

ClassPic: Automated Attendance Checking Web Application using Facial Recognition

Paolo Miguel R. Astudillo and Rizza DC. Mercado

Abstract— Attendance checking is an essential part of classroom management but it can be time-consuming and tedious. Thus, many technologies are developed in order to automate this task. These technologies are often expensive because they require machinery to operate. This paper presents a more economical and efficient method of automating attendance checking by utilizing facial recognition. Using the developed web application, the teacher can take images of the class containing the face of each student then the application will automatically record the in a .csv file the attendance based on the faces recognized.

Index Terms— Attendance Checking, Facial Recognition

I. INTRODUCTION

A. Background of the Study

All academic institutions have certain policies on class attendance. The importance of attendance in class participation is evident. A student cannot be involved in school works if he or she does not come to school. Thus, attendance checking is an essential aspect of classroom management, it makes sure that students maximize their potential for learning by attending every single class [1]; but the process of attendance checking can be tedious and time-consuming. It can also be a source of inaccuracy and cheating. The status quo of most educational institutions in attendance checking is through paper and pen, either by giving a quiz or passing around sheets of paper for students to sign which can also be lost, cheated, or stolen.

Automating attendance checking can increase the efficiency of the class. Instead of manually checking the attendance especially for large classes, the teacher can allocate the saved time into fulfilling other class agenda. It can also generate a more reliable and accurate attendance sheet compared to the traditional method of attendance checking. Existing technologies that automates attendance checking include Geolocation tracking, Radio Frequency Identification (RFID), and Quick Response (QR) code, which mostly require expensive gadgetry and are prone to cheating.

Facial recognition is a recognition technique used to detect faces of individuals whose images are stored in a database. It uses algorithms to pick out specific, distinctive details about a persons face. These details are then converted into mathematical representation and then compared to other faces

Presented to the Faculty of the Institute of Computer Science, University of the Philippines Los Baños in partial fulfillment of the requirements for the Degree of Bachelor of Science in Computer Science

in the database [2]. Today, facial recognition technologies are widely used for various sectors like security, health care and marketing.

In this day and age, there are companies and schools that have a multitude of students consisting of hundreds to thousands or more. To handle this large group of people more efficiently without sacrificing the accuracy, a more appropriate system that is more economical, practical, and precise, and which utilizes facial recognition and mobile application based technology must be developed.

B. Objectives of the Study

The general objective of the study is to implement a web application that utilizes facial recognition which will be used to automate attendance checking. It specifically aims to:

- 1) Create a web application for teachers to easily manage the attendance of a class;
- 2) Generate a spreadsheet that contains the attendance record using the web application; and
- 3) Assess the overall performance of the developed application.

II. RELATED STUDIES

There are various existing methods that have already come out and automated class attendance but in one way or another is still lacking in upholding an integral principle it could be affordability, convenience, or even security.

Radio-Frequency Identification (RFID) is the use of radio waves to read and capture information stored on a tag attached to an object the tag does not need to be in contact for the information to be read. This system calls for installation of expensive equipment across the entire campus to make sure that for every classroom or lecture hall there will be a RFID reader causing the school to put the burden on the students and raise the tuition fee higher. [3]

Geolocation is when a device uses Wi-Fi or Ethernet connection to identify the geographical information. An example of an application that checks attendance is Jibble it

when a staff or student can claim to be at a certain place, all they have to do is activate the Geolocation power-up then install the application on their phone then clock in or out with their location [4]. A problem with this is that although the system is cheaper and more portable there is no assurance that the student or staff is the one who is holding their device as it can easily be given to a friend or colleague in this light there will be problems with the integrity of the data collected over time.

Quick Response (QR) code is a machine-readable code that is a trademark for a type of matrix barcode first designed in 1994 for the automotive industry in Japan. In a study examining the QR code system for attendance checking, a QR code is displayed in a slide in the class screen then students must scan at the beginning of each lecture to confirm their attendance [5]. This method is once again assuming that all students have a personal device.

There is also existing an technology that utilizes facial recognition one of which is closed-circuit television (CCTV) is a TV system in which signals are not publicly distributed but are monitored, primarily for surveillance and security purposes. CCTV relies on strategic placement of cameras, and observation of the camera's input on monitors somewhere [6]. In a study done by a group of Tokyo Techies Student Research Project they were able to create an attendance checker using OpenFace. They utilized machine learning approach to build a classification model to identify a persons name from the unique facial features of an individual subject [7]. The problem using this system is that a lot of CCTV need to be installed in every classroom which is expensive.

Another method that was explored was an attendance checker using facial recognition for PC platform only, In a study conducted in India a team of researchers explored the possibility of using an automatic attendance system using image processing. First a video clip of the entire classroom is taken which is stored into a database then converted into frames and images. This is where Face detection techniques such as Adaboost algorithm to detect the faces in frames/images and then features are extracted of detected face by Histogram of Oriented Gradients (HOG) and Local Binary Pattern (LBP) algorithm. After storing the faces of the students in the database. The detected faces are compared with the faces stored in the database during face recognition by using Support Vector Machine (SVM) classifier. Attendance gets marked after matching the recognized face [8]. This however uses a computer as its platform, although it is not a specialized equipment, it does not have the portability of a phone which makes it difficult to implement in every lecture hall in a university.

III. METHODOLOGY

A. Image Acquisition

Due to the COVID-19 pandemic, the web application was tested not in a classroom setting but in a household of 10

people.

The class pictures were shot using the 12MP Main Camera of Samsung S9+ while the student selfies were shot using the 8MP Front Camera of Samsung S9+. In testing the accuracy of the program, the students were asked to arrange themselves randomly and also by their height. The class pictures were shot in different distances such as 3 meters away and the farthest distance possible in the house which was 13 meters and in an angle that can clearly capture all the faces of the students.

B. Architecture of the Application

Figure 1 presents the architecture of the application. It consists of four major components which are the Users, User Interface, Application Server and Kairos API//

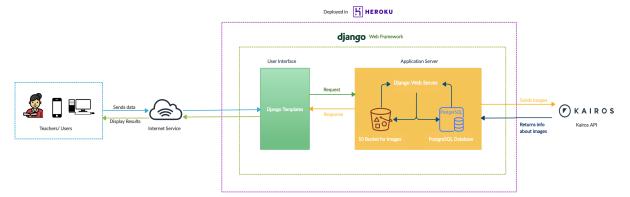


Fig. 1: Architecture of the Application

1) Users

The application has only one type of user which is the teacher. The teacher has access to the following functionalities:

- a) **Class Management:** the user may add, edit, or delete a subject;
- b) **Student Management:** the user may add, edit or delete a student belonging to a subject;
- c) **Attendance Management:** the user may view and delete the attendance record;
- d) **Take ClassPic:** the user may check the attendance of students by uploading a class picture; and
- e) **Export Attendance:** the user may export the attendance record of a subject in a .csv file.

2) User-Interface

The User Interface (UI) of the web application serves as the portal for the user to interact with the web application. This is where the user can manage their subjects, students, and attendances. This was developed using Django Templates which uses HTML and CSS. In order to improve the appearance of the UI, Bootstrap4 was also integrated.

3) Application Server

The Application Server houses all the data needed for the web application. This server is responsible for data storage and executing the different processes related to the management of subjects, students and attendances. This is

also where the process related to facial recognition is located, which is done through an API call from the application server to Kairos API. Lastly, this is where the process for creating a .csv file based on the attendances is also located. The application server is a Django Web Server which has a PostgreSQL database to store the models of the subject, students and attendances and an Amazon S3 bucket in order to securely store the images that were used for this web application.

4) Kairos API

Kairos API is a Facial Recognition API where a user submits images into the API, and its deep learning algorithms analyze the faces found, then the API returns a bunch of useful data about the faces found. The data returned can be used to search, match and compare faces, or measure characteristics such as age, and gender. Kairos API was the chosen Facial Recognition API because of its ease of use, accuracy, security and ethics. Its accuracy scores at 99.63% on the popular LFW database. It has a high level of security because of the High level encryption and tokenization of sensitive data. Lastly, it values ethics because it does not allow government agencies to use its API.

C. Generation of Face Templates

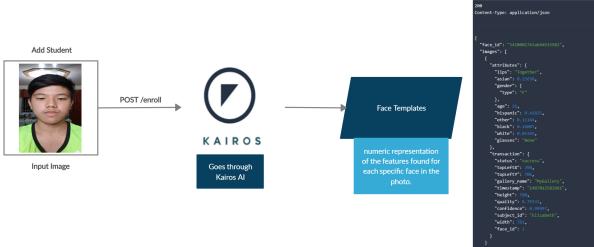


Fig. 2: Overview of Generation of Face Templates

Using the web application, when a student is added it needs to upload a selfie or a self-portrait. This image alongside the information of the student is sent to the Kairos API /enroll path where the image goes through Kairos AI in order to generate a face template which contains the information extracted based on the face uploaded such as age, gender and race. This face template is then saved to a Kairos API gallery which contains the other face templates of other students belonging to a subject.

D. Attendance Checking and Recording

In order to take a ClassPic or Attendance, the user needs to upload an image containing the picture of the class. The image is sent to Kairos API /recognize where each face in the image is detected and matched against the face templates saved to a Kairos API gallery. If a face in the uploaded image is greater than a set threshold, it is added to the list of potential candidates and then Kairos API returns the list

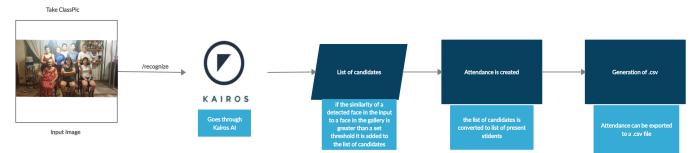


Fig. 3: Overview of Attendance Checking and Recording

of potential candidates and how closely they match. The list of potential candidates is then sent to the Attendance entity where the list of potential candidates are converted to the list of present students on the time the ClassPic was taken.

IV. RESULTS AND DISCUSSION

A. Web Application

A web application has been developed that automates attendance checking with the use of facial recognition. The User Interface for the web application developed is shown below.

1) Login Page

This page is the landing page of the web application. It shows the login form and the link to the registration. It uses Django's built-in authentication system in order to verify the credentials given. A session is generated for every successful login. The session framework lets you store and retrieve arbitrary data on a per-site-visitor basis. It stores data on the server side and abstracts the sending and receiving of cookies. Cookies contain a session ID not the data itself.

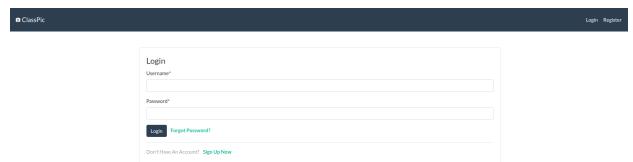


Fig. 4: Login Page

2) Registration Page

This page is for the users who want access to the web application. It generates credentials for the user to use in the login page. It also uses Django's built-in authentication system which generates a User object for every successful registration. In Django, passwords use the PBKDF2 algorithm with a SHA256 hash, a password stretching mechanism recommended by National Institute of Standards and Technology (NIST) which means it is secure and requires amounts of computing time to break.

3) Profile Page

This page enables the user to change their username and email.

Fig. 5: Register Page

Fig. 6: Profile Page

4) Forgot Password Page

This page allows users to reset their password by providing the correct user email. The Web application sends an email with a link containing the form that allows users to change their password.

Fig. 7: Forgot Password Page

Password reset on www.classpic.dev [Inbox](#)

 pmrastudillo@gmail.com
to me

You're receiving this email because you requested a password reset for your user account at www.classpic.dev.
Please go to the following page and choose a new password:
<https://www.classpic.dev/password-reset-confirm/MQ9gt-33btds0064ad0a4477/>
Your username, in case you've forgotten: paulo
Thanks for using our site!
The www.classpic.dev team

[Reply](#) [Forward](#)

Fig. 8: Forgot Password Email

5) Dashboard Page

This page is where the user is taken after a successful login. It shows all the created subjects by the user and a button that enables the user to create another subject. Each subject is displayed with their title, description, schedule, location and the number of students in the subject.

6) Subject Detail Page

This page appears when the title of the subject is clicked in the dashboard. It shows information about a subject. This page

Fig. 9: Dashboard Page

enables the user to create an attendance, view the attendance of the subject and add a student to the subject. This page also shows the class list which is a list to show the students belonging to a subject. It also enables the user to edit or delete a student by clicking the buttons near a student's name.

Fig. 10: Subject Detail Page

7) Take ClassPic/ Create Attendance Page

This page enables the user to create an attendance by taking a class picture. It can take up to two class pictures. It shows the guidelines in order to increase the accuracy of the facial recognition.

Fig. 11: Take ClassPic/ Create Attendance Page

8) Attendance List Page

After an attendance is taken, it redirects to this page. It shows all of the attendances taken with the corresponding date and time . It also allows the user to delete an attendance. This page enables the user to convert all of the attendances into a .csv file by using the export to csv button.

The screenshot shows a web application interface titled "Attendance List". At the top, there is a header with the class name "CMSC 123 CD", the date "May 25, 2020", and a "Logout" button. Below the header is a table with columns for "Date/Time" and "Attendance". The table lists ten rows of data, each representing an attendance entry for May 25, 2020, at various times between 02:31 AM and 02:59 AM. Each row has a red square checkbox next to it. At the bottom of the table, there are three small blue circular icons.

Fig. 12: Attendance List Page

9) Attendance Detail Page

This page shows all of the class pictures taken which can be viewed when discrepancies are encountered, the list of present and absent students, and the number of present students over the number of all students in a subject.

The screenshot shows a web application interface titled "Attendance for May-25-2020 02:31 AM". At the top, there is a header with the class name "CMSC 123 CD", the date "May 25, 2020", and a "Logout" button. Below the header is a large thumbnail image of a group of people sitting around a table. Below the thumbnail, there is a table with two columns: "Present" and "Absent". The "Present" column lists eight student names with their corresponding profile pictures. The "Absent" column is currently empty. At the bottom right of the table, it says "8/10".

Fig. 13: Attendance Detail Page

B. Attendance Checking

The web application was given different scenarios to test the accuracy of the automated attendance checking. Figure 15 shows some of the male students with new haircuts and even new trim of facial hairs compared to their appearance in Figure 14. It shows that facial hair or even change in hairstyle does not affect the accuracy of the automated attendance checking. They also show that wearing eyeglasses does not affect the accuracy of the automated attendance checking as can be observed by the man with the eyeglasses in Figure 14 which he removed in Figure 15. Figure 15 shows that even when there is a sit-in or unenrolled student, it does not detect that person and takes the attendance according to the class list. The web application can also take up to two images and combine those into a single attendance entity which can be useful in large classes that are too large to fit in a single picture which can be observed in Figure 16.



Fig. 14: No. of students=10, Automated Count=10



Fig. 15: No. of students=10 + 1 Sit-in, Automated Count=10



(a) Left side of the room:
No. of students=6



(b) Right side of the room:
No. of students=4+1 sit-in

Fig. 16: No. of students=10 + 1 Sit-in , Automated Count=10



Fig. 17: No. of students=10 , Automated Count=6

Randomized and according to height seating arrangements were tested. Table 1 shows the performance of the system. The second column refers to the correct count of the students, the third column refers to the automated count generated by the web application and the fourth column refers to the Accuracy of the system which declares the correctness of the Automated Count compared to the correct number of students. The first five images tested are in a randomized seating arrangement meanwhile the last five images were tested in according to height seating arrangement. The low accuracy in Image 4 which can be seen in Figure 17 is caused by some students faces not visible because it was blocked by the student in front of them which are taller than them. There were no false positives recorded which means that there is no absent student recorded as present. Therefore, if according to height seating arrangement were implemented in a room the web application performs better. On the average, the web applications accuracy is satisfactory with 94% accuracy.

TABLE I: Overall System Performance

Test Image	No of students	Automated Count	Accuracy
1	10	10	100%
2	10	10	100%
3	10	9	90%
4	10	6	60%
5	8	8	100%
6	10 + 1 sit-in	10	100%
7	10	10	100%
8	10	9	90%
9	8	8	100%
10	8	8	100%
AVERAGE			94%

C. Performance Issues

Due to time request limitations caused by Heroku a request which is greater than 30 seconds throws back a request timeout error. The web application requires an image upload when a class picture is taken which means if a user has slow internet connection the website will throw back a request timeout error and the taking of attendance cannot be executed. This error can also be encountered when attempting to upload two images in a take attendance form. This error can be caused by slow upload speed and due to the fact that Kairos

API /recognize can only cater one image per request.

V. CONCLUSIONS AND RECOMMENDATIONS

Attendance checking is an essential aspect of classroom management, it helps students maximize their potential for learning by attending every single class however attendance checking can be tedious, time-consuming and can be a source of discrepancies. Thus efforts to automate attendance checking are developed. These efforts are often expensive, because they require certain machinery. Thus, a web application was developed to make attendance checking more efficient, reliable and cost-effective. The accuracy of the web application in automating attendance checking was satisfactory given that the class follows an according to height seating arrangement because it improves the visibility of the faces.. Future works can include Using another API or developing a new facial recognition algorithm to improve accuracy of the attendance checking, fixed phone camera mounted on the top that can take images on a set time in order to take out the hassle of taking the time to take the pictures of the class and can even detect late students. Future works can also include integration to existing class management portals such as Google Classroom and Edmodo.

REFERENCES

- [1] B. Laws, "To research the correlation between school attendance and student," 2013.
- [2] Facial recognition. Techopedia. [Online]. Available: <https://www.techopedia.com/definition/32071/facial-recognition>
- [3] Why are students faking attendance? they feel cheated by the system. The Guardian. [Online]. Available: <https://www.theguardian.com/higher-education-network/2018/apr/13/why-are-students-faking-attendance-they-feel-cheated-by-the-system>
- [4] (2017) Jibble review. FinancesOnline. [Online]. Available: <https://reviews.financesonline.com/p/jibble/>
- [5] F. Masalha and N. Hirzallah, "A students attendance system using qr code," 2014.
- [6] Cctv (closed circuit television). WhatIs.com. [Online]. Available: <https://whatismytechtarget.com/definition/CCTV-closed-circuit-television>
- [7] (2017) Development of facial recognition algorithm for marking attendance. Tokyo Techies. [Online]. Available: <https://tokyotechies.com/using-facial-recognition-technology-smart-attendance-application/>
- [8] A. Ahmedi and S. Nandyal, "An automatic attendance system using image processing," 2015.