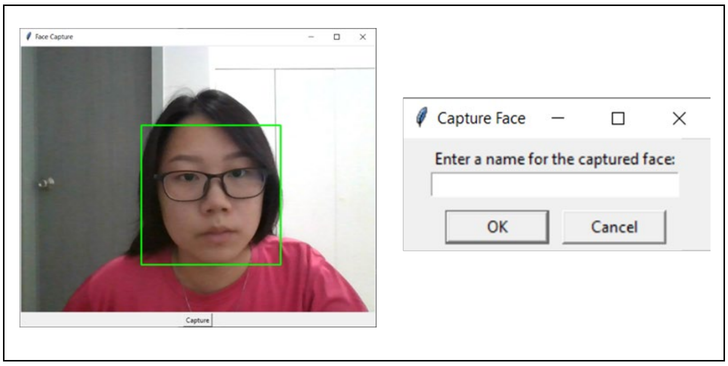

Fig. 11 *Interfaces of “face_collection.py*

After face detection, the extracted faces were scaled to (224, 224) pixels and saved in a separate folder called “preprocessed_dataset” and then further enhanced. Fig. 12 displays the resized images. The images were saved in RGB colour to allow image augmentation to take place on the original colour images before moving to model building and training.

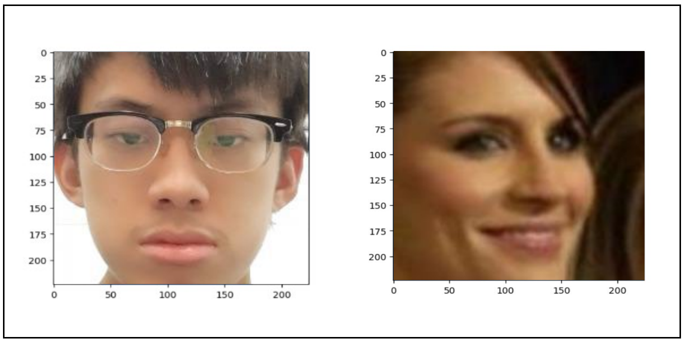

Fig. 12 *Resized images*

3.3 Image Augmentation

Successfully extracted and saved 159 faces in the “preprocessed_dataset” folder. Due to the small size of the dataset, image augmentation was used to increase the number of datasets by altering the available parameters, such as rotation range, width and height shift ranges, shear range, zoom range, fill mode as well as the horizontal and vertical flips, which were then saved in the "augmented_dataset" folder.

This study used a rotation range of ± 20 degrees, "nearest" fill mode and horizontal flip. During the augmentation, the "nearest" fill mode is employed to fill the empty pixels with the nearest pixel values. An example of the original image and its augmented images is shown below (refer Fig. 13). Each image in the dataset was augmented using the defined parameters and batch size of 6, where 6 augmented images will be made from a single image, resulting in 954 augmented images in total.

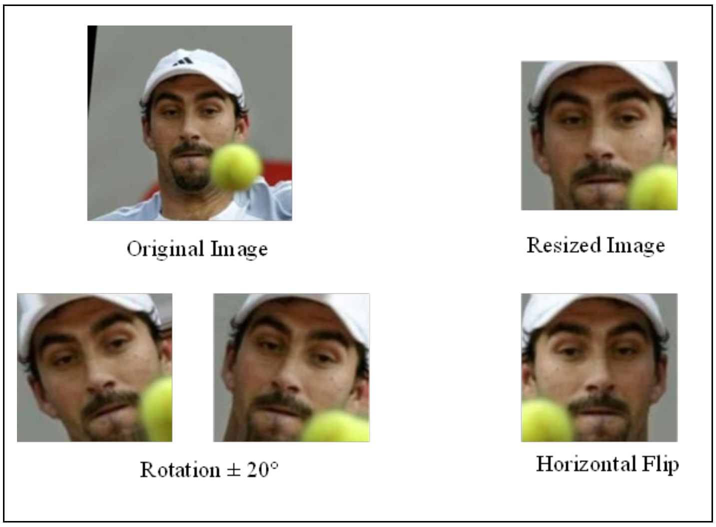

Fig. 13 *Image augmentation*

3.4 Face Recognition on Trained Model

The augmented dataset was split into three sets: training, validation and testing sets, each with 610, 153 and 191 images, respectively. In order to train the newly formed dataset, a model was built using transfer learning on VGGFace, which utilized the VGG16 model architecture. Feature extraction and classification of images were carried out during the execution of the model. 20 epochs were allotted for model training. The model loss and model accuracy can be determined through the line graphs in Fig. 14, where the model began to gain an accuracy of 1.0 after 4 epochs of training. Furthermore, the graphs indicated that the model was neither overfit nor underfit.

The individuals featured in the images were correctly recognized and predicted during the model evaluation on the testing set, as shown in Fig. 15. All of the test subjects were precisely predicted by the model when compared to the true labels for the whole testing period, yielding an accuracy of 1.0.

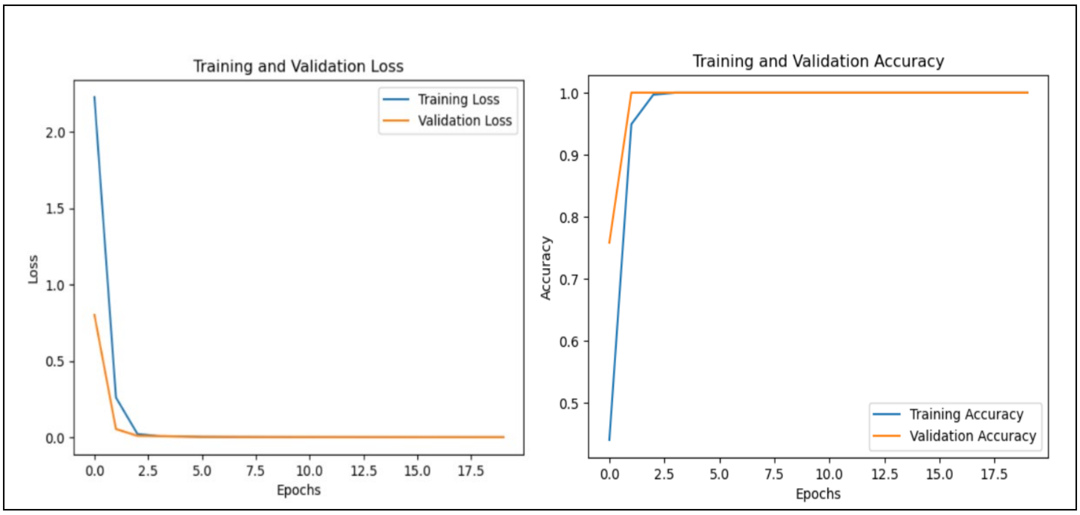

Fig. 14 *Model loss and model accuracy*

Int. Journal of Integrated Engineering Vol. 17 No. 2 (2025) p. 246-256

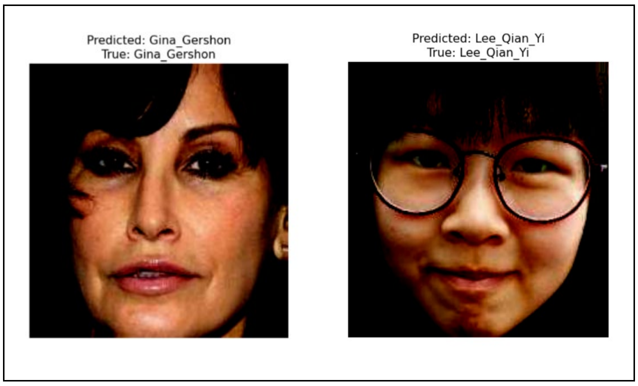

Fig. 15 *Model loss and model accuracy*

3.5 Face Recognition Attendance System

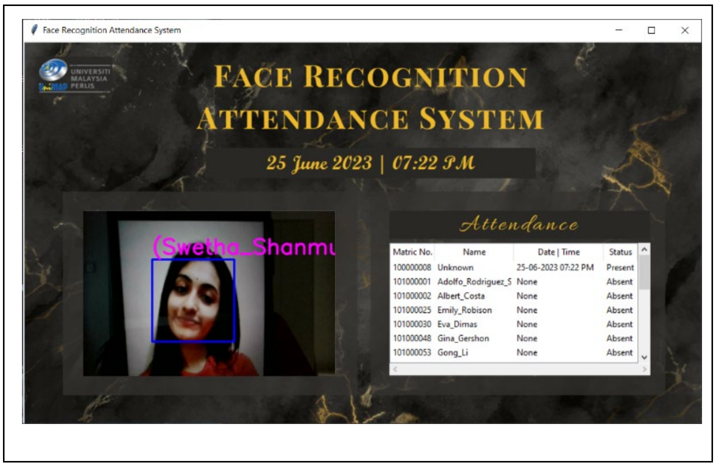

Fig. 16 *System interface with live recognized face and attendance record*

The next step was to mark the presence of detected and recognized faces. Fig. 16 shows a Tkinter GUI window that was created as the interface for the face recognition attendance system. As the system ran, the interface displayed the current time and date. The left box then presented a live webcam stream of a person who was correctly identified as “Swetha_Shanmugam” by the system. In addition, there is a table that contains the records of the information belonging to the respondents in the dataset. The attendance table is connected to the MySQL database server, enabling real-time updates every minute.

After the webcam successfully detected a face in front of it, the names of recognized individuals were immediately shown, whereas unrecognized individuals were marked as “unknown” (see Fig. 17).

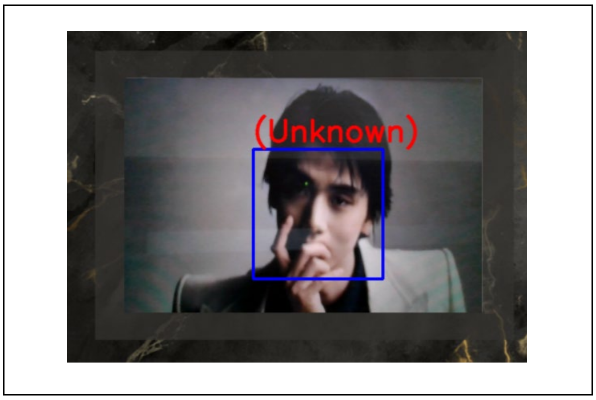

Fig. 17 *Snapshot of system interface on unrecognized individual*

The details, such as name, matric number (matric_no), and year, were saved in MySQL Workbench along with the status, exact date and time of the attendance marked. As shown in Fig. 18, the attendance table has five columns: Matric_no, Name, Year, Date_Time and Status. The matric number for the "Unknown" person was set to be generated randomly between 100000000 and 100000010. The spacing between individual names was recorded as "_" rather than a space. The current academic year for each student might be denoted as their 1st year, 2nd year, 3rd year or 4th year.

Matric_no	Name	Year	Date_Time	Status
101000048	Gina_Gershon	1st	NULL	Absent
101000053	Gong_Li	1st	NULL	Absent
101020009	Allyson_Felix	2nd	24-06-2023 12:25 AM	Present
101020020	Claire_Danes	2nd	NULL	Absent
101020032	Edward_Lu	2nd	23-06-2023 12:15 AM	Present
101020047	Fann_Wong	2nd	NULL	Absent
101020051	Fujio_Cho	2nd	25-06-2023 07:30 PM	Present
101020060	Grady_Irvin_Jr	2nd	NULL	Absent
111020015	Daisy_Fuentes	3rd	NULL	Absent
111020022	Ellen_MacArthur	3rd	22-06-2023 11:35 PM	Present
111020028	Elsa_Zylberstein	3rd	NULL	Absent
111020045	Faisal_Iqbal	3rd	NULL	Absent
111020055	Garry_McCoy	3rd	NULL	Absent
191020878	Choong_Ting_Quan	4th	24-06-2023 12:24 AM	Present
191020903	Lai_Tuck_Yeow	4th	25-06-2023 12:36 AM	Present
191020906	Lee_Qian_Yi	4th	25-06-2023 08:02 PM	Present
191020968	Quah_Xuan_Ying	4th	25-06-2023 08:04 PM	Present
191022310	Swetha_Shanmugam	4th	25-06-2023 07:22 PM	Present
191023027	Sharveni_Ramesh	4th	24-06-2023 12:25 AM	Present
191130607	Lim_Zuan_Huang	4th	25-06-2023 07:44 PM	Present
191392563	Yong_Min_Chiun	4th	25-06-2023 07:32 PM	Present

Fig. 18 *Snapshot of system interface on unrecognized individual*

Additionally, the date and time were noted in the format of DD-MM-YYYY HH:MM AM/PM. By default, no date or time will be recorded for each student. The date and time will be overwritten and updated in the database upon recognition of attendance. The attendance status for each student was set to "Absent" by default. Only students who have been identified by the attendance system can alter their status to "Present".

4. Conclusion

In this study, an attendance system based on face recognition was successfully developed. The proposed system achieved successful face detection using the Viola-Jones algorithm (Haar Cascade classifier) and employed the VGGFace model for accurate face recognition through transfer learning on the VGG16 model. It also enabled real-time attendance marking through a live webcam feed. Due to privacy concerns regarding student images, it was challenging to gather a large dataset from UniMAP students. To address this, the LFW database was utilized to augment the available datasets. Attendance records were securely stored in a MySQL database.

The trained transfer learning model exhibited impressive performance in detecting and recognizing individuals in front of the webcam. While the model evaluation reported an accuracy of 1.0, it is important to note that the face recognition attendance system's accuracy might be slightly lower than 100% during instances of significant facial movements, leading the system to classify individuals as "unknown". The accuracy of face recognition could be affected by the system's GUI, which would encounter significant latency during extended operations.

In summary, the project successfully achieved its objectives by deploying a face recognition attendance system capable of detecting, scanning, and marking attendance for recognized faces. The accuracy of the model was reported to be 1.0, although it should be considered that certain factors, such as facial movements and GUI latency, may slightly affect the system's overall accuracy in face recognition.

4.1 Future Work

For future work, it is recommended to perform improved fine-tuning as the dataset expands. Developing a stable GUI is highly advised to minimize system latency during execution. Furthermore, incorporating additional features, such as generating Excel files after completing attendance or adding remarks for "absent" students, can enhance the accessibility of attendance records and improve the overall attendance-taking process.

Acknowledgement

This project is supported by Higher Education Malaysia (MOHE) under FRGS Project kod: 9003-00904.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Quah Xuan Ying, Raja Abdullah Raja Ahmad;* ***data collection:*** *Quah Xuan Ying;* ***analysis and interpretation of results:*** *Quah Xuan Ying, Raja Abdullah Raja Ahmad, Muhammad Imran Ahmad;* ***draft manuscript preparation:*** *Quah Xuan Ying, Raja Abdullah Raja Ahmad. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Kaspersky. (2020). What Is Biometrics Security. Retrieved from Kaspersky.com website: https://www.kaspersky.com/resource-center/definitions/biometrics.

[2] Gupta, V., Kumar, A., Jawed, A., & Basu, A. (2015). Student Attendance System Based on the Face Recognition of Webcam'S Image of the Classroom. https://www.rcciit.org/students_projects/projects/cse/2018/GR15.pdf.

[3] Sawane, Nakhale, S., Rathod, V., & Ghadge, N. (2021). IJARCCE This work is licensed under a Creative Commons Attribution 4.0 International License Real-Time Smart Attendance System using Face Recognition Techniques. International Journal of Advanced Research in Computer and Communication Engineering, 10(5). https://doi.org/10.17148/IJARCCE.2021.10584.

[4] Ali, A., Mandhanya, R., Birla, S., Mandloi, U., & Jain, V. (2021). Automatic Face Recognition Attendance System using Python and OpenCv. GRD Journal for Engineering, 6. https://www.grdjournals.com/uploads/article/GRDJE/V06/I04/0008/GRDJEV06I040008.pdf.

[5] Sai, E. C., Hussain, S. A., Khaja, S., & Shyam, A. (2021, May 22). Student Attendance Monitoring System Using Face Recognition. Papers.ssrn.com. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3851056.

[6] Bhattacharya, S., Nainala, G. S., Das, P., & Routray, A. (2018). Smart attendance monitoring system (SAMS): A face recognition based attendance system for classroom environment. Proceedings - IEEE 18th International Conference on Advanced Learning Technologies, ICALT 2018, 358–360. https://doi.org/10.1109/ICALT.2018.00090.

[7] Huang, G., Mattar, M., Lee, H., & Learned-Miller, E. (2012). Learning to Align from Scratch. Retrieved from http://vis-www.cs.umass.edu/papers/nips2012_deep_congealing.pdf.

[8] Dwivedi, D. (2018, April 27). Face Detection For Beginners. Retrieved from Towards Data Science website: https://towardsdatascience.com/face-detection-for-beginners-e58e8f21aad9.

[9] Kortli, Y., Jridi, M., Al Falou, A., & Atri, M. (2020). Face Recognition Systems: A Survey. Sensors, 20(2), 342. https://doi.org/10.3390/s20020342

[10] Khiyari, H. E., & Wechsler, H. (2016). Face Recognition across Time Lapse Using Convolutional Neural Networks. Journal of Information Security. Retrieved from https://www.semanticscholar.org/paper/Face-Recognition-across-Time-Lapse-Using-Neural-Khiyari-Wechsler/1c147261f5ab1b8ee0a54021a3168fa191096df8/figure/0.