



A real-time attendance system using deep learning face recognition

Dr. Weidong Kuang, University of Texas Rio Grande Valley

Dr. Weidong Kuang received a Ph.D degree in Electrical Engineering at University of Central Florida in 2003. He has been with University of Texas Rio Grande Valley since 2004 starting as an assistant professor, up to an associate professor now. His research interests include VLSI design, machine learning, and digital signal processing.

Mr. ABHIJIT BAUL, The University of Texas Rio Grande Valley

I am a Master's student in the University of Texas Rio Grande Valley. I have completed my Bachelor of Science in Electrical Engineering from Bangladesh University of Engineering and Technology. My research interests are deep learning and computer vision.

A Real-Time Attendance System Using Deep Learning Face Recognition

Weidong Kuang, and Abhijit Baul, University of Texas Rio Grande Valley

Abstract

Attendance check plays an important role in classroom management. Checking attendance by calling names or passing around a sign-in sheet is time-consuming, and especially the latter is open to easy fraud. This paper presents the detailed implementation of a real-time attendance check system based on face recognition and its results. To recognize a student's face, the system must first take and save a picture of the student as a reference in a database. During the attendance check, the web camera takes face pictures for a student to be recognized, and then the computer automatically detects the face and identifies a student name who most likely matches the pictures, and finally an excel file will be updated for attendance record based on the face recognition results. In the system, a pre-trained Haar Cascade model is used to detect faces from web camera video. A FaceNet, which has been trained by minimizing the triplet loss, is used to generate a 128-dimensional encoding for a face image. The similarity between the encodings of two face images determines whether the two face images coming from the same students. The system has been used for a class, and the results are very satisfactory. A survey has been conducted to investigate the pros and cons of the attendance system on college education management.

I. Introduction

Attendance plays a pivotal role in academic performance of students in schools and colleges. The consistent absence from class increases the risk of failure and early dropout. Attendance check is an efficient way to improve the attendance rate of a student in a class. The traditional methods include calling names or signing on papers, which is highly time-consuming and insecure.

To solve the issues associated with traditional attendance check methods, different types of automated attendance systems have been developed using different technologies. In the fingerprint-based attendance system [1], a portable fingerprint device is required to collect and recognize students' fingerprints to mark their attendances. In the RFID-based attendance system [2], students need to present their RFID cards to an ID card reader to record their presences. In the Iris-based attendance system [3], a camera scans the Iris of students, which will be used to match the Iris database of students, and to update the attendance of students. Recently, face-recognition based attendance systems are getting more attentions [4] [5] [6].

In this paper, we proposed and implemented an attendance system based on face recognition using pre-trained deep neural networks. A survey from students in an implemented class has been conducted to investigate the pedagogical impacts and potential problems of the system. The paper is organized as follows. The technical details of the attendance system are described in Section II. The experimental results are summarized in Section III. Section IV concludes the work.

II. The Proposed Attendance System

2.1 Overview of Architecture

The proposed attendance system consists of four components: web camera face capture, student image database, face recognition, and attendance record update, as shown in Fig.1. The system has been implemented in a laptop computer with a built-in web camera. Let's consider a task of checking attendance for a class. To prepare for attendance check, all students in the class are required to take pictures (10 pictures in our project for averaging) through the built-in web camera. These pictures are used to generate the student face database as a reference for real-time face recognition. To check the attendance of a student for the class, the computer takes face pictures of the student through the real-time video stream and employs deep learning neural networks to predict whether the student matches anyone in the database, and (if yes) further identifies the name of the student. The result of this face recognition will be used to update the attendance record in the format of an excel file.

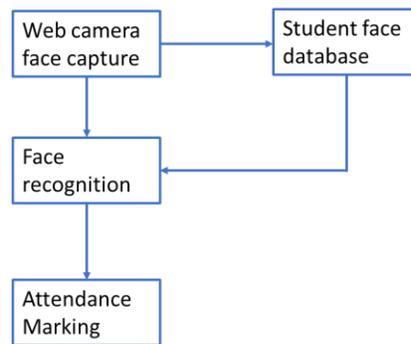


Fig.1 Architecture of the proposed attendance system

2.2 Face Detection

Since our identity recognition is based on a person's face only, it is desirable to save the face part as the image while excluding other parts of a person's body or background. Thus, a face detection technique is required to automatically detect the area of the face so that an image with almost only face can be cropped and saved for later face recognition.

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones in [7]. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. We use Haar Cascade classifier to detect faces in an image. The detected face portion will be resized to a 96x96 image (jpg). The resized image is either saved for database, or processed by a real time face recognition system, which will be discussed in the following sections. The basics of face detection using Haar Cascades can be found at [8]. Fig.2 shows the web camera screen with a detected face frame and saved face image (96x96 pixels).

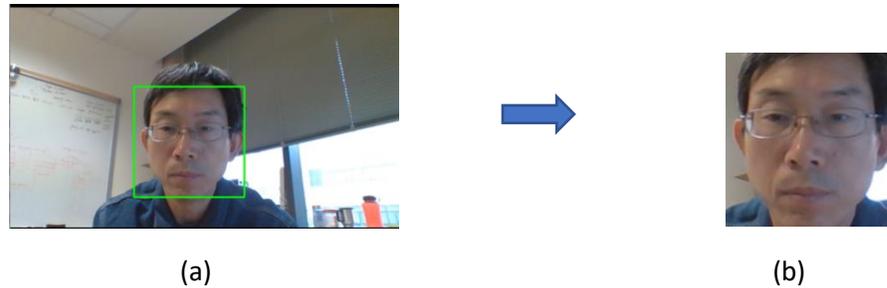


Fig.2 Face detection: (a) video screen with a green rectangle identifying a detected face; (b) cropped face image for database or face recognition.

2.3 Face Recognition

After the face detection, we can assume that the images either for database or for recognition are face-concentrated and resized to 96x96 pixels (e.g. Fig.2(b)). The algorithm of face recognition is illustrated in Fig.3.

The core component of the algorithm is the deep neural network, called FaceNet [9] [10], that directly learns a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of face similarity. Specifically, the FaceNet generates a 128-dimensional encoding (i.e. a 128-element vector) from an image in such a way that: 1) The encodings of two images of the same person are quite close to each other, and 2) the encodings of two images of different persons are far away from each other. Thus, the Euclidean distance between two such encodings is used to determine whether the two face images came from the same person or not. Since training this FaceNet requires a huge amount of data and computation, we directly load the previously trained model, called inception_blocks_v2 [11]. The number of parameters of the resulting model is 3,743,280.

In the database, there is a 128-dimensional encoding for each student in the class. To reduce variations, we took 10 face images for one student, and generated the corresponding 10 encodings, and saved the average of the 10 encodings as the encoding for the student in the database. For each student, his/her name and an encoding are stored in the database, as a *dictionary* object with the name as the key and the encoding as the content in Python.

During attendance check, students are required to show their faces at the front of the web camera one at a time. For one student, the program will automatically capture a face image and generate the corresponding 128-dimensional encoding by the same FaceNet. Then a search algorithm is performed to find an encoding in the database that has smallest distance from this real-time face image encoding, and the associated student name. If the smallest distance is less than a pre-defined threshold (e.g. 0.6), the student has been recognized; otherwise, the student is assumed as “not be in the database”. Of course, the selection of the threshold will affect the recognition accuracy.

To improve the accuracy of face recognition, the program captures multiple face images for one student in real time, and independently recognizes each face image, and generates the result based on these multiple recognitions. For example, in our implementation, the program took 10 face images, followed by independent face recognitions. If among the 10 recognitions, 5 of them are recognized as the

same person, a recognition is assumed to be successful. Of course, the strategy – “5 out of 10” – is adjustable to achieve a better accuracy.

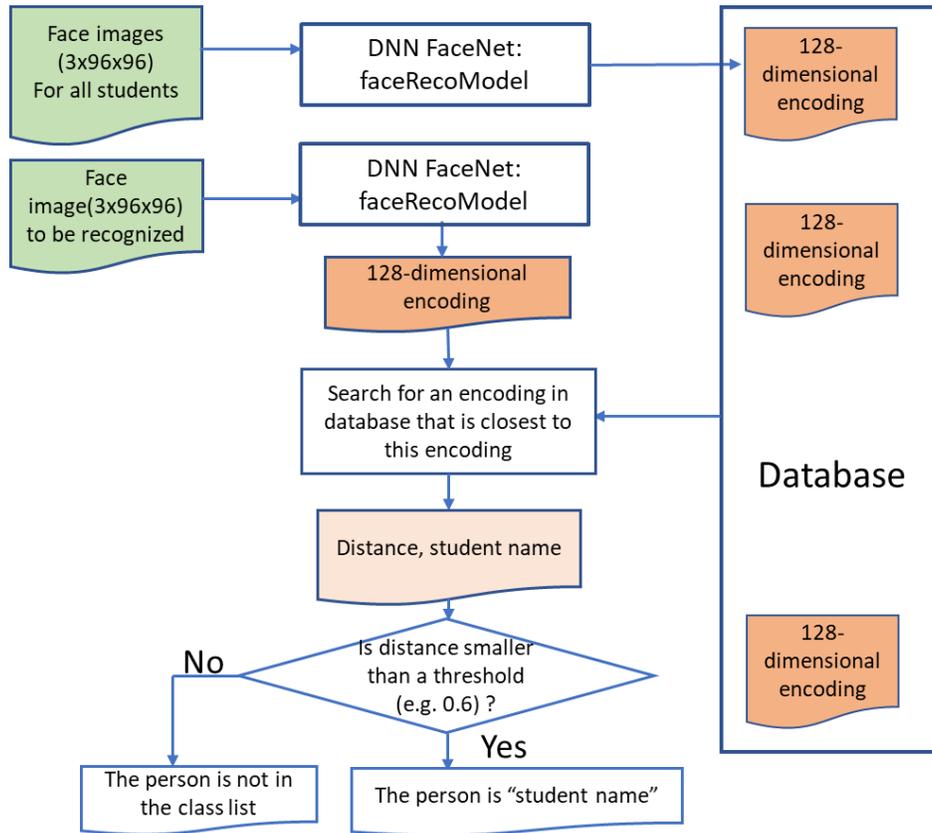


Fig.3 Flow chart for face recognition

2.4 Attendance Marking

The attendances for all students in the class are marked in an excel file, e.g. cmpe3403A.xlsx. The initial file of the excel file is created by filling the first column with student names. An example is shown in Fig.5. Each column (except the first column) is used for attendance record of one class meeting. If a student has been recognized more than 5 times out of 10 trials, the exact numbers of recognitions will be recorded in a corresponding cell that is associated with the student (row) and the date of the class (column). Thus, any student, who has a number (more than 5) recorded, is considered as attending the class, while a student without a number recorded will be considered as absence.

In case of a failure of face recognition for a student, i.e. the student is recognized as another student in the class, or the student cannot be recognized as any student in the class, the student should notice the instructor the situation for a manual attendance check in the excel file.

III. Results of Experiments

The attendance system has been implemented in a Lenovo laptop computer with the following specifications: Intel® core™ i5-7200U CPU @2.50GHz, RAM 8.00GB, Windows 10 Pro. The system consists of three programs developed in Python 3. The first program captures 10

face pictures for each student in the class. The second one compiles FaceNet model and loads parameters. The third one generates the database based on the pictures captured by the first program, recognizes the face at the real-time web camera, and updates the attendance record in the excel file. Thus, all visible results are delivered by the third program.

Fig.4 is a screenshot that shows the result of a face recognition. The information of the result includes the frame identifying the location of the detected face, the recognized name (Will) at the up left corner of the frame, and the corresponding distance (0.4891317) to Will’s face encoding in the database. This distance gives a confidence on how surely the person is “Will”: the smaller it is, the more likely he/she is “Will”. The bigger word “Will” at the top center is the result based on the 10 consecutive recognitions. Fig.5 shows a portion of attendance record in an excel file created by the attendance system. In Fig.5, the first column is for students’ names and the second is the attendance record. The number for a student is the number of recognitions among 10 trials. A blank for a student means that either the student did not check the attendance, or the number of recognitions is less than 5 (i.e. face recognition failed). In case of the failure of recognition, the student needs to inform the instructor for a manual attendance marking.

The test results in one class with 28 students showed that the accuracy of face recognition is about 95%. However, the attendance system is sensitive to the changes of lightening and the distance between face and camera. Dramatical change of these conditions can significantly reduce the accuracy.



Fig.4 Face recognition window

	A	B	C
1	name	11/07/19	
2	acevedo	8	
3	barrientos	6	
4	carrillo		
5	cisneros	9	
6	delaantonio	8	
7	delaorlando	6	
8	deleo		
9	diaz	7	
10	elizondo		
11	escobedo	5	
12	garcia	7	
13	gonzalez		
14	gutierrez	7	
15	lagare	9	
16	lara		
17	lozano	7	
18	mendez	8	
19	muniz	9	
20	ochoa	6	
21	pena		
22	ruiz	7	
23	salinas		

Fig.5 Example of attendance marking

To evaluate the pedagogical benefits or concerns of the attendance system, we conducted a survey on 21 students. On each survey there are 12 statements as follows:

- 1) Using this system was enjoyable.
- 2) This system is more efficient than attendance check by calling names.
- 3) This system is more efficient than attendance check by sign-sheet.
- 4) This system is more efficient than attendance check by doing quiz.

- 5) This system makes the attendance check easier for the instructor.
- 6) This a room for the improvement of this system.
- 7) This system should be used in other classes.
- 8) I don't like this system because I have a concern on my privacy.
- 9) This system is suitable for all people including disabled (deaf).
- 10) This system exposed me to a real AI project.
- 11) This system inspires me to learn machine learning in the future.
- 12) If I were the instructor in college, I would use the face recognition for attendance.

For each statement, there are the following five options for students to select:

- strongly agree agree do not know disagree strongly disagree

To quantize the overall opinions of students on each statement, we assign point numbers 2, 1, 0, -1, -2 for the above five options, respectively. The total point of a statement is the weighted sum of the selected options. The larger the point, the more strongly the students agree. A negative point shows disagree. For example, the first statement received 9 *strongly agree*, 11 *agree*, and 1 *do not know*. Thus, the point of the first statement is 29. The result of survey is visualized in Fig.6. The survey confirms the benefits of the attendance system such as efficiency and convenience. As a demo, the attendance system exposed the AI technology to students. The survey also shows that some students have privacy concerns and that the system could be improved further.

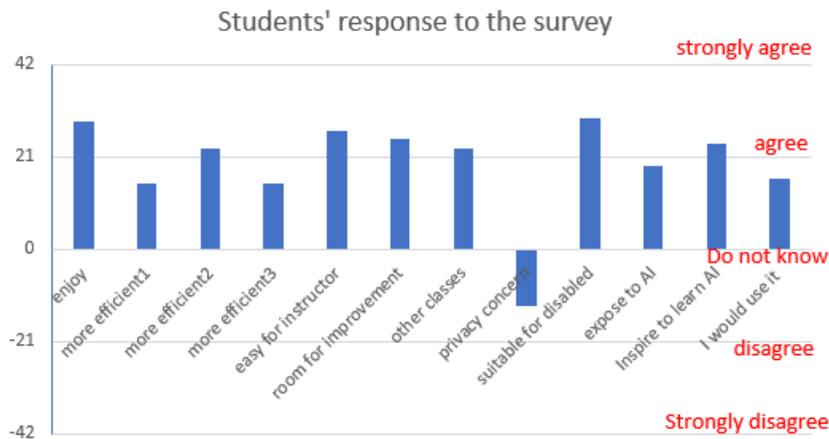


Fig. 6 A survey on the attendance system

IV. Conclusion

An attendance system has been implemented in a laptop computer using face recognition. The system was applied in a class for attendance checking. The accuracy of face recognition for a class of 28 students is about 95% if a consistently similar condition (e.g. light, face distance and face expression) for image capture is maintained. The survey from students shows that the system is effective and efficient, and accepted by students.

However, a few aspects for improvement can be pursued in the future work. First, the current system consists of three Python programs (or files), and the operation of the system requires

some command line inputs. It would be more convenient for a user (e.g. instructor) to use the system if the system could be integrated into a single application file with a Graphic User Interface (GUI). Second, we will update the system for a more robust face recognition if a corresponding algorithm is available. Third, based on the survey, some students have a privacy concern on face recognition. It is important to address this concern.

References

- [1] B.K. Mohamed and C. Raghu, "Fingerprint attendance system for classroom needs," India Conference (INDICON), Annual IEEE, 2012, pp. 433-438.
- [2] S. N. Shah and A. Abuzneid, "IoT based smart attendance system (SAS) using RFID," IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2019.
- [3] A. Khatun, A.K.M.Fazlul Haque, S. Ahmed, and M. M. Rahman, "Design and implementation of Iris recognition based attendance management system," 2nd Int'l Conf. on Electrical Engineering and Information & communication Technology (ICEEICT) 2015, Bangladesh.
- [4] S.Sawhney, K.Kacker, S. Jain, S. N.Singh, and R.Garg, "Real-time smart attendance system using face recognition techniques," 9th Int'l Conf. on Cloud Computing, Data Science & Engineering, 2019, pp. 522-525.
- [5] S.Sveleba, I. Katerynychuk, I.Karpa, I.Kunyo, S.Ugryn, and V.Ugryn, "The real time face recognition," 3rd Int'l Conf. on Advanced Information and Communication Technologies, 2019, pp.294-297.
- [6] R.Nandhini, N. Duraimurugan, and S.P.Chokkalingam, "Face recognition based attendance system," Int'l Journal of Engineering and Advanced Technology (IJEAT), Vol-8, Issue-3S, February 2019, pp.574-577.
- [7] P. Viola, and M. Jones, "Rapid object detection using a boosted cascade of simple features", Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001.
- [8] https://docs.opencv.org/3.3.0/d7/d8b/tutorial_py_face_detection.html .
- [9] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," Proceedings of the 2015 IEEE computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2015, also available at <https://arxiv.org/pdf/1503.03832.pdf>
- [10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," Proceedings of the 2015 IEEE computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2015, also available at: <https://arxiv.org/pdf/1409.4842.pdf>
- [11] <https://github.com/Skuldur/facenet-face-recognition>