Face Recognition System using Facenet Algorithm for Employee Presence

1st Ferry Cahyono
Department of Electrical Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
ferry.18071@mhs.its.ac.id

2nd Wirawan
Department of Electrical Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
wirawan@ee.its.ac.id

3rd Reza Fuad Rachmadi
Department of Electrical Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
fuad@its.ac.id

***Abstract*— This study discusses the appropriate method to be applied in a presence system using faces by comparing two deep learning architectural models, they are FaceNet and Openface. FaceNet is a model developed by Google researchers that has the highest accuracy in face recognition. While Openface is a development from FaceNet that is trained with smaller datasets but has an accuracy that is almost equal to FaceNet. This will start by taking the employee's face into an image dataset. From the dataset, the face preprocessing will be performed by detecting, cropping, and resizing the face. Then extracting facial features into 128 dimensions using the FaceNet and Openface. With the Support Vector Machine (SVM), the classification of facial features will be carried out to obtain accuracy. To validate the model, 5 fold cross-validations are used. FaceNet accuracy results that obtained are higher with perfect accuracy that is 100%, while Openface only 93.33% accuracy. The implementation using the model with the highest accuracy (FaceNet) has the same results as the model testing that is 100% using the introduction threshold probability of 0.25.**

Keywords: face recognition, facenet, openface, embedding, svm

I. INTRODUCTION

In the advanced digital era requires an institution to adapt to technological advancements. Technology is able to replace from the previous manual switch to digital. Likewise, with the recording of employee presence, the government encourages all agencies to use it digitally. The Government through the Ministry of Administrative and Bureaucratic Reform has issued a letter Number B/2338/M.PANRB/06/2016 dated June 27th, 2016 concerning Optimization of the Use of Electronic-Based Attendance in Government Agencies[1].

In Malang Regency Government, electronic employee presence began in 2017 using the Fingerprint Electronic Presence. But, the presence using fingerprints have several obstacles, such as the fingerprints of employees who have changed due to injury or the employee has a thin fingerprint skin so that it complicates the identification process by the scanner.

In many studies, human characteristics that are often used for identification are faces because they are natural and do not require any direct contact with the sensors used[2].

Face recognition system is getting better from year to year. The use of Artificial Intelligence which is based on the approach of artificial neural networks is increasingly popular and gets near-perfect accuracy results. Until now many facial recognition technologies have been developed using deep learning. In 2015, Google researchers developed FaceNet which is a face recognition architecture model that has an almost perfect accuracy of 99.63%[3]. In 2016, Openface was also developed, a model developed from DeepFace and Google FaceNet that has been redesigned with several changes that were trained using the CASIA dataset and FaceScrub[4].

In this paper, we will compare model performance for facial recognition systems that have high accuracy. It aims to determine the level of accuracy of the model on employee face datasets. Our study will compare the FaceNet and Openface with input images extracted features using the two models, then face recognition will be carried out using the Support Vector Machine (SVM) classification to obtain the identity of employees. Face dataset uses 150 pictures from 15 employee identities. The result of classification is the accuracy which will be evaluated using k-fold cross-validation. The implementation that will be used is the model with the highest accuracy.

It is expected that in this study, a face recognition model that has high accuracy can be obtained so that there will be recommendations for the presence system that will be made.

II. RELATED RESEARCH

Research on facial recognition has long been done. In the 1990s a facial recognition method was discovered called Eigenfaces[7]. Until now, many approaches and techniques have been built on facial recognition systems. The new method found today has achieved very satisfying accuracy with the use of deep learning.

The development of facial recognition systems can be classified into four stages which are tested using the Labeled Face in the Wild (LFW) dataset as shown in Fig. 1. In the first stage, it began in 1991 by constructing low dimensional representations through certain distribution assumptions, for example, Eigenfaces obtained an accuracy of around 60%. In the second stage of 1997 based on local-features such as Gabor and had an accuracy of around 70%. In the third stage of 2010 based on local descriptors had an accuracy of around 82%. In the fourth stage, starting in 2014 based on deep learning including Convolutional Neural Network (CNN) had almost 100% accuracy[8].

978-1-7281-8131-8/20/$31.00 ©2020 IEEE

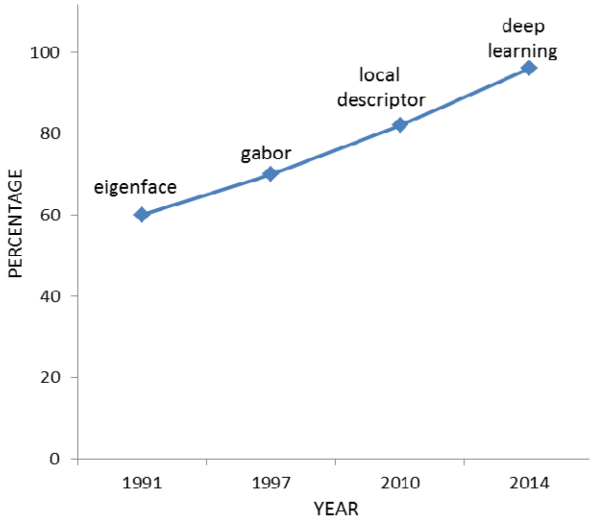

Fig. 1. Development of Facial Recognition System.

Some facial recognition systems using Deep Learning are DeepFace built by Facebook which has an accuracy of 97.35% with a standard deviation of 0.25% [9]. There is also, DeepID which produces higher accuracy with a relatively small difference of 97.45% with a standard deviation of 0.26% [10]. In 2015 FaceNet, which was built by Google researchers, had the highest accuracy of 99.63%[3] which was trained with a private dataset consisting of millions of images taken from social media[4]. While Openface is a model developed from DeepFace and GoogleNet that has been redesigned with several changes that are trained using the CASIA dataset and FaceScrub[4].

In 2019, a research on University Classroom Attendance System Using FaceNet and Support Vector Machine. The images obtained will be detected and the results extracted using Facenet and then classified using SVM for facial recognition. This study compares the performance of 3 deep learning model architectures, namely Facenet, VGG16 and Convolutional Neural Network (CNN) models. And the best accuracy results are obtained, namely the Facenet model with an accuracy rate of 99.6%. In this study, only looking at the performance of the deep learning architectural model, real time recognition and threshold determination for facial recognition has not been carried out[5].

In 2020, research on A Real-time Attendance System Using Deep-learning Face Recognition. The dataset used was 28 students and 10 pictures each were taken. Face recognition uses an euclidean distance with a threshold of less than 0.6 to be recognized and the accuracy is 95%. With deep learning architecture, very good accuracy is obtained using the Haar Cascade detection method[6].

III. METHODOLOGY

In this section, the face recognition process is carried out with five stages consisting of face input, preprocessing, feature extraction, classification, and face recognition results. Then it will be implemented into the recognition system in real-time by comparing faces in the database with faces originating from webcam input classified to obtain the identity of the employee, as shown in Fig. 2.

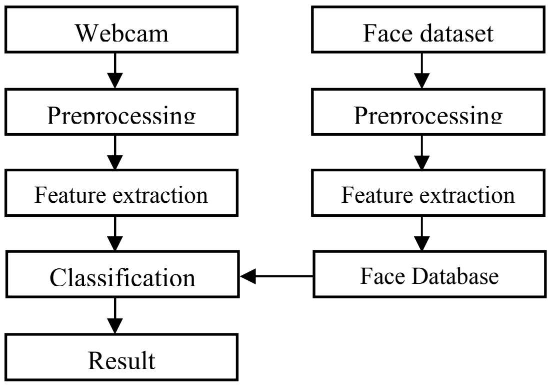

Fig. 2. Facial Recognition System

A. Dataset

Dataset of pictures taken was input images totaling 150 images of 15 employee identities. Face position was taken by facing the sensor camera.

B. Preprocessing

At the preprocessing stage, three processes were carried out, namely, detection, resizing, and cropping faces using the library's help from Multi-Task Cascaded Convolutional Neural (MTCNN). Detection was used to determine the position of the face in a given image which was then realized in the form of a bounding box. Then implemented the cropping based on the bounding box. After obtaining a face image would be scaled (resize) according to the input size of the model.

C. Feature extraction

1) Facenet

FaceNet was built by Google researchers using a Deep Convolutional Neural Network (DCNN) that maps images of a person's face into Euclidean spaces (collections of geometrical points) which are also called embedding. Embedding is obtained from the level of similarity and differences in faces, so that if the face has a similarity the value will get closer, and if the face is different the value will get farther [3].

In general, feature extraction using the FaceNet model as shown in Fig. 3, the input images will be getting into the deep learning architecture and then normalized L2 and the result is facial features (embedding) that are trained using Triplet Loss [3].

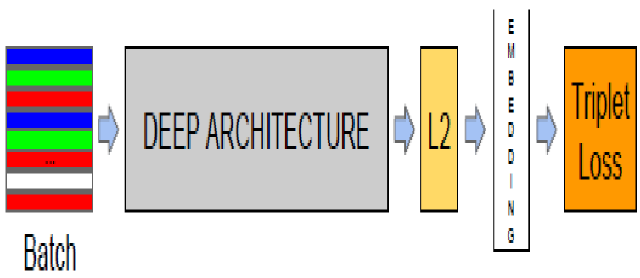

Fig. 3. Feature extractioin FaceNet[3].

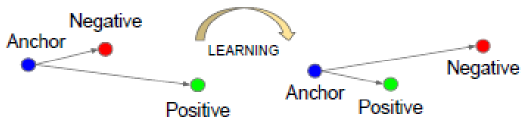

Fig. 4. Triplet Loss[3]

```
Layer (type)                    Output Shape          Param #
==============================================================
input_1 (InputLayer)            (None, 160, 160, 3)   0
Conv2d_1a_3x3 (Conv2D)          (None, 79, 79, 32)    864
Conv2d_1a_3x3_BatchNorm (BatchN (None, 79, 79, 32)    96
 ....
Dropout (Dropout)               (None, 1792)          0
Bottleneck (Dense)              (None, 128)           229376
Bottleneck_BatchNorm (BatchNorm (None, 128)           384
==============================================================
Total params: 22,808,144
Trainable params: 22,779,312
Non-trainable params: 28,832
```

Fig. 5. Architecture FaceNet

Feature extraction of the FaceNet architectural model using the Triplet Loss function as shown in Fig. 4, makes similar images closer, and makes different images farther away [3]. Feature extraction produced from FaceNet has high quality facial features (embedding) with 128 dimensions.

The FaceNet model was obtained from Github Hiroki Tanai [11]. The model is built using the Inception ResNet v1 model from TensorFlow which has been trained using the MS-Celeb-1M dataset [11]. This model has an input image of 160x160 pixels with 3 channels that produce 128-dimensional vectors, as shown in the FaceNet model architecture as shown in Fig. 5.

2) Openface

Openface was built with the FaceNet architecture with several changes, namely by modifying the parameters to be smaller and a smaller dataset that is 500 thousand images from a combination of CASIA and FaceScrub datasets. Openface still uses the same loss function that is Triplet Loss in making embedding [4].

The Openface model used has 96x96 pixel image input with 3 channels which produces 128 dimensional vectors with 6 times fewer parameters than FaceNet, which is 3,743,280 as shown in the Openface model architecture as shown in Fig. 6.

```
Layer (type)                    Output Shape          Param #
==============================================================
input_1 (InputLayer)            (None, 96, 96, 3)     0
zero_padding2d_1 (ZeroPadding2D (None, 102, 102, 3)   0
conv1 (Conv2D)                  (None, 48, 48, 64)    9472
 ...
flatten_1 (Flatten)             (None, 736)           0
dense_layer (Dense)             (None, 128)           94336
norm_layer (Lambda)             (None, 128)           0
==============================================================
Total params: 3,743,280
Trainable params: 3,733,968
Non-trainable params: 9,312
```

Fig. 6. Architecture Openface

D. Classification

Face recognition classification in this system uses SVM as shown in Fig. 7, because it has good performance and is widely used in face recognition [12]. SVM works by giving dividing boundaries to 2 adjacent classes. Margin is the closest point between Hyperplane and the closest point of each class which is then called the Support Vector Machine [13].

Linear Support Vector Machine (SVM) is very effective and widely used when separating face embedding vectors. By adjusting linear SVM to training data and setting 'kernel' attribute to 'linear'. In making predictions, will use the probability used to set it to be true[14].

In classifying the data is divided into 80% training data and 20% test data. The two groups of data will be processed by training and save the SVM classification model in a file. Classification performance will be evaluated using a confusion matrix.

E. System Testing

1) Confusion Matrix

The use of SVM classification will be analyzed for accuracy using a confusion matrix. Accuracy is calculated by dividing the number of correct predictions by the total amount of data [15].

2) k-Fold Cross Validation

Face recognition accuracy will be validated using k-fold cross validation by dividing the data into two parts, one part is used for training and the other part is used for validation. In k-fold cross validation, the test will be repeated k times. With the distribution of datasets, one part data validation and k-1 training data for model evaluation. [16] For k=5 as illustrated in Fig. 9.

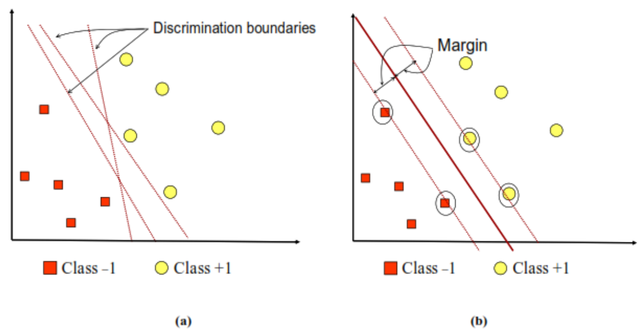

Fig. 7. Support Vector Machine

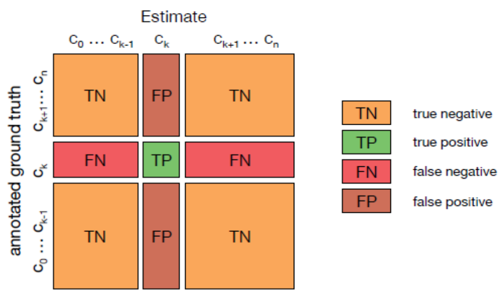

Fig. 8. Confusion matrix[15]

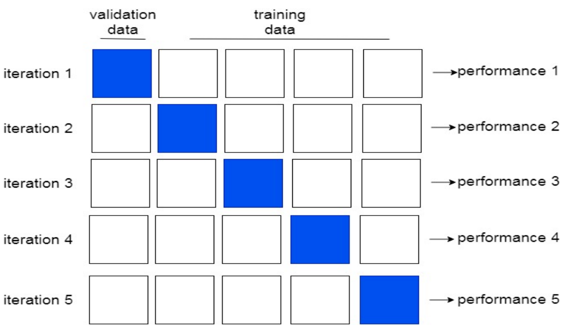

Fig. 9. k-Fold Cross Validation

At each iteration, a prediction will be made on the validation data and recorded performance metrics that have been determined, in this case using accuracy and also recorded the length of time in making predictions. In the final stage will be totaled and taken the average performance obtained to get the performance of the classification model.

In determining the k value, there is no formal provision, but usually 5 or 10. When choosing k with a large value, the set for sample data will be smaller [17]. This has been proven empirically by choosing k = 5 or k = 10 resulting in a small error rate [18].

IV. IMPLEMENTATION AND TESTING

In a comparison of FaceNet and Openface on a face recognition system for employee presence identification using a Core i5 3340M 2.70GHz laptop with 8GB RAM memory and Python 3 programming language. Testing on the model is done using a dataset and system testing is done by comparing the dataset and input from a webcam.

A. Model Testing

1) Preprocessing

In the preprocessing stage as shown in Fig. 10, three processes were carried out, namely detection, resizing and cropping the face using the library assistance from Multi-Task Cascaded Convolutional Neural (MTCNN). The image to be processed was given a bounding box which was then cropped based on the bounding box. After obtaining a face image would be scaled (resize) with a size in accordance with the input of the model used.

2) Feature Extraction

Feature extraction will make a face image into a facial feature (embedding) in 128 dimensions. This extraction process used input in the form of images with 3 channels (RGB), which produced 128-dimensional vectors as shown in Fig. 11.

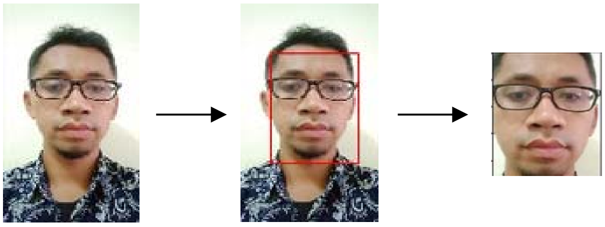

Fig. 10. Preprocessing Stage

```
array([ 1.8630524 , -1.2883751 , -0.26330343,  0.4060825 , -1.4930112 ,
        0.04194283,  0.5366742 , -1.5315478 , -0.8199895 ,  1.141684  ,
       -0.9820187 , -0.39845252, -0.26111835,  0.67693186, -1.1482742 ,
       -0.00436858,  0.10645946, -1.4834762 ,  0.03582224,  0.92423356,
        0.6945138 , -0.23059475, -0.8705194 ,  0.13284585,  0.9412558 ,
       -0.3150682 ,  0.25655016, -0.51082534, -0.4814728 , -0.02302771,
        0.10926945,  0.5947657 , -1.1671576 , -0.03692105,  0.24883932,
        0.05052355, -1.0284573 , -0.60186493,  1.096345  , -0.5197489 ,
        0.56739664,  0.8654664 ,  0.16737503, -1.2106476 ,  0.31544933,
       -1.08405   , -0.40912414,  2.3330898 , -1.4190598 ,  0.27882645,
       -0.55143285,  0.97984797,  0.19340041, -1.2868431 ,  0.6797951 ,
        0.10718144,  0.4374029 ,  0.21094814,  1.0192963 , -0.41130677,
        0.42843115,  0.17711377,  0.13484177,  1.3389393 ,  0.06584957,
        1.6536247 ,  0.7763515 ,  0.9670694 , -0.6451823 ,  1.2065457 ,
        0.5128179 , -0.59590405,  1.0535465 , -0.39617723, -0.4667322 ,
       -0.56861615,  0.11612426, -0.22631495, -0.1945281 , -1.2008042 ,
        0.0890996 , -0.90865064, -0.36028314,  1.2406405 ,  1.8607384 ,
       -2.063767  , -0.23035836, -1.86226   , -0.2555508 , -0.9054414 ,
        0.5506839 ,  0.09993919,  0.5578571 ,  1.2058746 ,  0.37803653,
       -0.09240614,  1.5897188 ,  1.4587421 , -1.5544238 ,  0.07799052,
        0.42279005,  0.26740518, -0.23773858, -0.65404725, -0.8052854 ,
       -0.22684993,  0.89914584,  0.01673314, -1.3139844 ,  0.21601504,
        0.45872065,  1.5960048 ,  2.3457112 , -0.78951114, -0.45617726,
        0.2467384 , -0.6505567 ,  1.8112195 , -0.82550323,  0.6479339 ,
        1.5175676 ,  0.03704251,  1.0670711 , -0.12793806,  1.3369213 ,
       -1.2630515 ,  0.3943154 , -0.15486093], dtype=float32)
```

Fig. 11. 128 dimensional face features

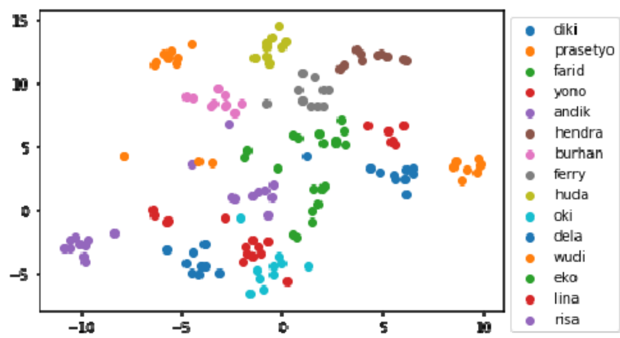

Fig. 12. Openface Visualization

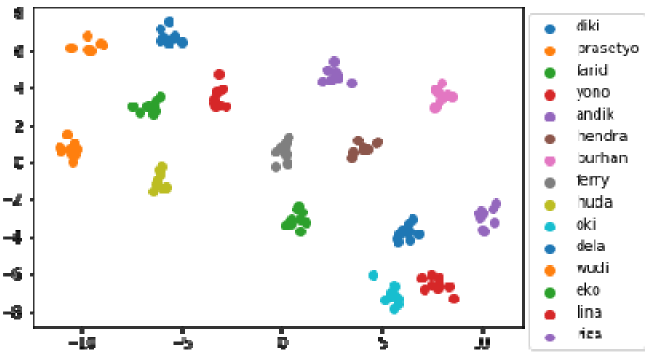

Fig. 13. FaceNet Visualization

Visualize 128-dimensional facial features into 2-dimensional space by comparing the results obtained from the FaceNet and Openface.

From the visualization results in Figs. 12 and 13 show that the identity of employees using the FaceNet was better than the Openface in separating between classes.

3) Classification

After facial features (embedding) were obtained, the process for facial recognition was done by classification. The classification used SVM with linear kernels and true probability settings.

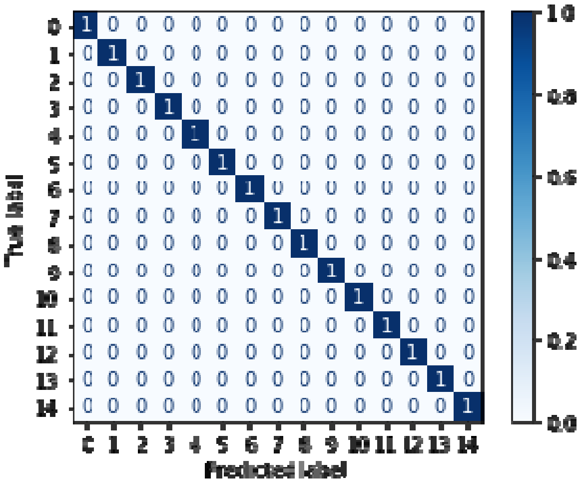

Fig. 14. FaceNet confusion matrix

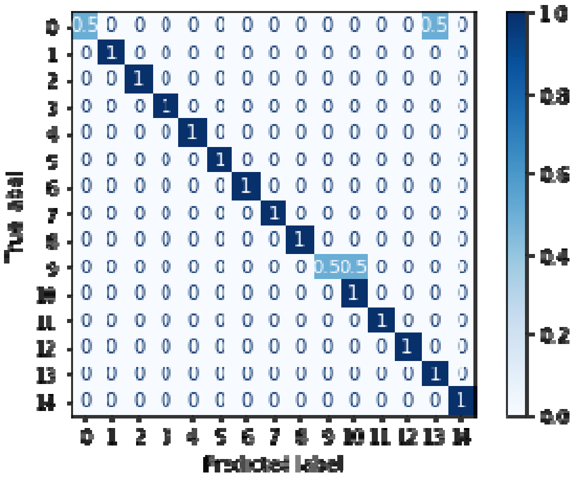

Fig. 15. Openface confusion matrix

To get the accuracy of the classification used confusion matrix. The results of Facenet are shown in Fig. 14.

The dataset used is divided into 80% training data and 20% testing data. From the 150 image datasets, 30 pictures were obtained from the test data and 120 pictures for training data. From the results of the confusion matrix, all exact predictions with no errors of accuracy can be calculated by dividing the number of correct predictions by the total amount of data multiplied by 100%.

Accuracy = 30 / 30 x 100%= 100%

The results of the confusion matrix from Openface are shown in Fig. 15:

From the results of the confusion matrix, it was found that one class failed to predict, namely the 10th class so that the number of predictions was correct 28 out of 30 total images.

Accuracy = 28/30 x 100% = 93,33%

To produce the best model, validation testing is done on the SVM classification using 5 fold cross validations.

Testing using 5 fold cross validation, dataset is divided into 5 subsets, each of which consists of 30 images. The repetition is done 5 times and the results of the 5 Fold Cross Validation test as shown in Table I. The results of SVM validation in the dataset obtained the same results using both the confusion matrix and 5 fold cross validation for both models, FaceNet has 100% accuracy and Openface has an accuracy of 93.33%.

TABLE I. 5-FOLD CROSS VALIDATION FACENET & OPENFACE

Fold	Accuracy	
	Openface	*FaceNet*
1	100%	100%
2	93,33%	100%
3	93,33%	100%
4	86,67%	100%
5	83,33%	100%
Average	93,33%	100%

B. System Testing

System testing was done using a model that gets the highest accuracy, it was FaceNet. It was done in two conditions, namely all faces that were in the database (faces were recognized) and faces that were not in the database (faces were not recognized).

Testing FaceNet on recognized faces, the results obtained in Table II.

TABLE II. TESTING FACENET ON RECOGNIZED FACES

No	Name	Recognized		Probability
		True	*False*	
1	andik	Yes	-	0,35-0,37
2	burhan	Yes	-	0,36-0,38
3	dela	Yes	-	0,41-0,43
4	diki	Yes	-	0,41-0,44
5	eko	Yes	-	0,39-0,41
6	farid	Yes	-	0,33-0,35
7	ferry	Yes	-	0,39-0,41
8	hendra	Yes	-	0,48-0,50
9	huda	Yes	-	0,35-0,37
10	lina	Yes	-	0,49-0,51
11	oki	Yes	-	0,42-0,43
12	prasetyo	Yes	-	0,25-0,27
13	risa	Yes	-	0,36-0,39
14	wudi	Yes	-	0,35-0,37
15	yono	Yes	-	0,34-0,36

Accuracy in facial recognition systems was obtained by dividing the correct value compared to the entire test data and the result was 100%.

Accuracy = 15/15 x 100% = 100%

Testing FaceNet on unrecognized faces, the results as shown in Table III.

TABLE III. TESTING FACENET ON UNRECOGNIZED FACES

No	Name	Recognized		Probability
		True	***False***	
1	Unknown1	-	Yes	0,13-0,14
2	Unknown2	-	Yes	0,16-0,17
3	Unknown3	-	Yes	0,18-0,19
4	Unknown4	-	Yes	0,16-0,18
5	Unknown5	-	Yes	0,15-0,17
6	Unknown6	-	Yes	0,11-0,13
7	Unknown7	-	Yes	0,14-0,15
8	Unknown8	-	Yes	0,15-0,17
9	Unknown9	-	Yes	0,09-0,11
10	Unknown10	-	Yes	0,12-0,14
11	Unknown11	-	Yes	0,13-0,16
12	Unknown12	-	Yes	0,14-0,15
13	Unknown13	-	Yes	0,11-0,13
14	Unknown14	-	Yes	0,10-0,12
15	Unknown15	-	Yes	0,12-0,15

From the two observation tables, it was found that recognizable faces had a probability range between 0.25 to 0.51 while unrecognized faces had a probability range between 0.09 to 0.17. The threshold to be recognized is taken from the smallest recognizable face value, which is 0.25 as shown in Fig. 16.

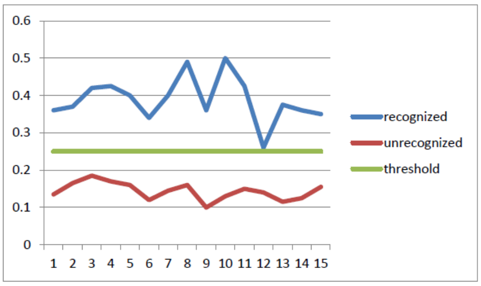

Fig. 16. Facial recognition threshold

CONCLUSION

From the tests conducted, it can be concluded that :

1. Face recognition using the FaceNet and Support Vector Machine (SVM) had perfect accuracy which was 100%, it was better than the accuracy obtained from the Openface which had an accuracy value of 93.3%. These results were in accordance with the baseline accuracy of the introduction of each model.
2. Implementation using the model with the highest accuracy, FaceNet, obtained an accuracy of 100% by taking the threshold of the probability introduction of 0.25.

REFERENCES

[1] MENPANRB. Use of Electronic-Based Attendance in Government Agencies, 2016. [Online].https://www.menpan.go.id/site/berita-terkini/info-terkini/surat-meteri-panrb-tentang-penggunaan-absensi-berbasis-elektronik-di-lingkungan-instansi-pemerintah.[diakses 12 Juni. 2020]

[2] Ben Aoun N, Mejdoub M, Ben Amar C. 2014. Graph-based approach for human action recognition using spatio-temporal features. J Vis Commun Image Represent.

[3] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 815–823, 2015.

[4] B. Amos, B. Ludwiczuk, and M. Satyanarayanan, “Openface: a general-purpose face recognition library with mobile applica-tions,” CMU-CS-16-118, CMU School of Computer Science,2016.

[5] T. Nyein and A. N. Oo, “University Classroom Attendance System Using FaceNet and Support Vector Machine,” in 2019 International Conference on Advanced Information Technologies (ICAIT), 2019, pp. 171–176.

[6] W. Kuang and A. Baul, “A Real-time Attendance System Using Deep-learning Face Recognition,” in 2020 ASEE Virtual Annual Conference Experience, 2020.

[7] Matthew Turk and Alex Pentland. Eigenfaces for recognition.Journal of cognitive neuro-science, 3(1):71–86, 1991.

[8] Wang M, Deng W. 2018 Deep face recognition: a survey.

[9] Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. Deep-face: Closing the gap to human-level performance in face verification. InCVPR, 2014.

[10] Sun, Y., Wang, X., and Tang, X. Deep learning face representation from predicting 10,000 classes. InCVPR,2014b.

[11] H. Taniai, “keras-facenet,” 2018. https://github.com/nyoki-mtl/keras-facenet (accessed Mar. 01, 2020).

[12] P. Navin, Y. Singh, "Support vector machines for face recognition", "IRJET Volume: 02 Issue: 08. 1521, November 2015.

[13] Nugroho, A.S., Witarto, A. B., Handoko,D., 2003. Support Vector Machine Teori dan aplikasinya dalam Bioinformatika, Kuliah umum Ilmu Komputer.com.

[14] J. Brownlee, Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python. Machine Learning Mastery, 2019.

[15] Kruger, F. 2016. Activity, Context, and Plan Recognition with Computational Causal Behaviour Models. Ph.D. Dissertation, University of Rostock.

[16] P. Refaeilzadeh, L. Tang dan H. Liu, “Cross-Validation,” 6 11 2008.

[17] Max Kuhn and Kjell Johnson.Applied Predictive Modeling. Springer, New York, 2013.

[18] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani.An introduction to statistical learning, volume 112. Springer, 2013.