JURNAL RISET INFORMATIKA **P-ISSN: 2656-1743 |E-ISSN: 2656-1735**
Vol. 5, No. 2 March 2023 DOI: https://doi.org/10.34288/jri.v5i2.340
Accredited rank 3 (SINTA 3), excerpts from the decision of the Minister of RISTEK-BRIN No. 200/M/KPT/2020

MOBILE BASED STUDENT PRESENCE SYSTEM USING HAAR CASCADE AND EIGENFACE FACIAL RECOGNITION METHODS

Suherman Achmad[1], Nazori A Z[2], Achmad Solichin[3]

Faculty of Information Technology
Universitas Budi Luhur
Jakarta, Indonesia
[1)]1811600202@student.budiluhur.ac.id, [2)]nazori@budiluhur.ac.id, [3)]achmad.solichin@budiluhur.ac.id
(*) Corresponding Author

Abstract

Using biometric technology for recording attendance in the school environment is still not widely done by researchers. In this study, a solution was proposed to the problems that occurred in the school environment where parents/guardians could not monitor the presence of their children in school. The solution offered is a student attendance recording system based on facial recognition algorithms (face recognition). The built system can record the presence of students when entering the classroom and when returning home or out of class. Proposed methods for identifying student attendance are the Haar Cascade and Eigenface algorithms. The system can also provide notice of attendance or absence of students in real time to parents/guardians via email that has been registered. Based on the test results, the method can provide accurate and fast facial recognition results. The presence system developed based on mobile can recognize faces up to a distance of 200-300 cm with low and moderate light intensity.

Keywords: presence system, Haar Cascade Classifier, Eigenface

Abstrak

Pemanfaatan teknologi biometrik untuk pencatatan kehadiran di lingkungan sekolah masih belum banyak dilakukan oleh peneliti sebelumnya. Dalam penelitian ini, diusulkan sebuah solusi atas permasalahan yang terjadi di lingkungan sekolah yang mana orang tua/wali tidak dapat melakukan monitoring terhadap kehadiran anaknya di sekolah. Solusi yang ditawarkan berupa sistem pencatatan kehadiran siswa berbasis algoritma pengenalan wajah (face recognition). Sistem yang dibangun dapat mencatat kehadiran siswa saat masuk ke kelas, maupun saat pulang atau keluar dari kelas. Metode yang diusulkan untuk mengidentifikasi kehadiran siswa adalah algoritma Haar Cascade dan Eigenface. Sistem juga dapat memberikan pemberitahuan kehadiran maupun ketidakhadiran siswa secara realtime ke orang tua/wali melalui email yang telah terdaftar. Berdasarkan hasil pengujian, metode tersebut mampu memberikan hasil pengenalan wajah yang akurat dan cepat. Sistem presensi yang dikembangkan berbasis mobile mampu mengenali wajah hingga jarak 200-300 cm dengan intensitas cahaya rendah dan sedang.

Kata kunci: sistem presensi, Haar Cascade Classifier, Eigenface

INTRODUCTION

The speed of access to information is currently the most fundamental need for managing and transferring data. The speed of access to information is currently the most fundamental need for managing and transferring data. Existing and rapidly developing technology is now expected to build a system to provide solutions for disciplining students and providing benefits for schools. A series of evaluations conducted by administrative bureaus and administrative staff found several weaknesses related to the presence of learners. In this case, the system of recording attendance in schools is still manually done with a signature recording system, which is considered easy to manipulate, so the lack of information received by parents/guardians on the presence of their children in school.

Therefore, a computerized system can manage information quickly and accurately to help smooth activity and become one of the influential factors in improving the discipline of learners. Facial recognition technology is growing and widely used for the identification process. Facial recognition is one of the technologies that has now been applied to many applications in the field of presence (Rijal & Ariefianto, 2008; Satwikayana et al., 2021; Septyanto et al., 2020). Among others, supporting the identification system in schools to become a tool to find out data collection such as names, nis, and majors, in addition to facial recognition conditions that are input (input) system, is also a significant problem.

Related to this study, there are several similar previous studies. Simaremare and Kurniawan compare LBPH and Eigenface for recognizing three faces at once in a real-time situation (Simaremare & Kurniawan, 2016). The researchers tested the accuracy of both methods in recognizing three faces at once. The test was conducted on 300 samples of facial imagery with four lighting conditions, namely indoor and outdoor daylight. The results of this test showed that the accuracy rate of LBPH is better than eigenface, with the average accuracy of LBPH being 93.54% and eigenface being 63.54%. The false rejection rate (FRR) in the LBPH method is lower than that of the eigenface method, with the average FRR LBPH being 0.24% and FRR eigenface being 6.38%. This study only compares the two methods and is not a merger of the two methods, so the results obtained are not maximal. Another researcher compared Eigenface and Fisherface methods for face recognition (Firasari et al., 2022).

Then, Septyanto et al. proposed a facial recognition presence application using Haar Cascade Algorithm (Septyanto et al., 2020). This test was conducted on 13 Starcross Store employees, each conducting 30 presidency trials. Successful absentees had success scores of 87%, and 13% failed from 390 attempts. Some absenteeism that fails occurs because several factors can affect absenteeism, such as high lighting, jacked head position, and the use of attributes (hats, glasses, etc.). The result of this study is that the system can identify faces with a reasonable degree of accuracy, but has limitations in knowing the face if the lack of lighting, conversely if the lighting is high or too bright, then the face cannot be identified, for it requires additional methods to overcome such weaknesses. Many studies have also used Haar Cascade (Anarki et al., 2021; Behera, 2020; Sulistiyo et al., 2014).

In another research, the Fisherface method supports the academic system (Amri & Rahmata, 2016; Firasari et al., 2022). The system is built using primary visual programming languages and databases using MySQL. In this test, the results obtained differed from one face to another, and the results came out in the name of the class of study program majors. Facial recognition processing in absenteeism can work well if the data in the database is not too much and at the same lighting, so the level of face search approaching in the database can be better. Based on testing, the percentage of facial recognition success reaches 80%. The disadvantage of this panel is that the detection results are sometimes not maximal if the distance of the face with the webcam is far and the position is not by the webcam. Another weakness is that the system cannot identify the face if the light is too bright or dark.

Mobile-based face recognition technology is a rapidly developing field with significant implications for security, law enforcement, and mobile applications (Arisandi et al., 2018; Samet & Tanriverdi, 2017). Researchers are currently investigating the accuracy and performance of mobile-based face recognition systems compared to traditional desktop-based systems (Abuzar et al., 2020; Alburaiki et al., 2021; Rodavia et al., 2017). There is growing concern about the privacy implications of mobile-based face recognition and how it can be used to identify individuals without their consent or knowledge (Ahmed Khan et al., 2021). Studies are being conducted to explore ways to improve the accuracy and performance of mobile-based face recognition systems.

Moreover, researchers are also studying the performance of mobile-based face recognition systems on diverse faces, including different races, ages, and genders. Studies have found that some systems perform better than others on diverse faces and that there are significant disparities in performance depending on the demographic group. Overall, ongoing studies in mobile-based face recognition focus on developing new algorithms and techniques to improve the technology's accuracy and performance, addressing privacy concerns, and improving performance on diverse faces.

This android-based attendance identification system aims to comfort parents/guardians, improve student discipline, and connect information online. The formula of the problem in this study is how the process of identifying the presence of learners entering and exiting the classroom can be detected early, accurately, and on target. Based on previous research mentioned above, it is necessary to build further research by combining two different methods to produce a model of the student presence system using android-based facial recognition patterns (Samet & Tanriverdi, 2017).

RESEARCH METHODOLOGY

This research method is used as a guideline for researchers to implement research so that the results achieved do not deviate from the goals set. In this case, the stages of research conducted are as follows:

Data Collection Stages

P-ISSN: 2656-1743 |E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i2.340
Accredited rank 3 (SINTA 3), excerpts from the decision of the Minister of RISTEK-BRIN No. 200/M/KPT/2020

The data collection stages in this study are as follows:

1. Literature Studies or Study Review. This technique begins with collecting data by studying the necessary materials, concepts, and theories from several written sources (books, magazines, tutorials, etc.), and the necessary understanding will be used as a reference for the preparation of research.
2. Direct Observation. This technique will be held direct observation of the main symptoms of what is being studied. Observations made in actual situations are necessary for particular purposes.
3. Designing student presence identification applications to support the proposed student presence identification system to provide information to parents/guardians via email using android-based facial recognition.
4. Prepare sample data as input in research.

System Development Stages

The stages of application system development of student presence using the waterfall model, where the model consists of analysis, design, programming (coding), testing, and maintenance.

The stages of the process use the waterfall model as follows:

1. Analysis
   At the analysis stage, collecting needs is complete, observed, defined in detail, and presented as a specification system.
2. Design
   At this stage, the system design process is carried out by requirement on hardware and software to form the overall system architecture. Software design will identify the basic abstraction of software systems and their relationships.
3. Coding
   At the coding stage, software design is realized as a program. Furthermore, the program that has been realized will be tested by verifying whether each unit meets its initial specifications.
4. Testing
   At this stage, the program that has been tested is confirmed to have fulfilled the software.
   Requirement. When fulfilled, the user can implement and use the software system.
5. Maintenance
   This stage is the maintenance part of the system, which carries out the program's operation, such as changes or improvements of the user's needs due to adaptation to the actual situation.
   The waterfall model is seen in the image below.

Design and Implementation

1. **System Planning**
   starting from entering the classroom and then doing the presence automatically to enter as a student. An image of the activity diagram identification of the student's presence can be seen in figure 1.

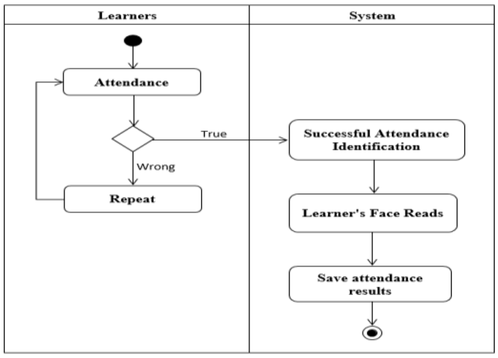


Figure 1. Activity Diagram Identifies Student Presence

The Flow Activity Diagram in figure 1 is described as follows:

a. Learners perform presence activities. If true, then the identification process of the presence is successful; if wrong, then facial matching is not successfully identified, and learners are asked to return to presence.
b. After successful identification, the student's face will be read on the system.
c. Next, the system will store the student's face on the server.

2. **Design of Functions and Infrastructure**

The infrastructure on the system will be built based on mutually agreed needs; outside of unified modeling language (UML) design, there is also application design for the user interface. This system will be described as infrastructurally by students connected to the presence system server and parents/guardians as receiving information. To be more clear can be seen in Figure 2 System network infrastructure.

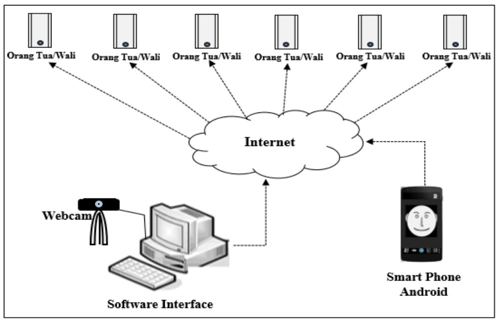


Figure 2. Design function and infrastructure

The flow of the function and system infrastructure design in Figure 2 is the stage in running the presence system. Here is an explanation of the flow of system and infrastructure function design that can be done, including:

a) The first step, learners make a presence using facial recognition through a webcam or smartphone camera
b) Step two, the system connected to the internet will verify the face of the learner and whether the face of the participant can perform facial matching on the system.
c) In Step three, the learner's facial data will be stored on the system storage media, where NIS, Name, Class, and Email parents/guardians are stored using facial recognition when making a facial list.
d) Step four, student data stored on storage media will automatically connect to the cloud or internet network.
e) In the last step, parents will receive a message about their child's presence through an online email address.

3. System Component

Each component in Figure 2 has different functions. The function of each component will be explained as follows:

a. The webcam component is hardware that serves as a tool to take the image of the face of learners. This webcam will be a facial matching tool when conducting presence checks and taking images of the learner's face.
b. The interface component is software on a desktop computer that serves as a tool to design and create a system that can connect with the web to take images of students' faces. Smartphone components are multi-function devices used to install android-based presence applications so that applications can be used easily and quickly.
c. The facial image component performs facial recognition mechanisms with the help of a webcam or smartphone camera, and this face capture is helpful for storage in the database and as input at the time of the presence.
d. Internet components function as an intermediary between the user and the system. The design is simple to make it easier for users to access information.

Facial Steps with Haar Algorithm

The process of facial shortness is needed in several stages. In the haar method, the process of facial shortness looks as in Figure 3.

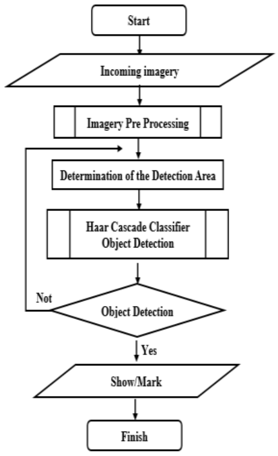


Figure 3. Facial shortness process

From Figure 3 above, it is determined in advance whether the area is detected whether there is an object or not. The next process is to detect objects using Haar Cascade Classifier, with steps to be described as follows.

a. Calculate feature samples with the haar algorithm.

Figure 4. Rectangular Feature Haar Cascade

Figure 4 is the result of camera detection, where there is a blue box which is a frame with values Y = 480 and X = 640. In the blue box is the exposed face. Figure 5 is the result of facial detection that has been recorded, so it has a size of 240 x 320. An original image converted to grayscale is shown in Figure 9

Figure 5. Face Detection with Haar-like Feature

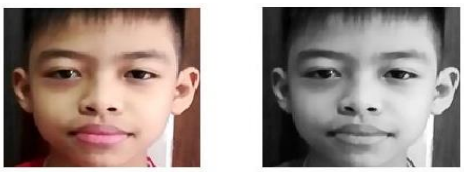


Figure 6. The difference in the Original Image with Grayscale

P-ISSN: 2656-1743 |E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i2.340
Accredited rank 3 (SINTA 3), excerpts from the decision of the Minister of RISTEK-BRIN No. 200/M/KPT/2020

From the results of figure 6, it will be converted into matrix values so that haar-like squares in the input image are obtained as follows in Table 1.

Table 1. Square Haar-Like Image Input

46	45	44	45	44	44
46	44	41	43	44	44
46	43	40	40	43	44
45	40	40	40	44	45
45	41	39	40	44	45
46	41	40	41	45	46

The process of calculating the dark value and the light value, i.e. Feature value = Number of Pixel Values (dark value) – Number of Pixel Values (light value), thus generating the feature value haar = 213. To calculate the haar value feature using the Summed Area Table, known as the integral image, first formed an integral value matrix image. Here is the matrix of integral values obtained from the input image, as seen in table 2.

Table 2. The integral value of the image from the input image

46	91	135	180	224	268
92	181	266	354	442	530
138	270	395	523	654	786
183	355	520	688	863	1040
228	441	645	853	1072	1294
274	528	772	1021	1285	1553

The haar feature value of the matrix area above is calculated using the following formula:

$i(x',y')=s(A)+s(D)+s(B)-s(C).$

So that the haar value feature is obtained = (528+46-274-91) - (772+91-528-135) + (1021+135-772-180) = 213

The haar = 213 value feature is then compared to the threshold determined as object detection. If haar's feature value is higher than the threshold, it can be said that the area meets the Haar filter. This process will continue to retest the area with other Haar filters, and if all Haar filters are met, then it is said that there are observed objects in that area.

RESULTS AND DISCUSSIONS

Facial Matching Steps with Eigenface Algorithm

Facial recognition algorithms begin by creating a matrix of columns from faces inputted into a database. A column matrix's average vector image (mean) is calculated by dividing it by the number of images stored in the database.

In the Eigenface algorithm, the first step before determining the eigenface value first arranges a flat vector image matrix. The process flow can be seen in Figure 7.

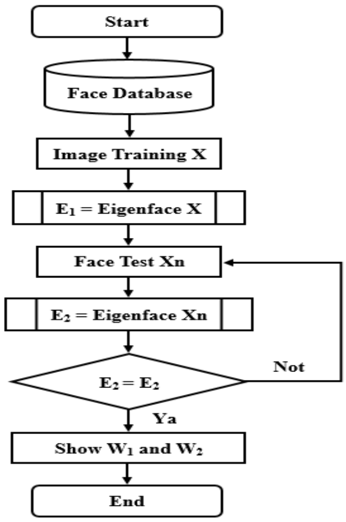


Figure 7. Facial Recognition Process

Facial recognition algorithms are performed through several stages.

1) The first step is arranging an S matrix set of all training images. Here is a training image of two facial data, as seen in Figure 8 and Figure 9, each of which has a matrix value.

$$C_1 = \begin{bmatrix} 1 & 0 & 2 \\ 1 & 2 & 1 \\ 0 & 2 & 2 \end{bmatrix}$$

Figure 8. Training Image Image Of Face Image 1

$$C_1 = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 2 & 1 \\ 1 & 2 & 4 \end{bmatrix}$$

Figure 9. Training Image image

P-ISSN: 2656-1743 | E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i2.340

Accredited rank 3 (SINTA 3), excerpts from the decision of the Minister of RISTEK-BRIN No. 200/M/KPT/2020

2) The second step is arranging the entire training image into one matrix.

$$\begin{bmatrix} a & b & c \\ x & y & z \end{bmatrix} \rightarrow [a \quad b \quad c \quad x \quad y \quad z]$$

$$C_1 = \begin{bmatrix} 1 & 0 & 2 \\ 0 & 2 & 1 \\ 0 & 2 & 2 \end{bmatrix} \rightarrow [1 \quad 0 \quad 2 \quad 0 \quad 2 \quad 1 \quad 0 \quad 2 \quad 2]$$

$$C_2 = \begin{bmatrix} 1 & 2 & 2 \\ 0 & 2 & 1 \\ 0 & 2 & 4 \end{bmatrix} \rightarrow [1 \quad 1 \quad 2 \quad 0 \quad 2 \quad 1 \quad 0 \quad 2 \quad 4]$$

3) Here is a count of flat vector averages.
Sum up the entire row from the flat vector obtained so that a matrix measuring 1 x (H x W) will be obtained.

$$C_1 + C_2 = \begin{bmatrix} 1 & 0 & 2 & 0 & 2 & 1 & 0 & 2 & 2 \\ 1 & 1 & 2 & 0 & 2 & 1 & 0 & 2 & 4 \end{bmatrix}$$

$$C_1 + C_2 = [2 \quad 1 \quad 4 \quad 0 \quad 4 \quad 2 \quad 0 \quad 4 \quad 6]$$

Next, divide the matrix result by the number of images N to get the flat vector average value.

$$\left[\frac{2 \quad 1 \quad 4 \quad 0 \quad 4 \quad 2 \quad 0 \quad 4 \quad 6}{2}\right] = [1 \quad 1 \quad 2 \quad 0 \quad 2 \quad 1 \quad 0 \quad 2 \quad 3]$$

4) The following flat vector average value will be used to calculate the eigenface value of the facial image in the training image.
Using the flat vector average value above, the eigenface can be calculated. How to reduce the rows on the flat vector matrix with flat vector average values. If the result is below zero, the value is replaced with zero.

$$C_1 = \begin{array}{ccccccccc} 1 & 0 & 2 & 0 & 2 & 1 & 0 & 2 & 2 \\ 1 & 1 & 2 & 0 & 2 & 1 & 0 & 2 & 3 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$C_1 = \begin{array}{ccccccccc} 1 & 1 & 2 & 0 & 2 & 1 & 0 & 2 & 3 \\ 1 & 1 & 2 & 0 & 2 & 1 & 0 & 2 & 4 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{array}$$

5) Here is the identified process.
The identification process calculates the eigenface value of the test face matrix to determine the eigenface and flat vector values. The results can be seen in Figure 10..

$$C_t = \begin{bmatrix} 2 & 2 & 4 \\ 1 & 2 & 2 \\ 2 & 4 & 4 \end{bmatrix}$$

Figure 10. Test Image (Testface)

$$C_t = \begin{bmatrix} 2 & 2 & 4 \\ 1 & 2 & 2 \\ 2 & 4 & 4 \end{bmatrix} \rightarrow [2 \quad 2 \quad 4 \quad 1 \quad 2 \quad 2 \quad 2 \quad 4 \quad 4]$$

$$C_t = \left[\begin{array}{ccccccccc} 2 & 2 & 4 & 1 & 2 & 2 & 2 & 4 & 4 \\ 1 & 1 & 2 & 1 & 0 & 0 & 0 & 2 & 3 \\ \hline 1 & 1 & 2 & 1 & 0 & 2 & 2 & 2 & 1 \\ & & & & & & & & \end{array}\right]$$

Eigenface value of test image

$$C_t = [1 \quad 1 \quad 2 \quad 1 \quad 0 \quad 2 \quad 2 \quad 2 \quad 1]$$

6) Once the eigenface value is obtained, it can be identified by determining the shortest distance with the eigenface of the eigenvector training image. Here determines the smallest eigenface value of the two image faces that are already known eigenface values. The results can be calculated as follows.
Eigenface value C1 = (0 0 0 0 0 0 0 0 0 0 0)

$$C_1 = \begin{array}{ccccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 2 & 1 & 0 & 2 & 2 & 2 & 1 \\ \hline -1 & -1 & -2 & -1 & 0 & -2 & -2 & -2 & -1 \end{array}$$

C_1 = 1+1+2+1+0+2+2+2+1 = 12

Eigenface value C_2 = (0 0 0 0 0 1 0 0 1)

$$C_1 = \begin{array}{ccccccccc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 2 & 1 & 0 & 2 & 2 & 2 & 1 \\ \hline -1 & -1 & -2 & -1 & 0 & -1 & -2 & -2 & 0 \end{array}$$

C_2 = 1+1+2+1+0+1+2+2+0 = 10

The smallest eigenface value of the two image faces above obtained from the distance of the image of face one has the smallest value of 10. The identification results concluded that the test face was more similar to face two than face one.

Application Testing

Once this pretension identification software using facial recognition is built, the next stage is the display trial stage. This trial stage includes testing from the beginning of student data entry to the face-matching process when doing the presence of entry and home. The details will be described as follows.

1. Test the app's entry window

The use of the application is initiated by the user who must log in first (Figure 11), if the login is successful then the user can enter the system. There are menus that can be selected for activities in the

JURNAL RISET INFORMATIKA
Vol. 5, No. 2 March 2023
P-ISSN: 2656-1743 |E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i2.340
Accredited rank 3 (SINTA 3), excerpts from the decision of the Minister of RISTEK-BRIN No. 200/M/KPT/2020

system according to the desired function (Figure 12).

Figure 11. App Login Menu View

Figure 12. App Main Menu View

2. Test the filling of data on the menu of the list of learners.

Before testing, you can perform the Face Detection Display (Figure 13). The detected face will show a blue focus box furthermore after the face is detected, Data Input on the Face List (Figure 14). If it has been saved, then if the face is detected, a green focus box will appear with a caption with the name according to the detected face. Figure 16 shows the faces stored on list data.

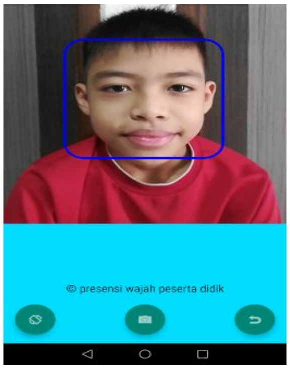


Figure 13. Face Detection View

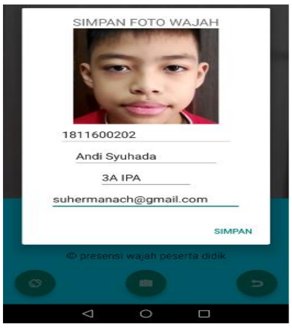


Figure 14. Data Input on the Face List

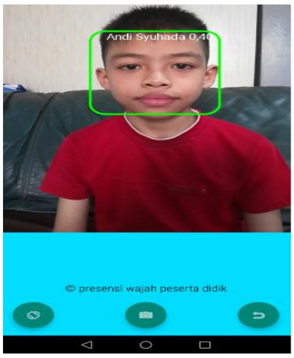


Figure 15. Faces of Registered Learners

Figure 16. Faces Stored On List Data

3. Test the filling of data on the class entry menu.

On this menu, students will do the presence of entering the class by using their faces. The step is that the learner faces the application, which is approximately 30cm away, then the face will be detected in the blue box and then face matching, then the face and name of the learner and the duration of the face matching process will be seen on the green box, after which automatically the student data will be stored and sent to the parent/guardian email, like the Figure 17, 18, and 19 below.

P-ISSN: 2656-1743 | E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i2.340

Accredited rank 3 (SINTA 3), excerpts from the decision of the Minister of RISTEK-BRIN No. 200/M/KPT/2020

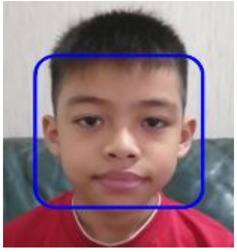


Figure 17. Face Entry Detection

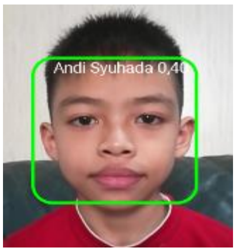


Figure 18. Face Entry Match

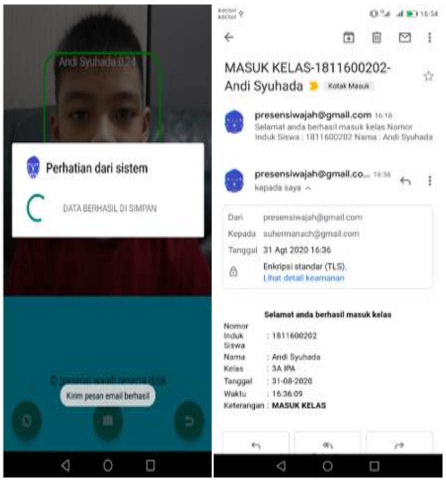


Figure 19. The process saves incoming data and messages sent to the email

4. Test data are filled on the out-of-class menu.

In this menu, students will do the presence of exiting the class / going home using the face. The step is that the learner faces the application, which is approximately 30cm away, then the face will be detected in the blue box and then face matching, then the face and name of the learner and the duration of the face matching process will be seen on the green box, after which automatically the student data will be stored and sent to the parent/guardian email, like the Figure 20, 21, and 22 below.

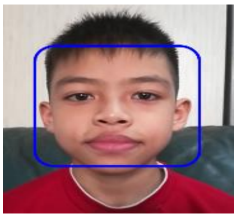


Figure 20. Face Entry Detection

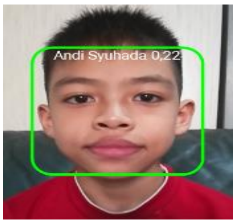


Figure 21. Face Entry Matc

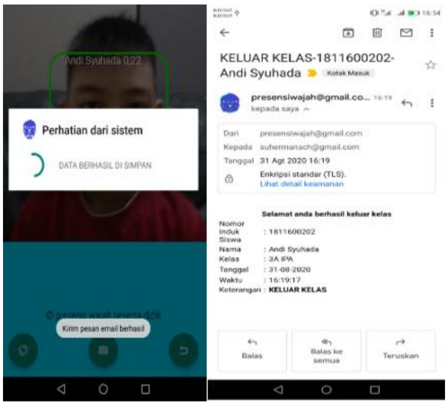


Figure 22 Process save data home and messages sent to the email

2.1. System Testing Based on Distance and Intensity of Light.

This test intends to determine the objects detected or undetected that can present learners in the classroom by matching faces at different distances/radii and light intensity. Here is a test by matching faces at a distance/ radius and intensity of light.

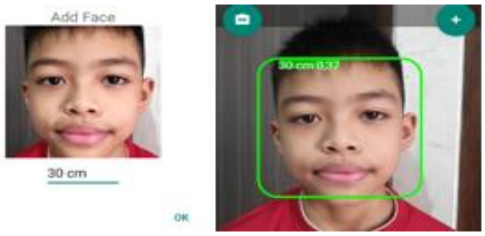
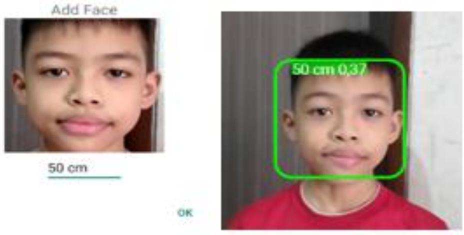


Figure 23. Face Matching Distance 30cm

Figure 23 is a facial matching test from 0cm to 30cm with low light intensity.



Gambar 24. Face Matching Distance 50cm

Figure 24 is a facial matching test from 0cm to 50cm with low light intensity.

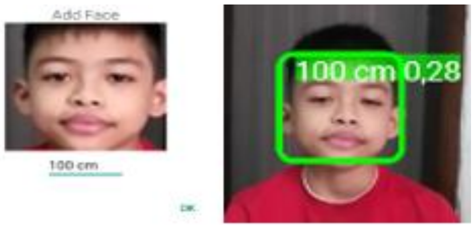


Figure 25. Face Matching Distance 100cm

P-ISSN: 2656-1743 |E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i2.340
Accredited rank 3 (SINTA 3), excerpts from the decision of the Minister of RISTEK-BRIN No. 200/M/KPT/2020

Figure 25 is a facial matching test at a distance of 0cm to 100cm with low light intensity.

Figure 26. Face Matching Distance 150cm

Figure 26 is a facial matching test from 0cm to 150cm with low light intensity.

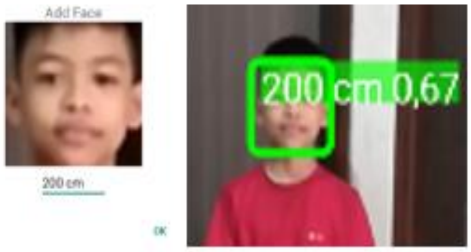


Figure 27. Face Matching Distance 200cm

Figure 27 is a facial matching test from 0cm to 200cm with low light intensity.

Figure 28. Face Matching Distance 250cm

Figure 28 is a facial matching test from 0cm to 250cm with low light intensity.

Figure 29. 300cm Distance Face Matching

Figure 29 is a facial matching test from 0cm to 250cm with low light intensity. Based on the face-matching image that has gone through the application test process, the following summary of the results of the face-matching trial can be seen in table 3 below.

Table 3. Presence Test Results with Distance Coverage and Light Intensity

No	Detection Object	Distance	Light Intensity	Detection
1	Figure IV-30	0cm until 30cm	Low	Detected
2	Figure IV-31	0cm until 50cm	Low	Detected
3	Figure IV-32	0cm until 100cm	Low	Detected
4	Figure IV-33	0cm until 150cm	Low	Detected
5	Figure IV-34	0cm until 200cm	Low	Detected
6	Figure IV-35	0cm until 250cm	Low	Detected
7	Figure IV-36	0cm until 300cm	Low	Detected

CONCLUSION

As for the conclusion of this study, where applications can be run automatically to read and analyze the presence of learners, applications can be run on android smartphones version 5.0 Lollipop, 6.0 Marshmallow, 7.0 Nougat, 8.0 Oreo, 9.0 Pie, and android 10, in addition, this study can write the results of detection done either verbose when executed or written in true-false values entered in the processed database. This study has found the results of testing with the merger of Two algorithms or methods that can recognize faces up to a distance of 300cm and obtain an average percentage of facial recognition system success reaches 90%, so that the application in this study can provide information to parents/guardians quickly, precisely and accurately. This research still has some limitations. Therefore it still needs to be developed to increase effectiveness, efficiency, and the addition of features to support the right target presence. For that, it is necessary to develop subsequent systems, such as software and hardware, that must be met for this application to work correctly. This study is limited to locating learners and needs to be developed before it can be applied to other fields.

REFERENCES

Abuzar, M., Ahmad, A. Bin, & Ahmad, A. A. Bin. (2020). A Survey on Student Attendance System Using Face Recognition. *ICRITO 2020 - IEEE 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)*, 1252–1257. https://doi.org/10.1109/ICRITO48877.2020.9197815

Ahmed Khan, D., Rizvi Assistant Professor, A., & Scholar, D. (2021). AI based Facial Recognition Technology and Criminal Justice: Issues and Challenges. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(14), 3384–3392. https://www.turcomat.org/index.php/turkbilmat/article/view/10923

Alburaiki, M. S. M., Johar, G. M., Helmi, R. A. A., & Alkawaz, M. H. (2021). Mobile Based Attendance System: Face Recognition and Location Detection using Machine Learning. *2021 IEEE 12th Control and System Graduate Research Colloquium, ICSGRC 2021 - Proceedings*, 177–182. https://doi.org/10.1109/ICSGRC53186.2021.9515221

Amri, A., & Rahmata, T. (2016). Pengenalan Wajah Menggunakan Metode Fisherface untuk Mendukung Sistem Akademik. *Proceeding Seminar Nasional Ilmu Komputer (Seminasik)*, *1*(1), 39–43. http://jurnal.umuslim.ac.id/index.php/seminasik/article/view/474

Anarki, G. A., Auliasari, K., & Orisa, M. (2021). Penerapan Metode Haar Cascade Pada Aplikasi Deteksi Masker. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *5*(1), 179–186. https://doi.org/10.36040/JATI.V5I1.3214

Arisandi, D., Syahputra, M. F., Putri, I. L., Purnamawati, S., Rahmat, R. F., & Sari, P. P. (2018). A real time mobile-based face recognition with fisherface methods. *Journal of Physics: Conference Series*, *978*(1), 012038. https://doi.org/10.1088/1742-6596/978/1/012038

Behera, G. S. (2020, December 24). *Face Detection with Haar Cascade*. Towards Data Science. https://towardsdatascience.com/face-detection-with-haar-cascade-727f68dafd08

Firasari, E., Cahyanti, F. L. D., Sarasati, F., & Widiastuti, W. (2022). Comparison Of Eigenface and Fisherface Methods For Face Recognition. *Techno Nusa Mandiri*, *19*(2), 125–130. https://doi.org/10.33480/TECHNO.V19I2.3470

Rijal, Y., & Ariefianto, R. D. (2008). Deteksi Wajah Berbasis Segmentasi Model Warna Menggunakan Template Matching Pada Objek Bergerak. *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, 1907–5022. https://journal.uii.ac.id/Snati/article/view/891

Rodavia, M. R. D., Bernaldez, O., & Ballita, M. (2017). Web and mobile based facial recognition security system using Eigenfaces algorithm. *Proceedings of 2016 IEEE International Conference on Teaching, Assessment and Learning for Engineering, TALE 2016*, 86–92. https://doi.org/10.1109/TALE.2016.7851776

Samet, R., & Tanriverdi, M. (2017). Face recognition-based mobile automatic classroom attendance management system. *Proceedings - 2017 International Conference on Cyberworlds, CW 2017 - in Cooperation with: Eurographics Association International Federation for Information Processing ACM SIGGRAPH*, *2017-January*, 253–256. https://doi.org/10.1109/CW.2017.34

Satwikayana, S., Wibowo, S. A., & Vendyansyah, N. (2021). Sistem Presensi Mahasiswa Otomatis Pada Zoom Meeting Menggunakan Face Recognition Dengan Metode Convulitional Neural Network Berbasis Web. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *5*(2), 785–793. https://doi.org/10.36040/JATI.V5I2.3762

Septyanto, M. W., Sofyan, H., Jayadianti, H., Simanjuntak, O. S., & Dessyanto, B. P. (2020). Aplikasi Presensi Pengenalan Wajah Dengan Menggunakan Algoritma Haar Cascade Classifier. *Telematika : Jurnal Informatika Dan Teknologi Informasi*, *16*(2), 87–96. https://doi.org/10.31315/TELEMATIKA.V16I2.3182

Simaremare, H., & Kurniawan, A. (2016). Perbandingan Akurasi Pengenalan Wajah Menggunakan Metode LBPH dan Eigenface dalam Mengenali Tiga Wajah Sekaligus secara Real-Time. *SITEKIN: Jurnal Sains, Teknologi Dan Industri*, *14*(1), 66–71. https://doi.org/10.24014/SITEKIN.V14I1.2703

Sulistiyo, W., Suyanto, B., Hestiningsih, I., Mardiyono, M., & Sukamto, S. (2014). Rancang Bangun Prototipe Aplikasi Pengenalan Wajah untuk Sistem Absensi Alternatif dengan Metode Haar Like Feature dan Eigenface. *JTET (Jurnal Teknik Elektro Terapan)*, *3*(2), 93–98. https://doi.org/10.32497/JTET.V3I2.180