

Automatic Attendance System for University Student Using Face Recognition Based on Deep Learning

Tata Sutabri, Pamungkur, Ade Kurniawan, and Raymond Erz Saragih

Abstract—Student attendance is essential in the learning process. To record student attendance, several ways can be done; one of them is through student signatures. The process has several shortcomings, such as requiring a long time to make attendance; the attendance paper is lost, the administration must enter attendance data one by one into the computer. To overcome this, the paper proposed a web-based student attendance system that uses face recognition. In the proposed system, Convolutional Neural Network (CNN) is used to detect faces in images, deep metric learning is used to produce facial embedding, and K-NN is used to classify student's faces. Thus, the computer can recognize faces. From the experiments conducted, the system was able to recognize the faces of students who did attend and their attendance data was automatically saved. Thus, the university administration is alleviated in recording attendance data.

Index Terms—Student attendance system, convolutional neural network, deep metric learning, K-nearest neighbor.

I. INTRODUCTION

Student attendance is an essential aspect of the learning process on the university. By attending class, student able to get valuable information from the lecturer, so that the student able to improve knowledge and understanding towards a particular field or even some skills [1]. Each university is implementing its attendance system to make record student's presence for tracking and administration purpose. The most common attendance record in Indonesia still using a manual approach. Below are two common ways for presence record can be found on today universities:

1) Lecturers call students one by one and record into attendance paper.

2) Students sign attendance paper on their own.

Of the two attendance record approaches, there are several shortcomings, including:

- It takes a long time to call the names of students one by one.
- Student can easily falsify their friends' signatures.
- Attendance record paper can be easily lost if not properly stored and managed by the university administration.

Manuscript received January 11, 2019; revised June 8, 2019.

Tata Sutabri is with the Department of Information Systems, Faculty Information Technology, Universitas Respati Indonesia (e-mail: tata.sutabri@gmail.com).

Pamungkur is with the Department of Economic Development, Sekolah Tinggi Ilmu Ekonomi Kuala Kapuas, Central Kalimantan, Indonesia (e-mail: pamungkur@yahoo.com).

Ade Kurniawan and Raymond Erz Saragih are with the Department of Informatics Engineering, Universal University, Batam, Kepulauan Riau, Indonesia (e-mail: ade.kurniawan@uvers.ac.id, raymonde.saragih@gmail.com).

- Additional work is needed to enter attendance data into the database.

To overcome the above problems, a solution is needed to automate the attendance process.

In this paper, a university student attendance system with face recognition is proposed. Machine learning algorithm like Dlib [2] is used to convert face image into 128-d embedding, and K-NN for face classification. By testing the attendance system, a student's face is successfully detected and recognised; then the attendance data is automatically recorded into the system, which consists of student ID number, date, and time. With this new system, the old manual attendance approach can be replaced.

II. BASIC THEORY

A. Face Detection

Face detection is a process carried out by a computer to detect faces in an input image. Algorithms that currently exist were used to detect frontal face [3]. Face detection technology has developed through the discovery of new and relatively faster algorithms that make computers capable of detecting faces in an image. The following are some of the algorithms used to perform face detection in an image:

1) Viola-jones

Viola-Jones was proposed by Paul Viola and Michael Jones in 2001 [4]. Viola-Jones is based on object detection, but its main application is for face detection [5], [6]. The detection rate of the Viola-Jones algorithm is high (true-positive level), and the false-positive rate is low and could detect face rapidly [6]. Although this method has a drawback, the training for this system is slow and less effective on non-frontal face [7], [8]. For face detection, the Viola-Jones algorithm goes through three main stages [5]-[7]:

- Computing Integral Image

This stage is used to convert the image into an integral image [6]. An integral image is a concept of Summed-Area Table, that is used to compute the sum of values in a subset of rectangular boxes. The integral image that is located at (x, y) is the sum result of pixels located above it, and pixels located to its left [7]. By creating an integral image, computation for Haar features can be done rapidly [6].

- Adaboost Algorithm

AdaBoost is one of a machine learning algorithm that is used for detecting face [5]. Classifiers are created by selecting a few essential features computed in the previous stage [6]. An AdaBoost algorithm is used to select these original features and train classifiers that would be using those features [7]. The AdaBoost algorithm is aiming to construct a robust classifier

from the linear combination of weak classifier [6].

- Cascading Classifiers

In this process, classifiers are combined in order to increase the detector's speed by focusing on face regions. This works in a way that the initial classifiers are more straightforward and are used to remove most rejected sub-windows and finally get multiple classifiers that can achieve a low false positive level [6].

2) Eigenface

Eigenface was introduced in 1987 by Sirovich and Kirby. When an image is used as input, the image has a lot of noise, such as poses, background colours, light effects, etc. However, all the images that contain face have several patterns that appear. These patterns are facial features. Facial features of a face are mouth, nose, and eyes and the distance between them. These facial features are referred to as "eigenface" or the principal components in general. In order to extract these facial features, a mathematical method called Principal Component Analysis (PCA) is used [3].

3) Neural network

Neural Network is inspired by the human brain, which consists of neurons or perceptron that are connected in several same or different layers. Neural Network is self-learning, and this can be achieved by training it. In case of face detection, neural network scans every matrix in an input image, to determine the existence of face. This approach is considered efficient because there is no need to train images which have no face in it. The process of face detection is divided into two. The first step is to use an area of the image as an input for the filter that made up of the neural network. The result of the filter is an array of -1 or 1, which represents the absence or presence of a face in the image. The second step is to omit false detection in the first step in order to get a better result. To achieve this, all overlapping detection are combined [9].

B. Face Recognition

Face recognition is a technique that is used to recognise a person from his/her face that has been previously trained from a dataset [10]. Face recognition is one of the most efficient biometric techniques to identify a person [11], and has advantages compared to other biometric methods, such as identifying could be done without requiring action from the user, it has non-intrusive characteristics [12]. Face recognition is a technology that can be applied in various fields, such as surveillance, smart cards, entertainment, law enforcement, information security, image database investigation, civilian applications, human-computer interactions [13].

Face recognition used digital image or video as an input and data of the person that appears in the image or video processed as an output [13]. The face recognition process can be divided into two parts, the first part is image processing, which consists of obtaining facial images through scanning, image quality improvement, image cropping, image filtering, edge detection, and extracting features in images. The second part is a recognition technique consisting of artificial intelligence compiled by genetic algorithms and other approaches for facial recognition [10]. To summarise the process of the face recognition system, there are three steps taken; those are face detection, feature extraction, and classification, as shown in Fig. 1 [13], [14].

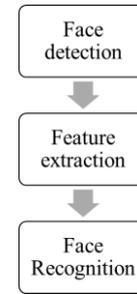


Fig. 1. Process in face recognition.

The first step taken in face recognition is face detection. As explained above, face detection is the process of which computer searching for a face-like object in an input image. Face detection's objective is to determine the existence of a face in the image. If the face exists, the output will be the location of the face and its extent. The next step is to detect facial feature and extract it. Facial features are such as eyes, nose, mouth, eyebrow, ears and chin. The last step is to recognise the face by comparing the output with the database [14].

Face recognition itself can be divided based on its approach. Those approaches are the feature-based approach, holistic-based approach, and hybrid-based approach [12], [14]. All three are explained below.

1) Feature-based approach

A feature-based approach to input image processing to identify and extract unique features in the face, such as eyes, mouth, nose and so on, then calculating the geometric relationship between the points of the face, so that the input face image is converted into a geometric feature vector [12]. Feature-based itself is divided into Geometric feature-based matching or template based, and elastic bunch graph. The geometric feature analyses facial features and their geometric relation. Elastic bunch graph is a technique based on dynamic link structures. For each face, a graph is generated by using fiducial points, with each fiducial point represents a node of a fully connected graph, and labelled using Gabor filter response [12].

2) Holistic based approach

To do facial recognition, a holistic-based approach examines global information from a given set of faces. Small features derived from pixels in the face image represent global information. These features are used as a reference to identify faces and represent variations between different face images; therefore the uniqueness of each can be identified [10]. The holistic-based approach can be divided into the statistical approach and artificial intelligent approach. Examples of statistical approach are Principal Component Analysis (PCA), Eigenfaces and fisher face, and Linear Discriminant Analysis (LDA), while examples of artificial intelligent approach are Neural Network (NN), Support Vector Machine (SVM), Hidden Markov Model (HMM), Radial Basic Fuzzy (RBF), and Local Binary Pattern (LBP) [10], [12]

3) Hybrid based approach

Hybrid based approach combines two or more approaches in order to get more effective results in recognising a face. By making a hybrid approach, deficiencies contained in one method can be overcome by another method [12]. Some examples of hybrid-based approach are combining specific

image pre-processing steps and CNN's [15], combining Principal Component Analysis and Linear Discriminant Analysis [16], using Generalized Two-Dimensional Fisher's Linear Discriminant (G-2DFLD) method [17], combining Eigenfaces and Neural Networks [18], [19].

C. Convolutional Neural Network

Convolutional Neural Network is a computational processing system similar to the Artificial Neural Network, inspired by the work of the human brain. CNN consists of neurons that can be optimised through training. The difference between CNN and Artificial Neural Networks is that CNN is primarily used in the field of pattern recognition in pictures [20].

D. Internet of Things

The Internet of Things or IoT is a term for connection between several electronic devices through the Internet, such as cellphones, electronic devices that can be used, and home automation systems [21]. The IoT system is considered complete when at least integrating several components such as sensors, actuators, connected devices, gateways, IoT Integration Middleware, and applications [22]. Those components are explained below.

1) Sensor

The sensor is a hardware device whose function is to retrieve information from the surrounding environment through reactions to certain things, such as temperature, distance, light, sound, pressure, or specific movements.

2) Actuator

The actuator is a hardware component that receives orders from electronic devices connected to them and translates electronic signals received into specific physical actions. For example, actuators that turn on or turn off the air conditioner when the room temperature reaches a certain point.

3) Device

The device is a hardware that is connected to a sensor and actuator using a cable or wireless. A device must have a processor and storage capacity to run the software and to make a connection with IoT Integration Middleware.

4) Gateway

The gateway provides the mechanisms needed to translate different protocols, communication technologies and load formats. Gateway act as a 'middleman' that carry forward communication between the device and the next system.

5) IoT integration middleware

IoT Integration Middleware, or IoTIM for short, functions as an integration layer for various types of sensors, actuators, devices, and applications. It is responsible for receiving data from connected devices, processing data received, providing data that has been received to the connected application, and controlling the devices.

6) Application

The application component represents software that uses IoTIM to obtain information on physical boundaries and to manipulate the physical environment

E. Single Board Computer

Single Board Computer is a computer that is made and

paired on a circuit board / PCB. A Single Board Computer has the same components as a computer in general, namely the processor, memory, input/output, USB port, ethernet port, and other additional features. One example of SBC is the famous and globally used, Raspberry Pi [23].

F. K-Nearest Neighbor

K-NN algorithm is an algorithm that is used to classify objects that have several dimensions n , based on their similarities with other objects that have dimensions of n . In the area of machine learning, this algorithm has gone through development and is used to identify and recognise data patterns without requiring accurate matching for each pattern or object to be analysed. The same object will have a close Euclidean distance, while the different object has a sizeable Euclidean distance [24].

III. METHOD

To achieve a student attendance system based on face recognition, the computer must be able detect student's face from the input image; then it will identify the student, and save the student's data, which is his or her student ID number, date, and time. In order to achieve the computer's ability to detect faces and recognise students' faces from a photo, several stages have to be taken. In this section, the steps used to create a student attendance system based on face recognition are explained.

A. Preparing Student Photos

This stage is done to prepare a dataset for training the neural network and classify student based on his or her face. In this test, three students' photo was taken, with five photos each. The photos used have a size of 600 px x 800 px. Photos of a student's face taken from several sides are shown in Fig. 2. The photos are taken from the frontal side, $\pm 30^\circ$ to the right, $\pm 60^\circ$ to the right, $\pm 30^\circ$ to the left, and $\pm 60^\circ$ to the left. This is done in order to achieve higher accuracy.



Fig. 2. Example of 5 photos that are used as a dataset for each student.

B. Recognising Face

Based on the face recognition diagram, the steps to recognise face are face detection, feature extraction, and face recognition. This student attendance system is likewise taken those three steps, with the specific method used for those three steps. The steps used are as follows:

- 1) Detecting face in an image
- 2) Marking the unique part of the face and adjust the image position
- 3) Embedding face (transform the image into 128-d embedding)
- 4) Classify the result using the K-Nearest Neighbor machine learning algorithm

Explanation of the above stages is as follows:

- 1) Detecting face in an image

This stage is done to scan the input image and determine the location of the student's face in the image.

To find out whether a face exists in the input image, Convolutional Neural Network (CNN) is used. The CNN face detection library used was created by Dlib [2]. CNN is used because of its ability to detect faces more accurately than HOG [25]. The face detection result using CNN is shown in Fig. 3. A yellow box is drawn in the face area to show that the computer successfully detects a face in the input image.



Fig. 3. Face detection result using CNN.

- 2) Marking the unique part of the face and adjust the image position

Due to differences in facial position, the computer may experience difficulty in recognising a student's face, because the location of the eyes, nose, mouth, and eyebrows have changed. To overcome this, an algorithm to make a face landmark is used, and image positioning is done so that the eyes, nose, mouth and eyebrows are centred. In general, the human face has 68 specific points. These points are called face landmarks. There are several ways to locate face landmarks, but in this paper, the technique used to locate face landmarks is developed by [26]. Locating face landmarks is done by using a python script. The result of locating face landmarks is shown in Fig. 4.



Fig. 4. Student's face with landmarks.

- 1) *Embedding face*

The next step is embedding face using Dlib's CNN or deep

metric learning. The basic idea behind this is to let computer generating measurement that can help it distinguish one person with the other. The CNN is trained to map face in the input image and generate 128-d embedding. 128-d embedding is a matrix with a dimension of 128×128 [27]. Each student's face image will be run through the pre-trained network in order to get the 128-d measurements.

- 2) *Classify the result using K-NN*

In this last step, a classifier is trained based on the face embedding that has been generated. In this system, K-NN is used. The result of the classifier is the student's ID number.

C. System Design

The system created is a web-based system. Raspberry Pi 3 model B+ is used by students to record attendance, Raspberry Pi NoIR v2 camera, which is the camera connected to Raspberry Pi, is used for taking student's photos, and computer administration used to receive photos, perform face recognition, and to insert attendance data to the database. Fig. 5 shows the system design.

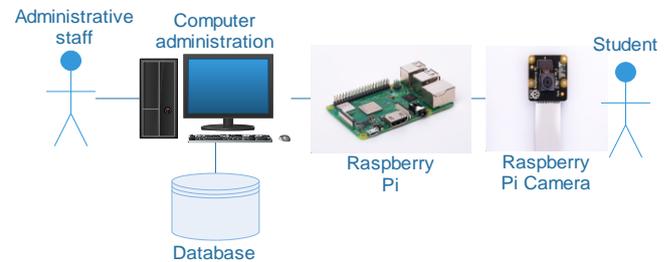


Fig. 5. Student attendance system design.

In Fig. 5, there are two actors involved, namely students and administrative staff. The administrative staff has the role of inserting data related to students, lecturers, subjects and schedules. While the student only has the role of making attendance. When a student wanted to make attendance, the camera takes a photo of the student's face. Then, data is sent from Raspberry Pi to the computer administration for processing. The computer administration determines student who is making attendance by recognising a face from a photo sent by Raspberry Pi. After successfully recognising the face in the photo, the student data, in the form of the student's ID number, date, and time, will be stored in the database.

IV. RESULTS AND DISCUSSION

This section discussed the result of facial recognition test and student attendance system that has been made. The system is tested using a computer with the specification as follows, Intel Core i5-3570 @3.40GHz 64bit, 8GB of RAM, NVIDIA GeForce GTX 750 2GB, 1TB Hard disk, and Raspberry Pi 3 model B+, with the specification of Quad Core 1.2GHz 64bit, 1GB of RAM, and 32GB MicroSD.

A. Facial Recognition Test Result

To find out the accuracy level of face recognition used in this system, a photo of a student whose face has been trained is taken. The size of the photo taken by the Raspberry Pi camera is $320 \text{ px} \times 240 \text{ px}$. This is done to make the face recognition process can be accomplished faster, because the more significant the image size, the time needed to recognize

the student's face becomes longer. Another reason is that with a small photo size, the process of sending photos from Raspberry Pi to the computer administration does not require more extended time. With sufficient lighting, as expected, the system can recognise the face of the student. Fig. 6 shows the facial recognition result from one of the student's face. A box with the student's ID number is shown as evidence that the system successfully recognises the student.

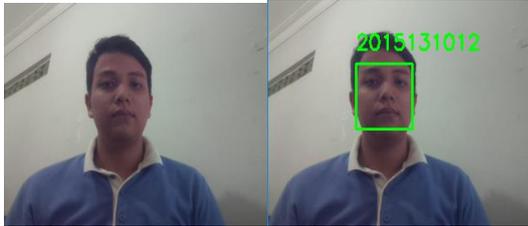


Fig. 6. Facial recognition test result.

B. Student Attendance System

The attendance system created is a web-based system. HTML, PHP, and CSS are used to build the system. XAMPP is used as a web server, and MySQL is used as the database for the attendance system. There are several pages used in this system. The user interface, along with the explanation of the usage of each page as follows:

1) Student attendance web page

Student attendance page is a page used by students to make attendance and. Through this page, students can take a photo

of their face. The appearance of the student attendance page is shown in Fig. 7. By clicking the blue button with the camera logo on it, the system will activate the Raspberry Pi Camera, and in 5 seconds, the camera will automatically take a picture. The picture will be sent to the computer in the administration office to be checked and recognise the student's face in the picture.

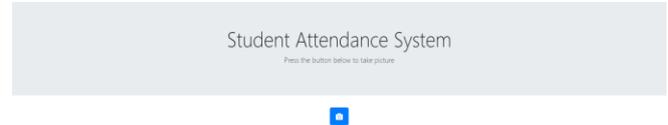


Fig. 7. Student attendance web page.

2) Administration web page

The administration web page function is to help the administrative staff enters student's data, lecturers, courses, class schedules, and student attendance reports.

- Student Web Page

Through the student web page, the staff can view student data, add student data, change student data, and delete student data. The student page view is shown in Fig. 8. To add a student, the staff needs to click the blue circle button, and the staff will be directed to a form page which has to be filled with student data. After the student's data is added, it will be shown on the student web page list. At any time, the staff can update or delete the student's data if needed.

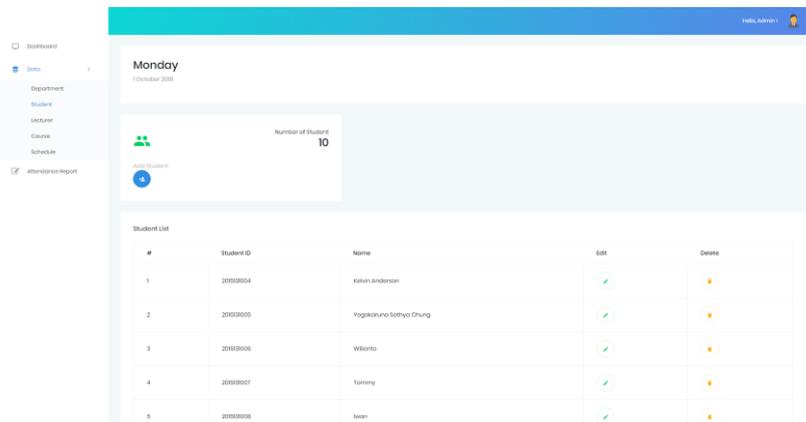


Fig. 8. Student web page.

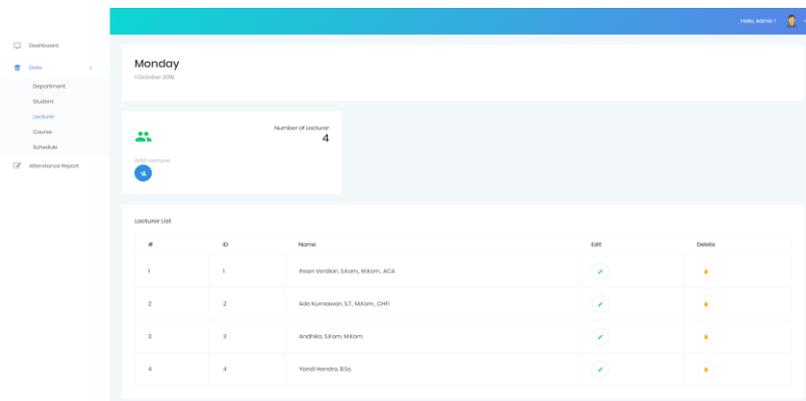


Fig. 9. Lecturer web page.

- Lecturer Web Page

The lecturer web page serves to assist the administrative staff in viewing lecturer data. The staff can add, change, and

delete lecturer data through this page. The view of the lecturer page is shown in Fig. 9. In the lecturer web page, a list of lecturer's names is shown. To add a new lecturer's data,

the staff have to click the add button. The staff will be directed to a form page where data can be inserted. From this page, the staff can update or even delete each lecturer's data if necessary, through the update or delete button.

- Course Web Page

The course web page, as shown in Fig. 10, is used by the administrative staff to view course data. Through this page, the staff can add, change, and delete course data. By clicking the add course button, the staff can add a new course data. In the add form page, course code, course name, and several credits for the course must be entered. The added new course data then will be listed on the course web page, along with other course data. Updating or deleting course data can be done by clicking the update button and the delete button.

- Schedule Web Page

The schedule web page is used by the administrative staff

to arrange class schedules that will be used for one semester. The schedule page is shown in Fig.11. To add a new schedule, the staff have to click the add button. In adding a new schedule, the staff enters the course's code, the lecturer's code, the room number, time, day and semester. A newly added schedule data will be automatically listed in the schedule web page along with another schedule. Schedule data that has been created can be changed or delete if necessary.

- Student Attendance Report Web Page

Student attendance report web page is a page used by the administrative staff to view student attendance data. The data displayed is the student's parent number, date, along with the student's time of making the attendance. These data are obtained automatically from the result of facial recognition. The student attendance report page can be seen in Fig. 12.

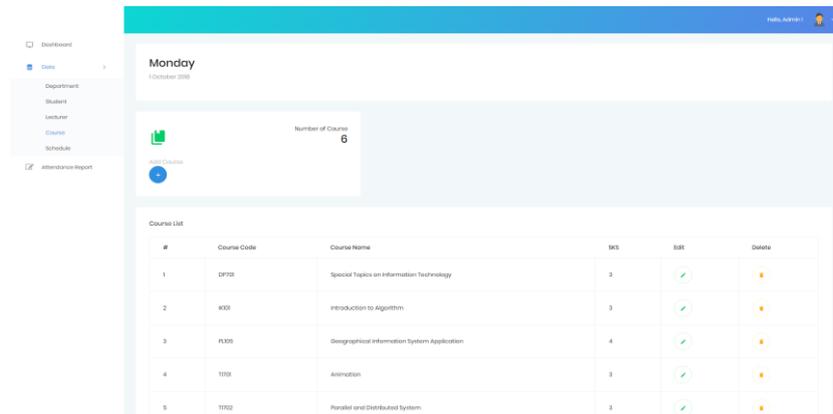


Fig. 10. Course web page.

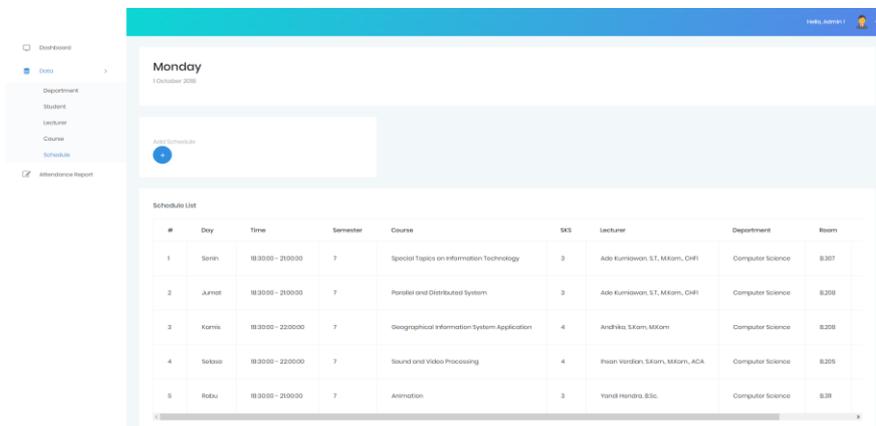


Fig. 11. Schedule web page.

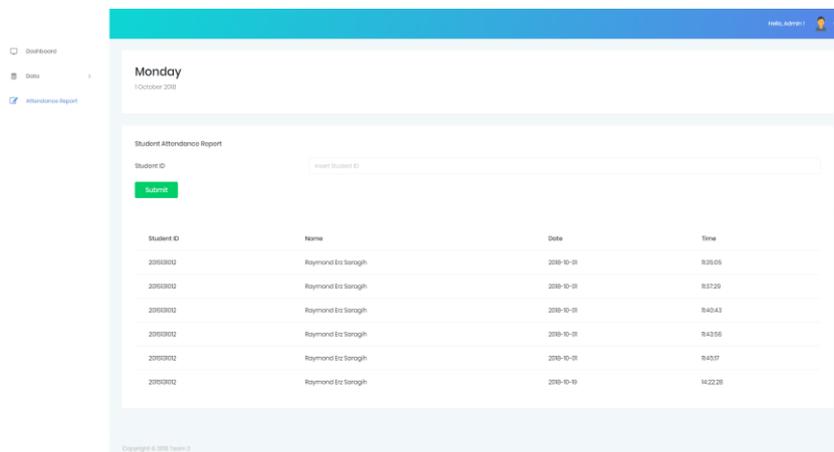


Fig. 12. Student attendance report web page.

V. CONCLUSION

In this paper, a student attendance system using facial recognition is proposed. By using Convolutional Neural Network to detect a face, Dlib's CNN or deep metric learning for facial embedding, and K-NN to classify faces, the system successfully recognises the face of a student who is making an attendance. Student data that has been identified in the form of the student's ID number, date and time, is used by the system to record student attendance. This system makes the student attendance process done automatically and is expected to be able to replace the old manual attendance process, which is currently used.

For future work, the plan is to use cloud-based face recognition in order to speed up the face recognition process. The use of another more sophisticated face recognition method is planned in order to be able to compare the performance, and hopefully gives better performance (speed and accuracy) than the method that has been used, in this case, the Convolutional Neural Network.

ACKNOWLEDGEMENT

This study was supported by Universitas Respati Indonesia and Universitas Universal.

REFERENCES

[1] C. Dobkin, R. Gil, and J. Marion, *Causes and Consequences of Skipping Class in College*, July 2007.

[2] D. E. King, "Dlib-ml: A machine learning toolkit," *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, 2009.

[3] A. S. S. Mane, A. Shah, N. Shrivastava, and B. Thakare, "A survey of face detection algorithms," in *Proc. 2017 Int. Conf. Inven. Syst. Control*, pp. 1–4, 2017.

[4] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. 2001 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition. CVPR, 2001*, vol. 1, p. I-511-I-518, 2001.

[5] [5] Vikram K and S. Padmavathi, "Facial Parts Detection Using Viola-Jones Algorithm," in *Proc. 4th International Conference on Advanced Computing and Communication Systems*, 2015, pp. 6–7.

[6] V. K. Kadam and D. G. Ganakwar, "Face detection: A literature review," *International Journal of Innovative Research in Science, Engineering and Technology*, pp. 13017–13023, 2017.

[7] K. D. Pandya, "Face Detection — A literature survey," *Int. J. Comput. Tech.*, vol. 3, no. 1, pp. 67–70, 2016.

[8] A. K. Datta, M. Datta, and P. K. Banerjee, *Face Detection and Recognition: Theory and Practice*, Florida: CRC Press, 2016.

[9] K. Dang and S. Sharma, "Review and Comparison of Face Detection Algorithms," in *Proc. 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence*, 2017, pp. 629–633.

[10] G. M. Zafaruddin and H. S. Fadewar, "Face recognition: A holistic approach review," in *Proc. 2014 International Conference on Contemporary Computing and Informatics (IC3I)*, 2014, pp. 175–178.

[11] P. Wagh, R. Thakare, J. Chaudhari, and S. Patil, "Attendance system based on face recognition using eigen face and PCA algorithms," in *Proc. 2015 Int. Conf. Green Comput. Internet Things*, 2016, pp. 303–308.

[12] L. Masupha, T. Zuva, S. Ngwira, and O. Esan, "Face recognition techniques, their advantages, disadvantages and performance evaluation," in *Proc. 2015 International Conference on Computing, Communication and Security (ICCCS)*, 2015, no. i, pp. 1–5.

[13] N. H. Barnouti, S. S. M. Al-Dabbagh, and W. E. Matti, "Face recognition: A literature review," *Int. J. Appl. Inf. Syst.*, vol. 11, no. 4, pp. 21–31, Sep. 2016.

[14] R. Patel and S. B. Yagnik, "A literature survey on face recognition techniques," *A Lit. Surv. Face Recognit. Tech.*, vol. 5, no. 4, pp. 189–194, Feb. 2015.

[15] R. V. Puthanidam, "A Hybrid approach for facial expression recognition," in *Proc. the 12th International Conference on Ubiquitous Information Management and Communication*, 2018.

[16] P. Sood, E. S. Puri, E. V. Kaur, and D. N. Dhillon, "Face recognition system using hybrid approach of both principal component analysis

and linear discriminant analysis," *IJARCCCE*, vol. 6, no. 3, pp. 276–280, 2017.

[17] S. Chowdhury, J. K. Sing, D. K. Basu, and M. Nasipuri, "A hybrid approach to face recognition using generalized two-dimensional fisher's linear discriminant method," in *Proc. 3rd Int. Conf. Emerg. Trends Eng. Technol.*, 2010, pp. 506–511.

[18] M. Rizon *et al.*, "Face recognition using eigenfaces and neural networks," *Am. J. Appl. Sci.*, vol. 2, no. 6, pp. 1872–1875, 2006.

[19] M. Agarwal, N. Jain, M. M. Kumar, and H. Agrawal, "Face Recognition Using Eigen Faces and Artificial Neural Network," *Int. J. Comput. Theory Eng.*, vol. 2, no. 4, pp. 624–629, 2010.

[20] K. O'Shea and R. Nash. (2015). An introduction to convolutional neural networks. [Online]. Available: https://white.stanford.edu/teach/index.php/An_Introduction_to_Convolutional_Neural_Networks

[21] G. Elhayatmy, N. Dey, and A. S. Ashour, *Internet of Things Based Wireless Body Area Network in Healthcare*, vol. 30, Cham: Springer International Publishing, 2018.

[22] B. Di, M. Kuan, C. Li, L. T. Yang, and A. Esposito, *Internet of Things Internet of Everything Algorithms, Methodologies, Technologies and Perspectives*.

[23] A. Pajankar, *Raspberry Pi Image Processing Programming Develop Real-Life Examples with Python, Pillow, and SciPy*. 2017.

[24] M. Parsian, *Data Algorithms*. Sebastopol: O'Reilly Media, 2015.

[25] A. Suleiman, Y. Chen, J. Emer, and V. Sze, "Towards closing the energy gap between hog and cnn features for embedded vision," in *Proc. 2017 IEEE International Symposium on Circuits and Systems*, 2017, pp. 1–4.

[26] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1867–1874.

[27] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2015, vol. 07–12–June, pp. 815–823.



Tata Sutabri received his doctoral degree in information technology in 2017 from Gunadarma University. He is currently a lecturer in the Department of Information System of Universitas Respati Indonesia. His research interests include information system, e-government, data mining, and machine learning.



Pamungkur completed his master of management degrees from Universitas Diponegoro, Indonesia. He is currently working as an associate professor at Department of Economic Development, Sekolah Tinggi Ilmu Ekonomi Kuala Kapuas, Central Kalimantan, Indonesia. His research interests include knowledge management, and now his research is interested in the use of machine learning in economics.



Ade Kurniawan received his master's degree in digital forensic in 2017 from Universitas Islam Indonesia. He is currently a lecturer in the Department of Informatics Engineering of Universal University. His research interests include anomaly detection, deep learning, digital forensics, IoT, machine learning, and network security.



Raymond Erz Saragih is currently an assistant researcher at Universal Universit, Batam, Riau Islands, Indonesia. He received an undergraduate degree in 2019 in informatics engineering at Universal University graduate with Cum laude. His research interests include artificial intelligence, in particular, computer vision, deep learning, business intelligence, blockchain, machine learning for security and technology for developing smart city and smart agriculture.