Face Recognition based on Attendance System

Koh Pei Cong
Institute of Computer Science and Digital Innovation, UCSI University, 56000 Kuala Lumpur, Malaysia

Heshalini Rajagopal
Department of Electrical and Electronics Engineering, MILA University, 71800 Nilai, Negeri Sembilan, Malaysia

Ghassan Saleh Hussein Al-Dharhani
Institute of Computer Science and Digital Innovation, UCSI University, 56000 Kuala Lumpur, Malaysia

Norrima Mokthar
Department of Electrical Engineering, Faculty of Engineering, University of Malaya, Malaysia

Dhanush Rachaveti
School of Electronics Engineering, Vellore Institute of Technology, Chennai, India
E-mail: heshalini@ucsiuniversity.edu.my, heshalini@gmail.com
www.ucsiuniversity.edu.my

Abstract

The use of face recognition technology for attendance tracking has grown popular in recent years. Hence, the main goal of this project is to produce an accurate, fast, and robust face recognition based on an attendance system. The system detects the user's unique features, understand the identity of the user through face recognition technology, and thus records the attendance from the face recognition dataset to the user that matches the user, in an attempt to help the user automatically check in with real-time attendance date. The system includes features such as face detection, face recognition, distance estimation, and attendance recording.

Keywords: Face Recognition, Attendance, Convolutional Neural Network

1. Introduction

A face recognition based on attendance system uses machine learning techniques to identify individuals and track their attendance automatically [1]. It captures an image of the person's face, processes it to extract unique features, and compares it with the face recognition datasets to determine the known faces [2]. If the known face is detected, the system updates his or her attendance record. This technology is commonly used in workplace environments, educational institutions, and other organizations to streamline the attendance-taking process and reduce manual errors [3].

In the 1980s and 1990s, researchers started to use linear algebra for face recognition. This approach was based on the idea that faces could be represented as mathematical objects and compared using linear algebraic operations. The first linear algebra-based face recognition algorithm was introduced by Turk and Pentland in 1991 [4]. They proposed a method called "Eigenface" which used eigenvectors and eigenvalues to represent the face space. This method involved transforming the images into a lower-dimensional space where faces could be represented as points in space. This approach was later improved upon with the introduction of "Fisherface" in 1997 by Belhumeur, Hespanha, and Kriegman [5]. This method used linear discriminant analysis to transform the face space into a space where faces could be separated more easily. This made it possible to use face recognition in a variety of applications, including security and surveillance, border control, and airport security [5].

In the 2000s and 2010s, the development of mobile devices and the widespread availability of the internet paved the way for cloud-based face recognition systems, which allowed for easy access and use from anywhere in the world. This also paved the way for the development of mobile face recognition applications, which could be used on smartphones and other mobile devices. In education, face recognition systems were used for attendance tracking and to enhance student safety [4], [6].

However, many conditions can affect the face recognition attendance system, such as the camera angles, the lighting conditions of the environment, the quality of the image, and various other factors. In addition, the images will get low contrast and low brightness if they are captured under low light conditions [1]. There are several algorithms used by experts to improve face recognition attendance systems, such as the Eigenface algorithm, Principle Component Analysis (PCA), Fisherface algorithm, and Convolutional Neural Networks (CNNs) [5], [7], [8]. The choice of algorithm will depend on the specific requirements of the application. However, CNN is widely used in face recognition attendance systems due to their high accuracy and ability to learn complex features from large amounts of data.

Therefore, this project uses a built-in camera to capture video from the camera, presents automatic attendance tracking, develops a Graphical User Interface(GUI), prevents system warnings and wrong execution from unknown faces, and recognizes faces with high accuracy and fast speed. The expected outcomes aim to ensure that the system can recognize faces under the camera angles, the lighting conditions of the environment, and the quality of the image.

2. Prior Work

This section reviews the methods which were used in those studies to implement face recognition technology and the corresponding to their results.

Shuhui and Xiaochen (2022) compared different SOTA deep face recognition models to extract features from a face image even in the presence of masks. By using data augmentation, the masked face recognition system became more robust to variations in the face images. The result proved that an appropriate margin aided the model's establishment of flexible embedding [9].

Winarno et. al (2019) proposed a combination of CNNs and PCA to identify and verify the attendance of individuals. CNN was used to extract features from a face image while PCA was used to reduce the dimensionality of the data and reduce computational costs. The result showed that the combination of CNNs and PCA can lead to improved performance in face recognition compared to using PCA alone [10].

Cahyono et. al (2020) designed a face recognition system for automating the process of tracking employee attendance. The linear SVM was used to classify the faces of employees in real-time as they entered or left the workplace. K-fold cross-validation was used to evaluate the performance and accuracy of a model. As a result, the combination of FaceNet and SVM achieved 100% accuracy [2].

Qu et. al (2018) compared algorithms, such as PCA, LBP, and CNN algorithms, for handling recognition speed and accuracy. PCA was suitable for fast recognition but with reduced accuracy, while CNN was suitable for high-accuracy recognition but at the cost of computational resources. LBP was a good compromise between the two, offering a balance between recognition speed and accuracy. It was possible to implement a CNN on FPGA due to its high performance and low power consumption [7].

Early work by Ilyas et. al (2019) was concerned with the use of the Viola-Jones algorithm to detect faces in images. Histogram Equalization was used in image processing to enhance the contrast of an image. The result showed that the recognition method using the CMU PIE face database has a higher recognition rate than the Extended Yale B face database [11].

KB and J (2020) designed a real-time face recognition system using CNNs. Viola-Jones algorithm was used to capture real-time input images from a camera. Then, the AT&T database was used to detect faces in each image and crop the image to include only the face. It was possible to simplify the feature extraction process by having all images of equal size [12].

Rahouma and Mahfouz (2021) implemented a face recognition system that leveraged the capabilities of the API mobile vision for face detection. API Google's mobile vision was used to detect faces in the pre-processed images while Pearson correlation was used to compare two face representations and determine their similarity. As a result, the Pearson correlation spent less time but also produced a higher recognition rate than other recognition methods [13].

Sarwar et. al (2021) developed an LBPH-based face recognition system for visually impaired people. Haar Cascade Classifier worked as a pre-processing step to detect faces in an image or video frame. The use of the Euclidean distance helped to determine the similarity between two faces and decide whether they belong to the same person or not. The result showed that LBPH gained higher average accuracy than the Fisherface algorithm [14].

Wang et. al (2020) compared algorithms, such as SGD, RMSprop, and Adam for evaluating the accuracy of the face recognition system. ReLU allowed the network to

model complex features in the input images, such as the shapes of eyes, noses, and mouths. The result proved that CNNs have significant practical application potential when used for face recognition [15].
Early work by Wandzik et. al (2018) was concerned with the use of an SVM classifier to distinguish between real faces and morphed faces. SVMs were able to handle high-dimensional data and can work well with noisy data. The Euclidean distance was used to compare the features of two faces and determined if they were significantly different. The result showed that the performance of CNN-based approaches dramatically increased [16].

3. Method

The face recognition based on attendance system goes through five stages:

(1) Set up the hardware: This involves installing the cameras and other necessary hardware components to capture the images and process them in real-time.
(2) Collect and prepare the training data: A large dataset of face images is collected and processed to extract the features needed for training the face recognition model.
(3) Train the face recognition model: The extracted features are used to train a deep Convolutional Neural Network (CNN) or other machine learning model to recognize faces.
(4) Integrate face recognition system with attendance system: The face recognition model is integrated with the attendance system so that the attendance of individuals can be recorded and tracked in real-time.
(5) Test and evaluate the system: The system is tested on a variety of real-world data to evaluate its accuracy and performance. Any essential modifications are performed to raise the system's efficiency.

Fig. 1 represents two actors: the user and the admin. A user can enter his or her personal information, such as name, id, and course code. This information will be stored in the system. Besides that, users can take face images, train a model, view the result, or play an audio file. If the username matches the result, the system marks user attendance automatically. Users can exit from the program after using the system. On the other hand, an admin can check user information, view the attendance records, and download an attendance list.

Fig. 1. Use Case Diagram

4. Result and discussion

There is an interface design of face recognition based on attendance system, as shown in Fig. 2. It is designed to be simple to use and user-friendly for end users of all ranges.

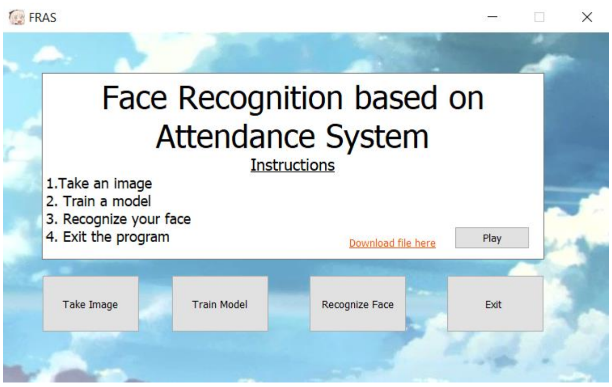

Fig. 2. Graphical User Interface (GUI) using QT Designer

The system includes six main functions, such as:

1. Play button: The user can click the play button to play an audio file. This audio file introduces the instructions should follow to complete all the processes.

2. Take image button: The user can click the take image button to register their faces. This button will bring the user to another window, as shown in Fig.3.

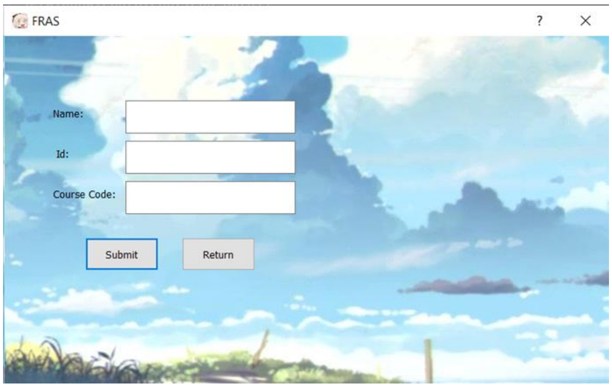

Fig. 3: Take image window

3. Train model button: The user can click the train model button to train a CNN model. The training process runs on the back-end interpreter. Please give a few moments of patience until the window pops up a success message.
4. Recognize face button: The user can click the recognize face button to recognize his or her face. The result shows the name, id, and the distance between an individual and the built-in camera. The system will automatically record the name, attendance date, and time.
5. Download link: The user can click the download file here to select the file to download and select the destination folder to store the file. The system will automatically download files in the destination folder.
6. The user can click the exit button to exit from the program immediately.

The system converts the images to grayscale, equalizes the histogram, and normalizes pixel values between 0 and 1., as shown in Fig. 4.

```
# Step 1: Load and preprocess images
img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)  # Convert to grayscale
img = cv2.equalizeHist(img)  # Standardize the lighting in a image
img = img / 255  # To normalize values between 0 and 1 instead of 0 to 255
```

Fig. 4. Preprocessing images

Next, the system preprocesses the data, creates the model, applies image augmentation, and starts the training loop, as shown in Fig. 5.

```
# Create a CNN model
model = create_model(noOfClasses)
print(model.summary())

# Image augmentation
dataGen = custom_data_generator(X_train, y_train, batch_size_val)

# Start the training process
history = model.fit(dataGen, steps_per_epoch=steps_per_epoch_val, epochs=epochs_val,
                    validation_data=(X_validation, y_validation), shuffle=False)
```

Fig. 5. Train CNN model

After the CNN model is trained, the model is evaluated on the test set to find the test score and test accuracy, as shown in Fig. 6.

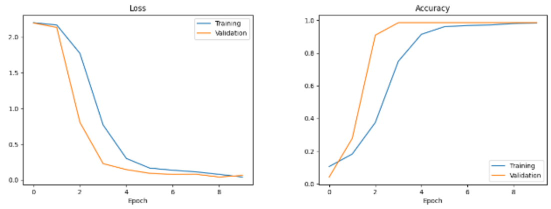

Fig. 6. Test Score(Loss) and Test Accuracy

Then, the system uses focal length calculation to estimate the distance of the recognized face from the camera, as shown in Fig.7.

```
# Calculate the Euclidean distance
distance = KNOWN_FACE_WIDTH * focal_length / w
```

Fig. 7. Focal Length Calculation

After that, the system predicts images from each detected face in the frame, as shown in Fig. 8.

```
# Predict image
predictions = model.predict(img)
classIndex = np.argmax(predictions)
```

Fig. 8. Predict Image

Lastly, the system updates the CSV file every time a face is recognized, as shown in Fig. 9.

	A	B	C	D
1	Id	Name	Date	Time
2	0	kpc	17/7/2023	10:45:01
3	0	kpc	17/7/2023	10:45:01
4	1	boon	17/7/2023	10:48:17
5	0	kpc	17/7/2023	10:45:01
6	0	kpc	17/7/2023	10:45:01
7	1	boon	17/7/2023	10:48:17
8	2	eline	17/7/2023	13:34:29
9	0	kpc	17/7/2023	10:45:01
10	0	kpc	17/7/2023	10:45:01
11	1	boon	17/7/2023	10:48:17
12	0	kpc	17/7/2023	10:45:01
13	0	kpc	17/7/2023	10:45:01
14	1	boon	17/7/2023	10:48:17
15	2	eline	17/7/2023	13:34:29
16	3	roland	17/7/2023	18:17:01

Fig. 9. Attendance records

The testing and evaluation phases are crucial for the developed system. The system is tested on a variety of real-world data to evaluate its accuracy and performance. Any essential modifications are performed to raise the system's efficiency. This project covers unit testing, functional testing, and user acceptance testing.

Unit testing involves testing individual components of the system to ensure they function correctly. In this project, the unit testing helps to check the images are captured successfully, as shown in Fig. 10. Besides that, this testing helps to check the model has been trained successfully, as shown in Fig. 11. Moreover, it helps to check the face recognition done successfully, as shown in Fig. 12.

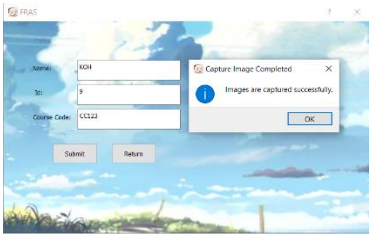

Fig. 10. Capture Image Completed

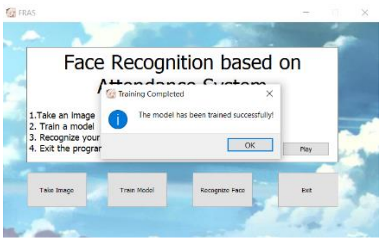

Fig. 11. Training Completed

Fig.12. Testing Completed

Functional testing involves testing the system as a whole to ensure that it meets the specified functional requirements and performs correctly. In this project, the system recognizes known faces and unknown faces, as shown in Fig. 13. Moreover, the system downloads an attendance file in the destination folder.

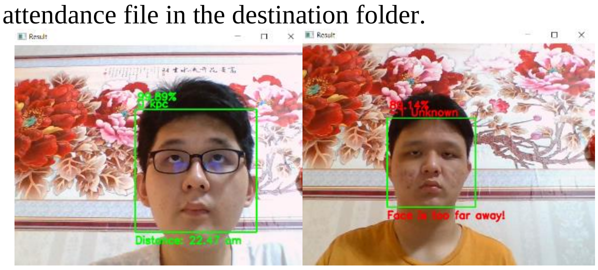

Fig. 13: Known face and Unknown face

5. Conclusion

In conclusion, face recognition based on attendance system provides a functional solution for tracking attendance automatically. The system includes features such as face detection, face recognition, distance estimation, and attendance recording. It exploits a pre-trained Convolutional Neural Network (CNN) model for face recognition. Moreover, the system can recognize faces with high accuracy and fast speed. Furthermore, the system can accurately recognize known faces and unknown faces. In addition, the system stores attendance records in a CSV file, making it easy to manage and track their attendance. Besides that, the system works on a graphical user interface to interact with end users easily.

References

1. K. Alhanaee, M. Alhammadi, N. Almenhali, and M. Shatnawi, "Face recognition smart attendance system using deep transfer learning," in *Procedia Computer Science*, 2021, vol. 192, pp. 4093–4102. doi: 10.1016/j.procs.2021.09.184.
2. F. Cahyono, W. Wirawan, and R. Fuad Rachmadi, "Face recognition system using facenet algorithm for employee presence," in *4th International Conference on Vocational Education and Training, ICOVET 2020*, Sep. 2020, pp. 57–62. doi: 10.1109/ICOVET50258.2020.9229888.
3. H. Yang and X. Han, "Face recognition attendance system based on real-time video processing," IEEE Access, vol. 8, pp. 159143–159150, 2020, doi: 10.1109/ACCESS.2020.3007205.
4. I. Adjabi, A. Ouahabi, A. Benzaoui, and A. Taleb-Ahmed, "Past, present, and future of face recognition: A review," Electronics (Switzerland), vol. 9, no. 8. MDPI AG, pp. 1–53, Aug. 01, 2020. doi: 10.3390/electronics9081188.
5. T. Schenkel, O. Ringhage, A. Márki, and J. Bae, "A Comparative Study Of Facial Recognition Techniques With focus on low computational power Bachelor Degree Project in Information Technology Basic level 30 ECTS Spring term 2019."
6. V. Kumar, G. Jacek, M. Zurada, B. Raman, and G. R. Gangadharan Editors, "Studies in Computational Intelligence 885 Modern Approaches in Machine Learning and Cognitive Science: A Walkthrough Latest Trends in AI."
7. X. Qu, T. Wei, C. Peng, and P. Du, "A Fast Face Recognition System Based on Deep Learning," in *Proceedings - 2018 11th International Symposium on Computational Intelligence and Design, ISCID 2018*, Jul. 2018, vol. 1, pp. 289–292. doi: 10.1109/ISCID.2018.00072.
8. H. Zhi and S. Liu, "Face recognition based on genetic algorithm," *J Vis Commun Image Represent*, vol. 58, pp. 495–502, Jan. 2019, doi: 10.1016/j.jvcir.2018.12.012.
9. W. Shuhui and M. Xiaochen, "Dual-Proxy Modeling for Masked Face Recognition," in *Procedia Computer Science*, 2022, vol. 208, pp. 145–151. doi: 10.1016/j.procs.2022.10.022.
10. Winarno, E., Al Amin, I. H., Februariyanti, H., Adi, P. W., Hadikurniawati, W., & Anwar, M. T. (2019, December). Attendance system based on face recognition

system using cnn-pca method and real-time camera. In 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI) (pp. 301-304). IEEE.

11. Ilyas, B. R., Mohammed, B., Khaled, M., & Miloud, K. (2019, November). Enhanced face recognition system based on deep CNN. In 2019 6th International Conference on Image and Signal Processing and their Applications (ISPA) (pp. 1-6). IEEE.
12. K. B. Pranav and J. Manikandan, "Design and Evaluation of a Real-Time Face Recognition System using Convolutional Neural Networks," in *Procedia Computer Science*, 2020, vol. 171, pp. 1651–1659. doi: 10.1016/j.procs.2020.04.177.
13. K. H. Rahouma and A. Z. Mahfouz, "Design and Implementation of a Face Recognition System Based on API mobile vision and Normalized Features of Still Images," in *Procedia Computer Science*, 2021, vol. 194, pp. 32–44. doi: 10.1016/j.procs.2021.10.057.
14. M. G. Sarwar, A. Dey, and A. Das, "Developing a LBPH-based Face Recognition System for Visually Impaired People," in *2021 1st International Conference on Artificial Intelligence and Data Analytics, CAIDA 2021*, Apr. 2021, pp. 286–289. doi: 10.1109/CAIDA51941.2021.9425275.
15. D. Wang, H. Yu, D. Wang, and G. Li, "Face recognition system based on CNN," in *Proceedings - 2020 International Conference on Computer Information and Big Data Applications, CIBDA 2020*, Apr. 2020, pp. 470–473. doi: 10.1109/CIBDA50819.2020.00111.
16. Wandzik, L., Kaeding, G., & Garcia, R. V. (2018, September). Morphing detection using a general-purpose face recognition system. In 2018 26th European Signal Processing Conference (EUSIPCO) (pp. 1012-1016). IEEE.

Authors Introduction

Koh Pei Cong

He has completed his Bachelor of Science (Hons) in Computing, UCSI University, Malaysia.

Dr. Heshalini Rajagopal

She received her PhD and Master's degree from the Department of Electrical Engineering, University of Malaya, Malaysia in 2021 and 2016, respectively. Her research interest includes image processing, artificial intelligence and machine learning.

Dr. Ghassan Saleh Hussein Al-Dharhani

He is currently an Assistant Professor with the Institute of Computer Science and Digital Innovation, UCSI University, Malaysia. His research interests include Artificial Intelligence, Data Mining and Knowledge Discovery, Machine Learning, Data Science & Analysis

Dr. Norrima Mokhtar

Norrima Mokhtar was appointed as a lecturer to serve the Department of Electrical Engineering, University of Malaya immediately after graduating with her Master of Engineering. She is now serving Department of Electrical Engineering, University of Malaya.

Dr. Dhanush R.

Dr. Dhanush R. is an Associate Professor in the School of Electronics Engineering, VIT Chennai, India. His research interests include Neuroscience, Bio mechanics, Motor learning, Motor Control.