

Implementation of eigenface method and support vector machine for face recognition absence information system

Chakim Annubaha, Aris Puji Widodo, Kusworo Adi

Master of Information Systems, Postgraduate School, Diponegoro University, Semarang, Indonesia

Article Info

Article history:

Received Nov 22, 2021

Revised Mar 29, 2022

Accepted Apr 6, 2022

Keywords:

Attendance

Eigenface

Face recognition

Principal component analysis

Support vector machine

ABSTRACT

The student attendance system is what is needed in the process of recording attendance in learning and the development of student achievement. Currently several modern educational institutions have implemented a student attendance system using QR codes or fingerprints, but many still use the traditional system by calculating the number of students attending class. Based on these problems, the solution that can be given is to implement a student attendance system through face matching in the Android mobile application with Eigenface algorithm and support vector machine (SVM) algorithm. Eigenface using the principal component analysis (PCA) method can be used to reduce the dimensions of facial images so that they produce fewer variables and are easier to handle. The results obtained are then entered into a pattern classifier to determine the identity of the owner of the face. This study used 100 facial data as test data and training data. The system test results show that the use of Eigenface with SVM as a classifier can provide a fairly high level of accuracy. For facial images that were included in the training, 91% of the identification was correct.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Aris Puji Widodo

Master of Information Systems, Postgraduate School, Diponegoro University

Semarang, Central Java, Indonesia

Email: arispw@gmail.com

1. INTRODUCTION

The student attendance system constitutes the process of recording student attendance in the learning class [1]. This attendance system can serve as an evidence whether or not a student has been in class (present or absent) [2]. Student attendance record can be carried out in various ways. In this traditional or rural school, attendance has not been carried out, it is still using the traditional method, and the traditional method is still in use, that is, the teacher observes and counts all students one by one; if it is felt that there is a student not in class yet, the teacher will invite the student and verify the reason underlying his/her absence in the class. This causes the boarding school to not have real-time student data and student activities due to the absence of an automatic attendance information system.

One of the factors that influence student learning success is their presence [3], [4]. Student attendance is used as a requirement to take exams at several universities [5]. Fingerprint, face recognition, an introduction of palm veins are kind of the modern methods used to automatically detect student attendance. Other proposed attendance systems use radio frequency identification (RFID) are Arulogun *et al.* [6], Bhalla *et al.* [7], Hussain *et al.* [8], and Rjeib *et al.* [9]. Several attendance systems were developed in portable devices and smartphones [5], [10]. Previous researchers proposed an automatic student presence system based on facial recognition. The proposed system uses a camera to capture the faces of all students based on the principles of component analysis (PCA) and local binary patterns (LBP), and combines multiple

classifiers for face recognition by using LBP and Euclidean for 80 students get distance. Sayeed *et al.* proposed an attendance system using PCA and Euclidean distance for real-time face recognition [11]. Yusof *et al.* proposed a web-based real-time attendance system using facial recognition [12]. The proposed system uses Haar Cascade for face recognition and combines LBP for face recognition. The system proposed by Chintalapati *et al.* [13], Sayeed *et al.* [11], Yusof *et al.* [12] are inefficient because only using cameras to take pictures of students' faces one by one, while the systems proposed by Fuzail *et al.* [14], Mehta *et al.* [15] use a camera to capture all the faces of students in the class at once. This strategy can avoid queuing during the attendance process. However, presence systems using this strategy are less accurate in face recognition.

The use of RFID becomes another alternative for recording students' identities in the attendance system. The attendance process in the proposed system is very simple, that the students just need to scan students card containing barcodes, QR codes, or RFID to record students presence. However, there can also occur fake attendance because student cards containing barcodes, QR codes, or RFID and near field communication (NFC) phones are easily transferred from someone to others. Besides that the NFC system is not all available on smartphones, to solve this problem we can use palm veins to recognize students in the attendance system [16]. However, not all cameras can be used to take palm veins picture. However, the palm vein recognition system is only 78% accurate, we can conclude that the most accurate and best automatic students presence system is facial recognition.

The attendance system in this application implements the face recognition method into an Android-based information system. The face detection process becomes an important part. The presence of faces in the image can be detected by the system with various poses, light, expressions, barriers (glasses, mustache and beard) and sizes. Image processing functions to highlight certain features and reduce noise, so the image is ready to use for analysis purposes. Image processing includes skin detection, light normalization, and dimensional normalization [17].

This application is implemented using Eigenface method and support vector machine (SVM). In general, the face recognition procedure is divided into two stages. The first stage is face detection, which is the stage where the application is in search for faces in the camera-captured image to ensure that someone's face has been captured. The second stage is the identification stage or face recognition. At this stage, the results of the extraction of detected facial features are compared with facial images stored in the database. This comparison is made to allow the system to get the identification results of someone from the detected face. Giving this identity is carried out by conducting training on previous applications. The training step involves detecting faces, which is then processed using several image processing techniques to be able to be well received by the application, after which teaches the system whose face identity is detected using a training algorithm, which has also been available by the face recognition reference used in this research.

Based on the above-said background, the formulation of the problem in this research is to apply Eigenface and SVM algorithm in designing student attendance system application through recognizing face photos in an Android mobile application. The purpose of this research is to apply the SVM algorithm for facial photo classification and Eigenface algorithm in designing a student attendance system with face recognition in an Android mobile application. The author expects that the benefits of this research are the realization of an intelligent attendance system that will ease to monitor student discipline and real-time control of learning by administrators or student guardians since it simply uses an android smartphone connected to the internet. In the end, this research shall mean one of the efforts to get grace from Allah in order to utilize technology to make learning more easily controlled and effective.

2. MATERIAL AND METHOD

This application is implemented using Eigenface method and SVM for face recognition. Face recognition is a biometric technology widely applied in security systems besides eye retina recognition and fingerprint and iris recognition [18], [19]. Face recognition is also part of a computer vision application which is able to identify or verify a person from a digital image or video frame. One way to conduct this is by comparing the facial features from images with faces stored in a database [20]. Face recognition generally involves two stages. The first stage is face detection, where photos are searched to find any faces. The second stage is face recognition, where faces are detected and processed and compared with a database of recognized faces to determine the person.

Eigenface itself is a set of eigenvectors that are used in computer vision for facial recognition needs which form the basis set of all images used to build a covariance matrix [21]. This will reduce the dimensions of the face image into a smaller set of images that can represent the original image in the training dataset [22]. The reduction of the original face into eigenface is presented in Figure 1.

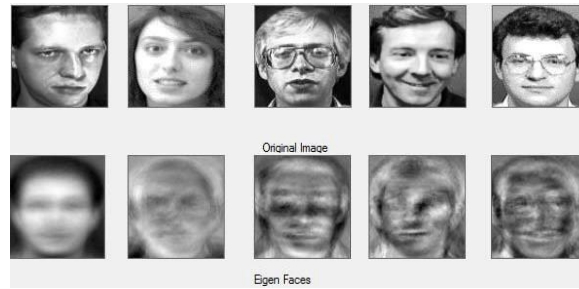


Figure 1. Original face and Eigenface

In its practice, to obtain Eigenface, techniques such as principal component analysis (PCA) are used [20]. PCA constitutes a dimensionality reduction technique which can produce low-dimensional data output from high-dimensional data input such as facial images. Before being applied to PCA, a dataset of N samples of facial images will be processed unfolding into a 1D vector. After that, m columns of eigenvectors will be generated (lower dimensions). The eigenvector will be used to process the folding into an Eigenface image. The process can be seen in Figure 2.

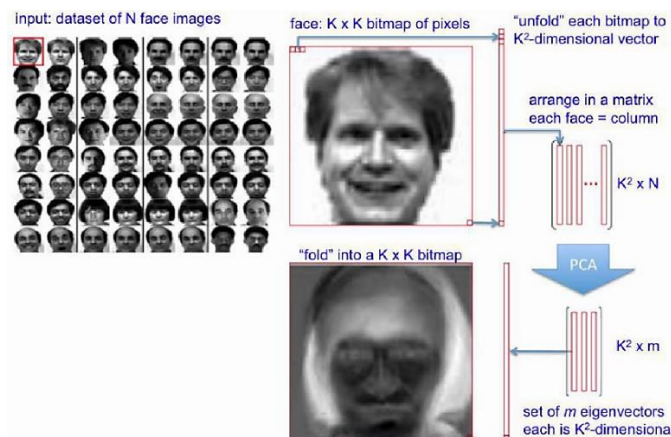


Figure 2. Eigenface illustration using PCA

Vapnik first introduced SVM as a pattern recognition field which superior and harmonious concept in 1992 [23]. SRM is a principle used in machine learning methods or SVM which aims to get a hyperplane link in an input space that is separated by two classes [24]. The process of classification and regression analysis often uses the SVM method which works by defining the boundary between the two classes with the furthest and closest data distances. The optimal hyperplane must be created in the input space obtained by measuring the boundary and finding the maximum point. While the distance from the hyperplane and the closest point in each class is called the margin. The next point is called the support vector [25].

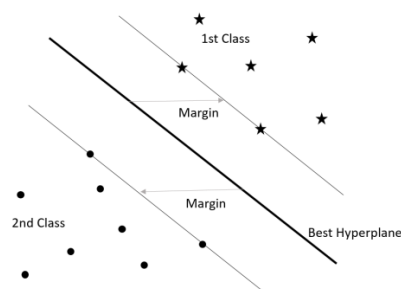


Figure 3. Illustration of class separation in SVM

From Figure 3, it can be explained that the optimization solution is solved by using the Lagrange function as (1).

$$L(w, b, \alpha) = \frac{1}{2}w^T w - \sum_{i=1}^n \alpha_i \{y_i [(w^T \cdot x_i) + b] - 1\} \tag{1}$$

The minimum value of the Lagrange function is given by (2).

$$\begin{aligned} \frac{\partial L}{\partial b} = 0 &\Rightarrow \sum_{i=1}^n \alpha_i y_i = 0 \\ \frac{\partial L}{\partial w} = 0 &\Rightarrow w = \sum_{i=1}^n \alpha_i x_i y_i \end{aligned} \tag{2}$$

To simplify equation (1) must be transformed into the Lagrange Multiplier function itself, so that (1) becomes (3).

$$L(w, b, \alpha) = \frac{1}{2}w^T w - \sum_{i=1}^n \alpha_i y_i (w^T \cdot x_i) + b \sum_{i=1}^n \alpha_i y_i + \sum_{i=1}^n \alpha_i \tag{3}$$

Based on (2) and (3), a dual problem is obtained [26]:

$$\max_{\alpha} L_D = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j$$

with the provision of:

$$\sum_{i=1}^n \alpha_i y_i = 0, \quad \alpha_i \geq 0$$

where $\{x_1, \dots, x_n\}$ is the input dataset, and $y_i \in \{+1, -1\}$ is the class label of the x_i data. With these results there will be a value of α_i for each training data. The training data that has a value of $\alpha_i > 0$ is a support vector while the rest has a value of $\alpha_i = 0$. We can determine the test data for x based on the value of the decision function if a solution to the quadratic programming problem has been found (value of α_i):

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i x_i x_d + b$$

x_i is the support vector, ns = the number of support vectors and x_d is the classified data [27].

2.1. Research data

The research was conducted using facial data samples from Al-Islah students, Pati Regency, Central Java. The facial data used in this study is data from facial images with a total of 11 types of faces with 20 images each with different backgrounds and expressions. Then for the process of adding classes, it was added 2 times with 15 images for the addition of the first class and 10 images for the addition of the second class. The following is the student's face data that will be used as a sample in the study. An example of the dataset used can be seen in Figure 4.



Figure 4. Al-Islah student face data

The validation process of the build system is performed using a k-fold cross-validation method. The data is divided into two parts, that is 80% of the image data is used for the training process, and 20% of the image data is used for the testing process. From this split, they were divided into 5 groups, 4 of which were used as training data and 1 as test data, and then 5 experiments were performed with different training and test data according to the following scheme in Table 1.

Table 1. Trial data

Trial data					
1	Test	Train	Train	Train	Train
2	Train	Test	Train	Train	Train
3	Train	Train	Test	Train	Train
4	Train	Train	Train	Test	Train
5	Train	Train	Train	Train	Test

2.2. Research materials and tools

Hardware, software, and face image data sets are the materials used for the development of the proposed attendance system. In testing the application, it was carried out on a laptop computer with Inter(R) Core(TM) i3-3217U CPU @ 2.30 GHz processor specifications, Windows 8.1 Home Single Language 64-bit operating system, screen size 1280×720 pixels, and 4.00 GB RAM.

This application is built on a desktop basis, using the Python programming language, OpenCV, and the MySQL relational database management system. OpenCV is used as a tool to implement Eigenface method which is used in the process of detecting and recognizing student faces to record attendance status in a lecture. The MySQL relational database management system is used as a database server to store all data in the application which contains login, student, photo, teacher, admin, and attendance tables.

2.3. System design

The system design in this research is divided into three, namely input, process and output. At the input stage, namely face data input. The process stage for processing face data using the selected method. Then, the output or results gained are facial images having been processed and the attendance information system application. An overview of the input, process, and output processes is presented in Figure 5.

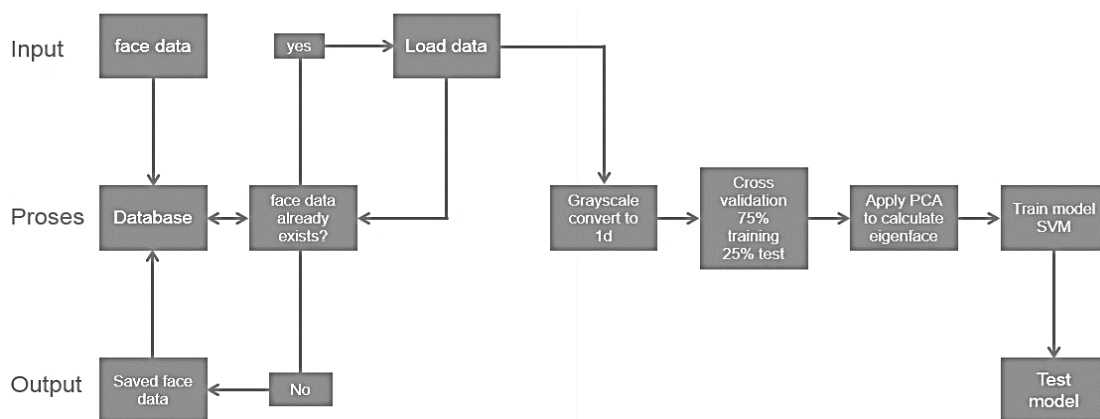


Figure 5. System framework

2.4. System implementation

At the implementation stage, the first thing to do in this research is to pre-process the image data, namely by getting a histogram on the image. The histogram is obtained by calculating the relative frequency of occurrence of the intensity in the image. Suppose an image has L degrees of gray (0 to $L-1$) or can be formulated as:

$$h_i = \frac{n_i}{n}, i = 0, 1, \dots, L - 1$$

where n_i is the number of pixels that have a degree of gray i , n is the total number of pixels in the image.

After the histogram is obtained, the next step is to learn or train the data. Because SVM can only classify data in a binary manner (only two classes) then the learning technique is carried out in a One-against-all way. To get the results of the classification with many classes. The learning scheme using the one-against-all method is as shown in Figure 6.

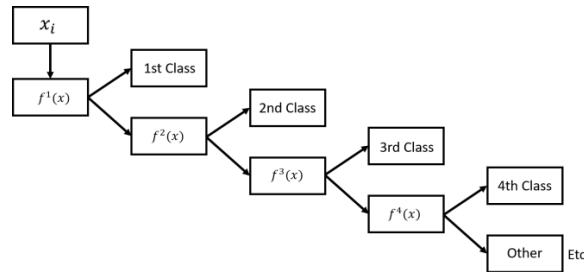


Figure 6. One-against-all classification scheme

In this training process, the kernel function used is a multi-kernel function which is obtained by combining kernel functions including linear kernels, polynomial kernels with order 3 and RBF kernels. This combination is carried out according to:

$$K(x, x') = \sum_{m=1}^M d_m K_m(x, x')$$

with obstacles:

$$d_m \geq 0, \quad \sum_{m=1}^M d_m = 1$$

where M is the number of kernel functions.

2.5. General view of the system

Figure 7 presents an android application login menu interface image containing a username and password form to log in to the attendance information system application. Figure 8 is the dashboard view when the user has successfully logged in, which is also used for attendance by clicking the scan button. Figure 9 is when the system is taking pictures of the user's face, so it is known whether or not the user data has been stored in the database.



Figure 7. Login mobile app

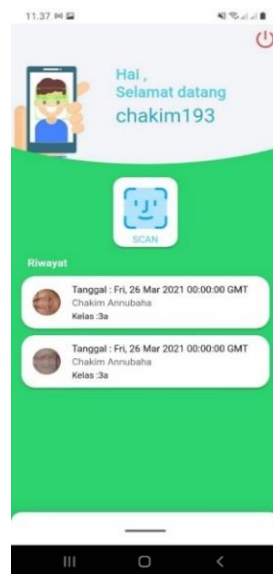


Figure 8. Dashboard view



Figure 9. Photo capture

3. RESULTS AND DISCUSSION

3.1. Application trial

Testing in this application is carried out by training the training data gained from 8 people, in which each face uses 10 training data, so the training data used are 80 data sets and training data. The following is the scene of the face scan results when the user fails or succeeds in making attendance. In Figure 10, the system is performing a face scanning to allow the results to be shown in Figures 11 and 12, stating whether the data are recognized or not. On the other hand, Figure 13 shows the attendance data stored in the database



Figure 10. Face scanning

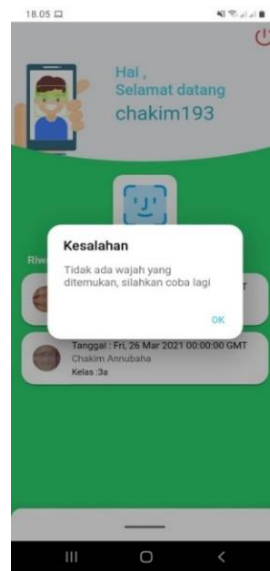


Figure 11. Unrecognized picture

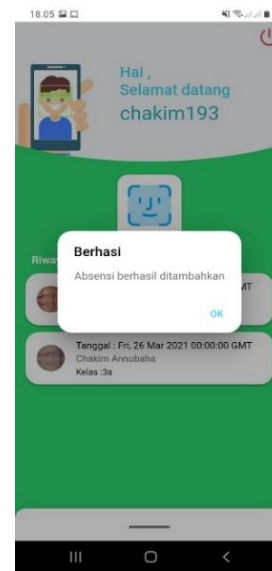


Figure 12. Recognized picture

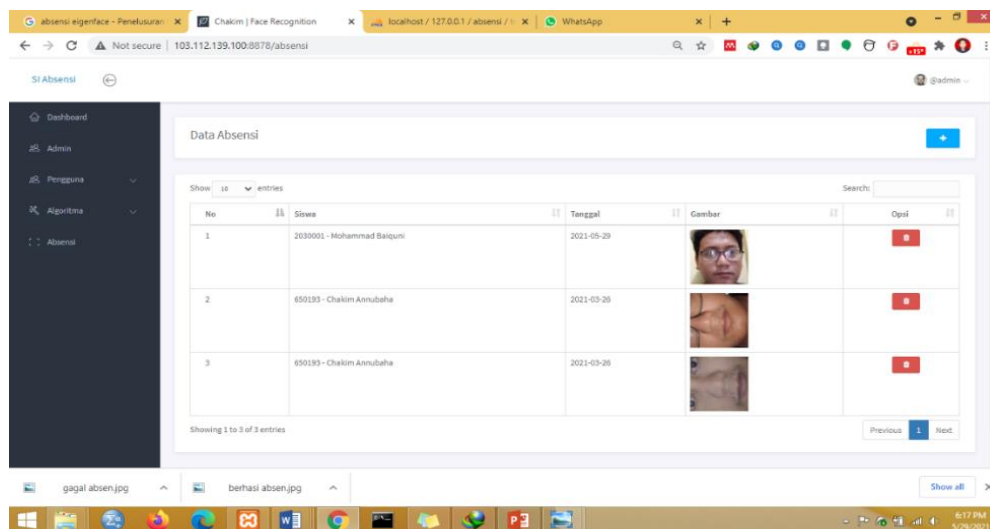


Figure 13. Attendance data database

3.2. Method implementation trial

The trial was carried out by calculating the time required by the system during the training process for the input image dataset. The training process was carried out three times, where the first training process was carried out on 8 types of faces with a total of 160 images. Then the next training was an increased learning process, namely training with adding one group of face types totalling 12 images. Then for training on updating or adding a second class, adding one face group totalling eight.

For the testing process, a trial was carried out by testing the testing database group consisting of 8 types of faces with five images each. So, that the total number of testing data in the initial 8 face group was 40 images. Next for the testing process on the results of the first class addition, the image data in the testing dataset is added three images so that the total number of faces tested on the classification results with the addition of the first class is 43 images as well as the addition of the second class is added the number of testing images is two image. From these experiments, it was found that a number of images were recognized correctly or not. From that number, precision, recall and accuracy were calculated with the following formula in Table 2.

Table 2. Confusion matrix of predicted value and actual value

		Actual value	
		True	False
Predicted value	True	True positive	False positive
	False	False negative	True negative

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \times 100\%$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \times 100\%$$

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{Total data}} \times 100\%$$

From the trials that have been carried out, the results are as shown by the graph of the test results for the introduction of 10 kinds of faces, the trials for the introduction of 11 kinds of faces or the addition of the first new class (update), and the results of the trials for the recognition of 12 kinds of faces or the addition of a new class (update) the second. Sequentially, Figure 14 shows the calculation of the trial time where for every addition of a new class, the results are not too much time difference between 10, 11, and 12 kinds of faces, which are shown in Figures 14(a)-(c). In Figure 15 is results obtained for precision, recall, and accuracy values for 10, 11, 12 kinds of faces, respectively, are in Figure 15(a)-(c).

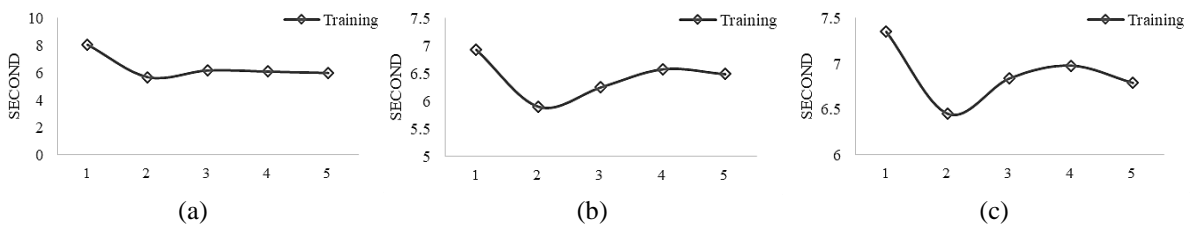


Figure 14. Computing time for the training process of (a) 10 kinds of faces, (b) first class update, and (c) second class update

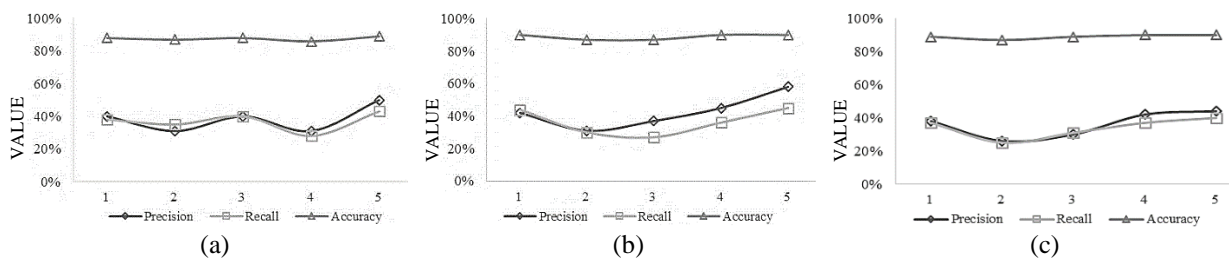


Figure 15. Value of precision, recall, accuracy of classification results of (a) 10 types of faces, (b) first class update, and (c) second class update

From the results shown in Figures 14 and 15, if the average is calculated, it can be seen that face recognition using SVM multi-kernel has a good performance with an average accuracy of 89% with 10 types of faces recognized. Then with the increasing learning method carried out with two additional classes, the average value of accuracy tends to remain unchanged at 89%. On the other hand, the precision value in the

first 10 classes has an average of 38% and does not change much when adding classes, namely 43% and 44%. For the recall scores in the first 10 classes, the average value reached 43% and then tended to decrease to 37% and 35% when the first and second classes were added.

When viewed in terms of computational time during the training process, the average time required for the system to train as many as 180 images is 5,993 seconds. While when there is an additional class which means that the input data being trained also increases the average time required for the training process is not much. changed, namely 6,494 seconds and 6,969 seconds. For convenience, the average value of the results from the trial can be seen in Table 3.

From the test results, it can be seen that facial recognition using a multi-kernel SVM with increased learning has good performance in terms of time as well as in terms of precision, recall and accuracy. From the performance results when adding classes, it shows that there is no significant change or decrease in performance even though the training carried out when adding classes does not use the overall data from the previous learning or in other words. Does not repeat new learning with the addition of new classes or data.

Table 3. Face recognition trial results

	Training Time	Precision	Recall	Accuracy
I	5.993	38%	43%	87%
II	6.494	43%	37%	88%
III	6.969	44%	35%	89%

Note:

I = Initial 10 class introduction

II = Introduction with the addition of 1 first class

III = Introduction with the addition of 1 second class

4. CONCLUSION

Based on the research having been conducted, it can be concluded that the implementation of the face detection information system using Eigenface algorithm and SVM produces a very good level of accuracy, namely the face accuracy for recognized face data output of 89%. Thereby, the designed system can be used as a reference for student attendance based on mobile android. Henceforth, for this research to be better, it is necessary to develop a system with other algorithms, so it can be compared between Eigenface and SVM algorithms with other algorithms.




REFERENCES

- [1] R. Joshi, V. V. Shete, and S. B. Somani, "Android based smart learning and attendance management system," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 4, no. 6, pp. 256–260, 2015.
- [2] U. K. Nkalo, E. O. Agwu, and E. C. Stanley, "Radio frequency identification (RFID) based attendance system with short message service (SMS) fackup," *J. Comput. Eng.*, vol. 21, no. 2, pp. 1–8, 2019.
- [3] J. K. Durfee, W. R. Loendorf, D. C. Richter, T. L. D. Geyer, and D. M. Munson, "A formal research study on correlating student attendance to student success," *ASEE Annu. Conf. Expo. Conf. Proc.*, no. June, 2012, doi: 10.18260/1-2--20810.
- [4] D. Sunaryono, J. Siswanto, and R. Anggoro, "An android based course attendance system using face recognition," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 33, no. 3, pp. 304–312, 2021, doi: 10.1016/j.jksuci.2019.01.006.
- [5] M. M. Islam, M. K. Hasan, M. M. Billah, and M. M. Uddin, "Development of smartphone-based student attendance system," *5th IEEE Reg. 10 Humanit. Technol. Conf. 2017, R10-HTC 2017*, vol. 2018-Janua, pp. 230–233, 2018, doi: 10.1109/R10-HTC.2017.8288945.
- [6] M. O. Arulogun, O. T., Olatunbosun, A., Fakolujo, O. A. and Olaniyi, "RFID-based students attendance management system," *Int. J. Sci. Eng. Res.*, vol. 4, no. 2, 2013, [Online]. Available: <http://www.ijser.org>.
- [7] V. Bhalla, T. Singla, A. Gahlot, and V. Gupta, "Bluetooth based attendance management system," *Int. J. Innov. Eng. Technol.*, vol. 3, no. 1, pp. 227–233, 2013.
- [8] E. Hussain, P. Dugar, V. Deka, and A. Hannan, "RFID based student attendance system," *Int. J. Comput. Appl. (0975 – 8887) Natl. Conf. cum Work. Bioinforma. Comput. Biol. NCWBCB- 2014*, vol. 3, no. 2, pp. 30–32, 2014.
- [9] H. D. Rjeib, N. S. Ali, A. Al Farawn, B. Al-Sadawi, and H. Alsharqi, "Attendance and information system using RFID and web-based application for academic sector," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 1, pp. 266–274, 2018, doi: 10.14569/IJACSA.2018.090137.
- [10] B. Soewito, F. L. Gaol, E. Simanjuntak, and F. E. Gunawan, "Attendance system on Android smartphone," *ICCEREC 2015 - Int. Conf. Control. Electron. Renew. Energy Commun.*, pp. 208–211, 2015, doi: 10.1109/ICCEREC.2015.7337046.
- [11] S. Sayeed, J. Hossen, S. M. A. Kalaiaarasi, V. Jayakumar, I. Yusof, and A. Samraj, "Real-time face recognition for attendance monitoring system," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 1, pp. 24–30, 2017.
- [12] Y. W. M. Yusof, M. A. M. Nasir, K. A. Othman, S. I. Suliman, S. Shahbudin, and R. Mohamad, "Real-time internet based attendance using face recognition system," *Int. J. Eng. Technol.*, vol. 7, no. 3, pp. 174–178, 2018, doi: 10.14419/ijet.v7i3.15.17524.
- [13] S. Chintalapati and M. V. Raghunadh, "Automated attendance management system based on face recognition algorithms," *2013 IEEE Int. Conf. Comput. Intell. Comput. Res. IEEE ICCIC 2013*, pp. 1–5, 2013, doi: 10.1109/ICCIC.2013.6724266.
- [14] M. Fuzail *et al.*, "Face detection system for attendance of class' Students," *Int. J. Multidiscip. Sci. Eng.*, vol. 5, no. 4, 2014.
- [15] P. Mehta and P. Tomar, "An efficient attendance management system based on face recognition using Matlab and Raspberry Pi 2," *Int. J. Eng. Technol. Sci. Res.*, vol. 3, no. 5, pp. 2394–3386, 2016.




- [16] S. Bayoumi, A. Aldayel, M. Alotaibi, M. Aldraihem, S. Alrashed, and S. Alzahrahi, "Class attendance system based-on palm vein as biometric information," *J. Theor. Appl. Inf. Technol.*, vol. 77, no. 2, pp. 266–272, 2015.
- [17] D. S. D. Kumar and P. V Rao, "Analysis and design of principal component analysis and hidden markov model for face recognition," *Procedia Mater. Sci.*, vol. 10, pp. 616–625, 2015.
- [18] J. Harikrishnan, A. Sudarsan, A. Sadashiv, and R. A. S. Ajai, "Vision-face recognition attendance monitoring system for surveillance using deep learning technology and computer vision," in *2019 international conference on vision towards emerging trends in communication and networking (ViTECoN)*, 2019, pp. 1–5.
- [19] S. M. Bah and F. Ming, "An improved face recognition algorithm and its application in attendance management system," *Array*, vol. 5, no. February 2019, p. 100014, 2020, doi: 10.1016/j.array.2019.100014.
- [20] Y. Bakhshi, S. Kaur, and P. Verma, "A study based on various face recognition algorithms," *Int. J. Comput. Appl.*, vol. 129, no. 13, pp. 16–20, 2015.
- [21] V. P. Kshirsagar, M. R. Baviskar, and M. E. Gaikwad, "Face recognition using eigenfaces," in *2011 3rd International Conference on Computer Research and Development*, 2011, vol. 2, pp. 302–306.
- [22] M. Slavkovic and D. Jevtic, "Face recognition using eigenface approach," *Serbian J. Electr. Eng.*, vol. 9, no. 1, pp. 121–130, 2012, doi: 10.2298/sjee1201121s.
- [23] S. Ghosh, A. Dasgupta, and A. Swetapadma, "A study on support vector machine based linear and non-linear pattern classification," in *2019 International Conference on Intelligent Sustainable Systems (ICISS)*, 2019, pp. 24–28.
- [24] M. A. Chandra and S. S. Bedi, "Survey on SVM and their application in image classification," *Int. J. Inf. Technol.*, vol. 13, no. 5, pp. 1–11, 2021.
- [25] O. F.Y, A. J.E.T, A. O, H. J. O, O. O, and A. J, "Supervised machine learning algorithms: classification and comparison," *Int. J. Comput. Trends Technol.*, vol. 48, no. 3, pp. 128–138, 2017, doi: 10.14445/22312803/ijctt-v48p126.
- [26] D. C. Toledo-Pérez, J. Rodríguez-Reséndiz, R. A. Gómez-Loenzo, and J. C. Jauregui-Correa, "Support vector machine-based EMG Signal Classification Techniques: A Review," *Appl. Sci.*, vol. 9, no. 20, p. 4402, 2019.
- [27] F. Bazikar, S. Ketabchi, and H. Moosaei, "DC programming and DCA for parametric-margin v-support vector machine," *Appl. Intell.*, vol. 50, no. 6, pp. 1763–1774, 2020.

BIOGRAPHIES OF AUTHORS






Chakim Annubaha    born in Pati, August 09, 1990. He is postgraduate student in the information systems at Diponegoro University, Indonesia. Completed the bachelor's degree in the informatics engineering at UIN Maulana Malik Ibrahim Malang, Indonesia. The research being conducted is examining the implementation of eigenface and support vector machines for attendance information systems. To contact the author via email chakimannubaha@gmail.com.



Aris Puji Widodo    born in Grobogan, April 01, 1974. He is student in doctoral education at the Gajah Mada University, Indonesia, majoring in Computer Sciences in 2015. Research area in the field of government interoperability and software engineering. To contact the author via email arispw@gmail.com.



Kusworo Adi    born in Temanggung, March 17, 1972. He is undergraduate education at Diponegoro University, Indonesia, majoring in physics from 1991 to 1996. Then the master's education at the Bandung Institute of Technology, Indonesia, majoring in electrical engineering from 2000 to 2002. Doctoral education at the Bandung Institute of Technology, Indonesia, majoring in electrical engineering in 2005 to 2010. Research area in the field of instrumentation and image processing. Several publications received in reputable international journals, especially in the field of instrumentation and image processing. The research being carried out now is about transportation safety by utilizing the development of instrumentation and image processing. Dr. Kusworo Adi is currently a member of the IEEE. To contact the author via email kusworoadi@gmail.com.