Jurnal SimanteC
Vol. 13, No. 2 Juni 2025
P-ISSN : 2088-2130
E-ISSN : 2502-4884

Optimization of facial recognition authentication system using InceptionResNetV1 with Pretrained VGGFACE2

[1]Ellexia Leonie Gunawan, [2] I Gede Susrama Mas Diyasa*, [3] Wahyu Syaifullah Jauharis Saputra

[1,3]Program Studi Sains Data, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jawa Timur
[2]Program Magister Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jawa Timur
Jl. Rungkut Madya, Gn. Anyar, Kec. Gn. Anyar, Surabaya, Jawa Timur
*e-mail: igsusrama.if@upnjatim.ac.id

Abstract

Face recognition as a biometric authentication method continues to evolve due to its high security and ease of use. However, training models from scratch faces challenges such as the need for large datasets and high computational resources. This study aims to optimize the face authentication system using the InceptionResNetV1 architecture with a transfer learning approach from the pretrained VGGFace2 model and to compare its performance with CASIA-WebFace. Face detection is conducted using YOLOv8, face embeddings are generated by InceptionResNetV1, and authentication is performed by calculating the Euclidean distance between embeddings. Face data were collected from university students and divided into training and testing datasets. Performance evaluation includes accuracy, precision, recall, F1-score, and the confusion matrix. The results show that the VGGFace2 model achieved an accuracy of 98.75%, a recall of 100%, and an F1-score of 99.26%, with no False Negatives, while CASIA-WebFace achieved an accuracy of 86.25% with a recall of 85.07%. The main contribution of this study is to demonstrate that the use of transfer learning with the pretrained VGGFace2 model can significantly improve the accuracy of face authentication systems and to show its effectiveness for developing systems with limited data and computational resources. This study contributes by highlighting the superiority of the pretrained VGGFace2 model in face authentication systems and emphasizing the effectiveness of transfer learning for implementing accurate systems under resource constraints.

Keywords: *Authentication System, InceptionResNetV1, Face Recognition, Transfer Learning, VGGFace2*

1 INTRODUCTION

Biometric authentication has increasingly become a reliable approach in enhancing security systems, especially due to the limitations of traditional methods such as passwords and ID cards, which are vulnerable to theft, forgery, and misuse [1][2]. One of the most widely used biometric methods is facial recognition. This technology enables the unique identification of individuals based on facial features, and has been widely adopted across various domains such as security systems [3], electronic payment systems [4], and digital attendance systems [5].

Conventional facial recognition methods, such as Eigenfaces [6] and Fisherfaces [7], which are based on Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) techniques, were widely utilized during the early development of this system. However, these approaches have limitations in handling variations in lighting, facial poses, and expressions, which frequently occur in real-world conditions [10][11]. As artificial intelligence technology has advanced, deep learning architectures, particularly Convolutional Neural Networks (CNNs), have emerged as more adaptive solutions [10]. CNNs possess the capability to extract deep and representative facial features, making them more accurate and robust under various visual conditions [11][12].

One of the current **state-of-the-art** approaches is the use of pretrained CNN-based models—models that have been previously trained on large and complex datasets [13]. These models are able to transfer prior knowledge from their training data, making the training process more efficient and yielding more stable and accurate results, even when applied to different domains.

Commonly used pretrained models include VGGFace, VGGFace2, and CASIA-WebFace, employing popular architectures such as ResNet, Inception, and InceptionResNet.

Despite CNNs being proven effective, their implementation requires large datasets [14] and high computational power [15], which poses a challenge in deploying face-based authentication systems in environments with limited hardware capabilities [16]. Transfer learning offers an efficient alternative to address this issue [17]. By leveraging weights from models trained on large datasets, such as VGGFace2, the training process can be accelerated while maintaining accuracy, even when using smaller datasets [18]. The VGGFace2 dataset itself contains over 3 million facial images of diverse individuals with a variety of poses and lighting conditions, making it highly representative for facial recognition model training [19].

Numerous studies have utilized pretrained VGGFace2 on different model architectures and reported promising results in facial recognition tasks [20][21][22]. However, few studies have specifically examined the effectiveness of the InceptionResNetV1 architecture with pretrained VGGFace2 in the context of face-based user authentication. In fact, the InceptionResNetV1 architecture offers advantages in processing efficiency and performance stability, making it an ideal candidate for real-time authentication systems.

This study was conducted with the objective of exploring and optimizing a web-based facial authentication system by leveraging the InceptionResNetV1 architecture and transfer learning from VGGFace2. The specific goals of this research include: (1) Implementing and evaluating the InceptionResNetV1 model for facial authentication, (2) Comparing the performance of pretrained models from VGGFace2 and CASIA-WebFace to assess their effectiveness, and (3) Conducting performance evaluation using a confusion matrix as the performance indicator.

The contribution of this study is to provide a facial authentication solution that is not only accurate but also computationally efficient and widely implementable on web-based systems. Furthermore, this research offers an empirical overview of the optimal use of pretrained models for facial authentication tasks, serving as a reference for future studies and the development of biometric systems.

2 LITERATURE REVIEW

Research on face authentication systems based on deep learning has grown rapidly in recent years. One of the milestone approaches is FaceNet, introduced by Schroff et al. This model utilizes triplet loss to generate facial embeddings in a vector space, allowing the system to distinguish identities by maximizing the distance between different individuals and minimizing the distance between images of the same individual. This approach has proven highly effective in facial verification tasks and has served as a foundation for many subsequent studies. This advancement was further supported by the work of Anwarul et al. [23], who demonstrated the effectiveness of Convolutional Neural Networks (CNNs) in generating facial embeddings with high accuracy. The method has shown excellent performance in renowned benchmarks such as Labeled Faces in the Wild (LFW), reinforcing CNN-based embeddings as a primary approach in modern facial recognition systems.

Beyond identification, CNNs have also been proven effective in facial expression classification. Riyantoko et al. [24] employed a combination of CNN and Haar-Cascade to classify seven facial expressions using the FER2013 dataset, showing that CNNs can effectively extract visual features even under varied lighting conditions and expressions. This approach is highly relevant to facial authentication, which also requires resilience to expression variability. A similar study by Diyasa et al. [25] developed a multi-face recognition system for inmate detection in detention rooms using CNN and the Haar Cascade Classifier. Although aimed at security surveillance, this method is relevant for real-time facial recognition, particularly in detecting multiple individuals, managing viewpoint angles, and handling uneven lighting. Such approaches are highly applicable in real-world environments where simultaneous detection of multiple faces and robustness to pose and lighting variations are essential.

The performance of a facial recognition system is heavily influenced by the quality and diversity of the training dataset. VGGFace2 stands out as a comprehensive dataset encompassing a wide range of facial expressions, lighting conditions, ages, and poses across thousands of individuals. Research by Daoud et al. [26] revealed that models trained on VGGFace2 exhibit stronger robustness against facial variations, making them suitable for dynamic real-world scenarios. The dataset enables deep learning models to learn more accurate facial representations, even under challenging lighting or extreme pose variations. With over 3.31 million images from 9,131 subjects, VGGFace2 serves as a robust foundation for enhancing the performance of modern facial recognition systems, and is considered one of the most prominent datasets in this field.

In addition to datasets, the model architecture also plays a crucial role in system accuracy and efficiency. The InceptionResNetV1 model, which combines the Inception and ResNet architectures, has proven to be highly effective in handling facial data complexity while maintaining computational efficiency. Hidayati et al. [27] showed that the Inception-ResNet model can achieve high accuracy despite having a smaller parameter size compared to other architectures such as ResNet-101 or VGG16. This advantage makes Inception-ResNet an attractive choice for facial shape recognition, particularly in applications that demand a balance between efficiency and performance. However, the study was limited to conventional testing using separate test data and did not explore its application in facial authentication systems.

Based on this review, several research gaps are identified. First, the application of the InceptionResNetV1 model trained on large datasets such as VGGFace2 and CASIA-WebFace for face authentication has not been widely explored. Second, most previous studies focused on classification or identity verification rather than embedding-based authentication using one or a few images per individual, as is commonly used in login systems. Third, the impact of the pretraining dataset on the quality of generated embeddings has not been extensively analyzed in a comparative manner.

This study aims to fill those gaps by implementing the InceptionResNetV1 model pretrained on two large datasets—VGGFace2 and CASIA-WebFace—to generate facial embeddings in an authentication system. The evaluation process involves computing the Euclidean distance between embeddings and testing various threshold values to assess accuracy. Through this approach, the research is expected to contribute to the optimization of embedding-based face authentication systems and to demonstrate how the selection of a pretraining dataset influences the overall effectiveness of the system.

3 RESEARCH METHOD

The face recognition process in this study is systematically illustrated in Figure 1, which outlines the main stages starting from data collection, pre-processing, feature extraction, to matching and model evaluation.

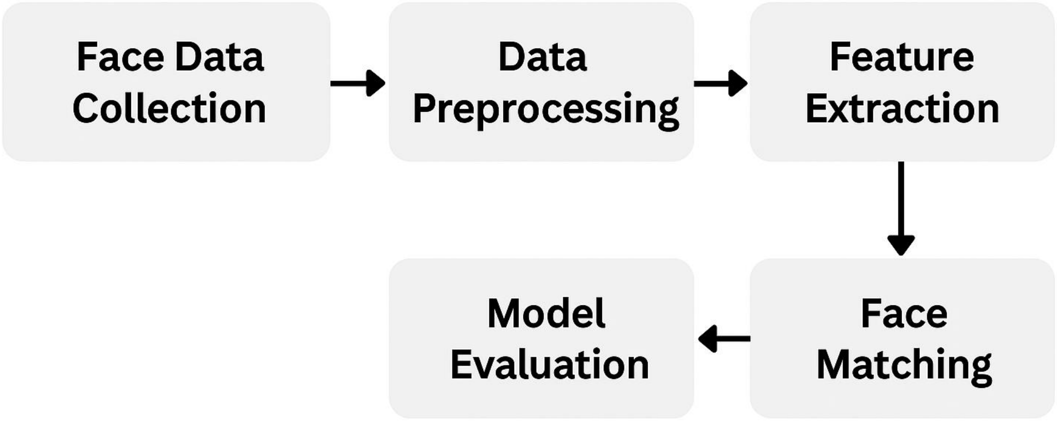


Figure 1. General research workflow

Figure 1 illustrates a flow diagram that shows how each stage is interconnected to produce an optimal face authentication system. In the initial stage, facial data is collected and then processed through face detection and image normalization. The processed facial images are then used in the feature extraction stage using the InceptionResNetV1 model, which has been pretrained on the VGGFace2 dataset. Once the facial embeddings are generated, they are matched with the embeddings in the database using Euclidean distance. The matching results are then evaluated using various performance metrics, which are analyzed to determine the effectiveness of the developed authentication system. The following is a step-by-step explanation of the procedures carried out in this research:

a. Data Collection

This study used facial data from 67 students of Universitas Pembangunan Nasional "Veteran" Jawa Timur. The data collection process was carried out by recording short facial videos of 5–10 seconds per individual, covering various angles such as frontal, left side, right side, top, and bottom views. This approach aimed to capture variations in expression and head pose to enhance the model's robustness to facial pose changes.

The recordings were made using the front camera of an Oppo Reno 3 smartphone with a resolution of 44 MP, allowing high-resolution facial image capture under various natural lighting conditions. After recording, each video was converted into a sequence of static image frames at 10-frame intervals. From each video, approximately 15 to 25 images per person were obtained, depending on the duration and variety of recorded expressions. A quality selection process was then carried out to remove blurry or noisy images, ensuring that only clear facial images were used for model training and testing.

As part of the system evaluation, the researchers also added 13 facial images of individuals who were not included in the training database, representing about 20% of the total subjects. These additional images served as negative samples to test the system's ability to detect unregistered faces, which is important for authentication and security. An example of the collected facial data used in this study is shown in Figure 2.

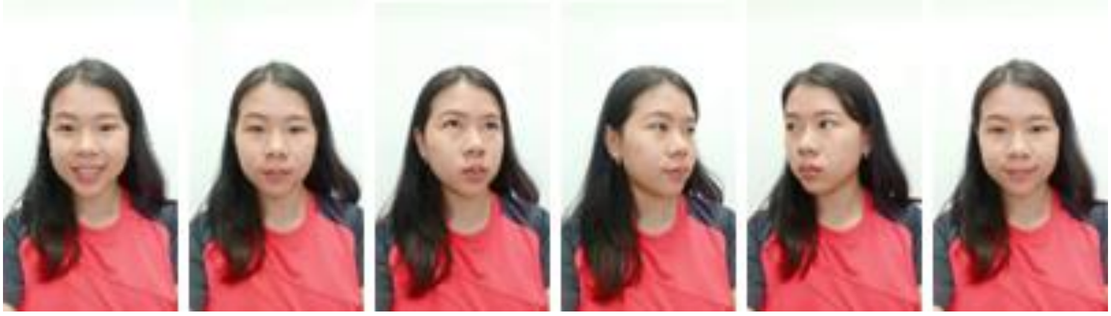


Figure 2. Example of facial data

Figure 2 shows the variation of facial images collected in this study, covering various viewpoints to improve the model's robustness against facial variation in real-world conditions.

b. Data Pre-processing

After the data was obtained in the form of images, the next step was to split the dataset into a training set and a testing set. This division was done manually, where one image from each individual was selected as test data (testing set), while the remaining images were used for training the model (training set). This strategy aims to ensure that each individual is represented in the evaluation process, enabling performance testing on an individual basis [30].

Each facial image then underwent several pre-processing stages to ensure optimal data quality before being used in the training and testing processes. The first step was face detection and cropping, which was performed automatically using the YOLOv8 Face Detection model. This process detects the location of the face in the image and crops the relevant area so that only the facial region is used in the next stages. This cropping step is crucial to eliminate unnecessary background and enhance the focus of feature extraction [31]. After cropping, all images were resized to 160×160 pixels.

To improve the model's stability in handling lighting variations and contrast differences, pixel normalization was applied to each image. Normalization was performed using equation (1) as follows:

$$N = \frac{image - 0.5}{0.5} \qquad (1)$$

Equation (1) transforms the pixel value range of the image from [0, 1] to [-1, 1], which is a common prerequisite in neural network training to accelerate convergence and improve model accuracy [32]. By applying these pre-processing steps, the facial data becomes more structured, clean, and consistent, making it ready for use in the training and evaluation phases of the face recognition-based authentication system. A visualization of the pre-processing results is shown in Figure 3.

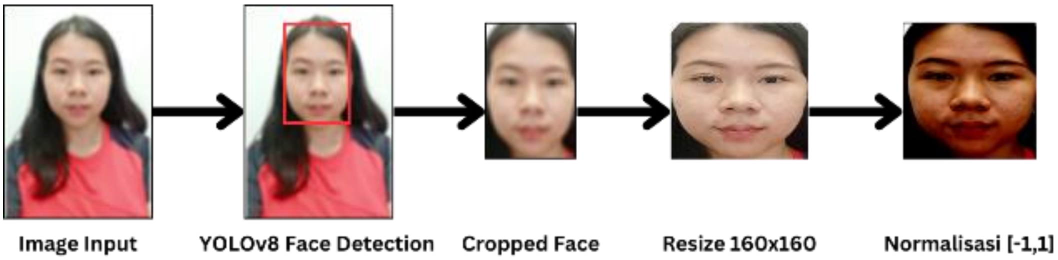


Figure 3. Data Pre-processing

Figure 3 shows the results of the pre-processing stage, which includes face detection, facial area cropping, and image normalization before proceeding to the feature extraction stage.

c. **Feature Extraction**

The feature extraction stage is a key process in face recognition systems, aimed at obtaining a numerical representation of each face in the form of embeddings [33]. In this study, the extraction process was carried out using the InceptionResNetV1 architecture, which was pretrained on a large-scale dataset. The feature extraction process applied in this study is illustrated in Figure 4.

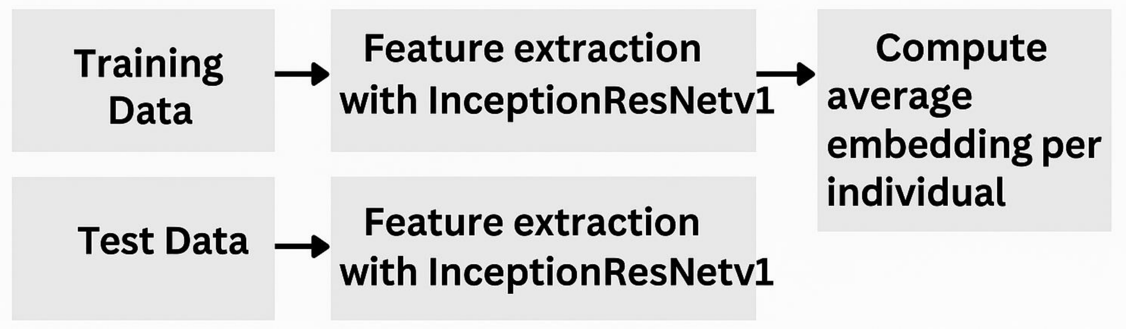

Training Data
Feature extraction with InceptionResNetv1
Compute average embedding per individual
Test Data
Feature extraction with InceptionResNetv1

Figure 4. Feature Extraction Process

In the diagram in Figure 4, there is a distinction in the treatment of training and testing data. For the training data, after facial features are extracted, all embeddings from a single individual are averaged to obtain a stable and representative feature vector. This approach aims to reduce variability caused by lighting, viewpoints, or facial expressions in the training images. In contrast, for the testing data, feature extraction is performed independently on each image without averaging. This simulates real-world authentication conditions, where the system must be able to recognize a person based on a single image.

1) InceptionResNetV1 Architecture

InceptionResNetV1 is an artificial neural network architecture that combines two advanced approaches: Inception and Residual Network (ResNet). This model was developed to address challenges in extracting complex features from images, particularly for facial recognition tasks [34]. The Inception architecture was first introduced through GoogLeNet (InceptionV1), with its core concept being the use of multiple convolution filters in parallel within a single module. This approach allows the model to capture information at various scales within an image, thereby improving memory and computational efficiency. By combining multiple filter sizes such as 1x1, 3x3, and 5x5 simultaneously, this architecture can extract complex features from images at different levels of representation.

Meanwhile, ResNet was developed to address the problem of performance degradation that often occurs when neural networks become deeper. By introducing skip connections or shortcut connections, ResNet enables gradients to flow more easily during backpropagation, effectively mitigating the vanishing gradient problem that frequently arises in deep network training. These shortcut connections allow the model to learn faster and retain important information from earlier layers.

InceptionResNetV1 merges the Inception architecture with the skip connection mechanism from ResNet to enhance training efficiency and stability. Each module in InceptionResNetV1 consists of several convolution branches with different filter sizes, designed to capture multi-scale features. Additionally, the skip connections in this architecture allow the output from one layer to be directly passed to a deeper layer in the network without going through all intermediate layers, minimizing information loss during backpropagation. With this combination, the model maintains the strength of Inception in capturing multi-scale features while ensuring gradient flow stability through the residual learning mechanism of ResNet.

More specifically, the InceptionResNetV1 architecture consists of several main stages: the initial convolution layers, Inception residual blocks (Block35, Block17, Block8), reduction layers (Mixed_6a and Mixed_7a), and the final layers that produce a 512-dimensional facial embedding. The feature extraction process begins with a series of convolution and pooling layers to capture the basic features of facial images. These features are then processed through Inception residual blocks, which include several convolution branches operating in parallel and are combined with skip connections. Next, reduction layers are applied to adjust the feature dimensions before entering the subsequent Inception blocks. In the final stage, the model applies a fully connected layer, batch normalization, and dropout before producing a 512-dimensional facial embedding that can be used for classification or identity verification tasks. Each layer of the InceptionResNetV1 is detailed in Table 1.

Table 1. InceptionResNetV1 Layer Architecture

Layer	**Details**
Conv2D_1a	3x3, 32 filters, stride 2
Conv2D_2a	3x3, 32 filters, stride 1
Conv2D_2b	3x3, 64 filters, stride 1, padding 1
MaxPool_3a	3x3, stride 2
Conv2D_3b	1x1, 80 filters, stride 1
Conv2D_4a	3x3, 192 filters, stride 1
Conv2D_4b	3x3, 256 filters, stride 2
Block35 (x5)	Scale=0.17
Mixed_6a	Reduction block
Block17 (x10)	Scale=0.10
Mixed_7a	Reduction block
Block8 (x5)	Scale=0.20
Final Conv2D	1x1, 1792 filters
AvgPool	8x8
Dropout	p=0.6
Linear	512 units
BatchNorm	512 features

Table 1 presents the details of each layer in the InceptionResNetV1 architecture, consisting of several key stages from the initial convolution to the fully connected layer that produces a 512-dimensional facial embedding. With this architecture, the model is capable of generating richer and more robust facial representations and has been proven effective in various face recognition applications.

2) Pretrained Datasets VGGFace2 and CASIA-WebFace

This study utilizes pretrained weights from VGGFace2 and CASIA-WebFace to apply transfer learning in generating facial embeddings. VGGFace2 contains 3,310,000 images from 9,131 individuals, while CASIA-WebFace includes 494,414 images from 10,575 individuals.

Both datasets are widely used in facial recognition research due to their comprehensive coverage and diversity. This study compares the impact of transfer learning from the two datasets on the resulting model accuracy. The evaluation aims to determine to what extent the choice of pretrained dataset affects the quality of facial embeddings and the resulting authentication performance.

3) Averaging Embeddings

Averaging embeddings is an identity representation approach in face recognition systems that aims to improve the efficiency of the identification or verification process without sacrificing accuracy [35]. This technique involves combining several feature vectors (embeddings) obtained from various facial images of the same individual and calculating the average value for each vector dimension. The result is a single embedding vector that consistently represents the unique characteristics of the individual's face as a whole.

In this study, the averaging embedding method is used to reduce the impact of variation among facial images within the same identity class, such as differences in facial expression, lighting, or viewing angle. By averaging the available embeddings, the system produces a more stable identity vector, enabling more efficient and accurate matching.

d. Face Matching

In facial recognition systems, identity matching is performed by calculating the distance between two embedding vectors using Euclidean Distance [36]. This approach measures the similarity between two facial features extracted using a deep learning model such as InceptionResNetV1.

1) Euclidean Distance

Euclidean Distance is used to measure the distance between two vectors in a high-dimensional space [37]. The smaller the distance, the higher the similarity. Mathematically, the Euclidean Distance between two vectors A and B with n dimensions can be calculated using Equation (2) as follows:

$$d(A,B) = \sqrt{\sum_{i=1}^{n}(A_i - B_i)^2} \qquad (2)$$

where:

A: the facial embedding vector to be identified,

B: the facial embedding vector registered in the database,

n: the number of embedding dimensions.

Based on Equation (2), if the value of d(A, B) is small, the new face is considered highly similar to a face in the database. Conversely, a large distance value indicates that the facial identities are different [38].

2) Face Matching Process

The face matching process is carried out by comparing the embedding of the test facial image with all embeddings stored in the database. Distance calculation is performed using Equation (2) to determine the level of similarity. If the smallest distance found is below a certain threshold, the system considers the face to match one of the existing identities. On the other hand, if all distances exceed the threshold, the system outputs "No match found," indicating no match was detected.

e. Model Evaluation

After the face matching process is completed, the next step is to evaluate the model's performance to ensure its effectiveness in accurately identifying faces. The evaluation is conducted by calculating several metrics that reflect the system's performance in correctly or incorrectly classifying facial images.

The model's performance is measured using four main categories in face matching:

a) **True Positive (TP):** The model correctly identifies a face that is present in the database as a match.

b) **True Negative (TN):** The model correctly identifies that a face not in the database has no match.

c) **False Positive (FP):** The model incorrectly identifies a face that is not in the database as a match.

d) **False Negative (FN):** The model fails to identify a face that is actually in the database.

Based on the classification results, several evaluation metrics are calculated to describe the performance of the face recognition system. These metrics include accuracy Equations (3), precision Equations (4), recall Equations (5), and F1-score Equations (6), each defined in Equations (3) through (6).

a) Akurasi

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

b) Precision

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

c) Recall

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

d) F1-Score

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{6}$$

4 RESULTS AND DISCUSSION

In this study, facial authentication system testing was conducted on 67 student facial data entries registered in the database, as well as several unregistered facial data. The system was tested using two pretrained models—VGGFace2 and CASIA-WebFace—implemented on the InceptionResNetV1 architecture. The tests were conducted using a threshold value of 0.6, where facial matching was based on the Euclidean distance derived from the embeddings.

a. Model Evaluation with Pretrained VGGFace2

The evaluation results of the model using the pretrained VGGFace2 showed optimal performance in recognizing faces registered in the database. Based on the confusion matrix in Figure 5, the model produced 67 True Positives (TP), meaning all 67 faces were correctly recognized as individuals present in the database. There were no misclassifications in the form of False Negatives (FN = 0), indicating that the model did not miss any faces that should have been identified.

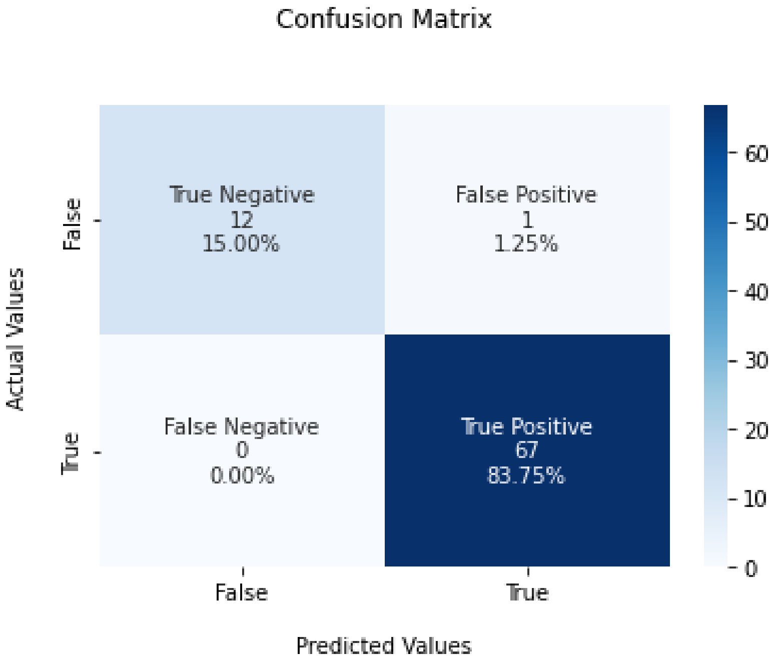


Figure 5. Confusion Matrix Results Using the Pretrained VGGFace2 Model

Furthermore, the model was also able to correctly identify faces that were not present in the database, as indicated by 12 True Negatives (TN). However, there was 1 False Positive (FP), meaning the model incorrectly identified one face—one that should not have matched—as a valid match. With high accuracy and zero False Negatives, the VGGFace2 pretrained model demonstrated strong performance in ensuring that recognized faces truly matched the existing identities. This indicates that the model has high sensitivity and does not miss faces that should be identified.

b. Model Evaluation with Pretrained CASIA-WebFace

Meanwhile, the model trained using the CASIA-WebFace pretrained dataset showed lower performance compared to VGGFace2. Evaluation results indicated that the model correctly identified 57 faces (TP = 57), but there were 10 False Negatives (FN = 10), meaning the model failed to recognize 10 faces that were actually present in the database. According to the confusion matrix shown in Figure 6, this model also demonstrated weaker performance in detecting registered faces.

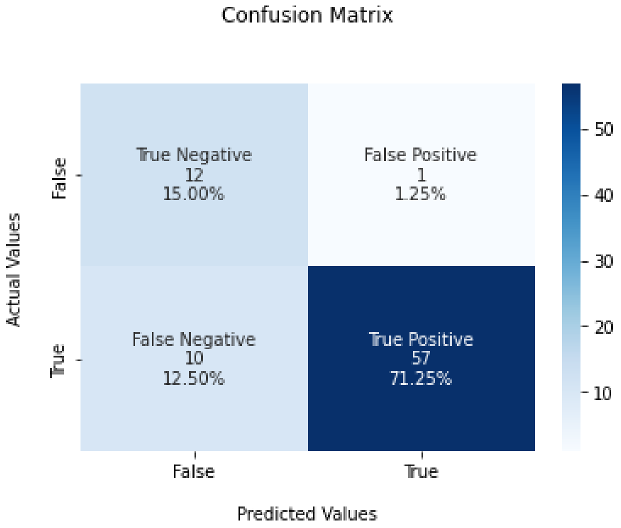


Figure 6. Confusion Matrix Results Using the Pretrained CASIA-WebFace Model

In terms of detecting faces not registered in the database, the model showed the same performance as the pretrained VGGFace2 model, with 12 True Negatives (TN) and 1 False Positive (FP). However, the higher number of False Negatives (FN) indicates that the model using the CASIA-WebFace pretrained weights is less optimal in recognizing faces that should be identified. The main difference between the two models lies in their ability to detect faces registered in the database. The model with VGGFace2 pretrained weights exhibits higher sensitivity compared to CASIA-WebFace, as shown by the lower number of False Negatives.

c. Model Performance Comparison

To gain a clearer picture of the performance of both models, several key evaluation metrics were calculated, including accuracy, precision, recall, specificity, and F1-score. Table 2 presents the model evaluation results based on the calculated metrics:

Table 2. Performance Comparison of Both Models

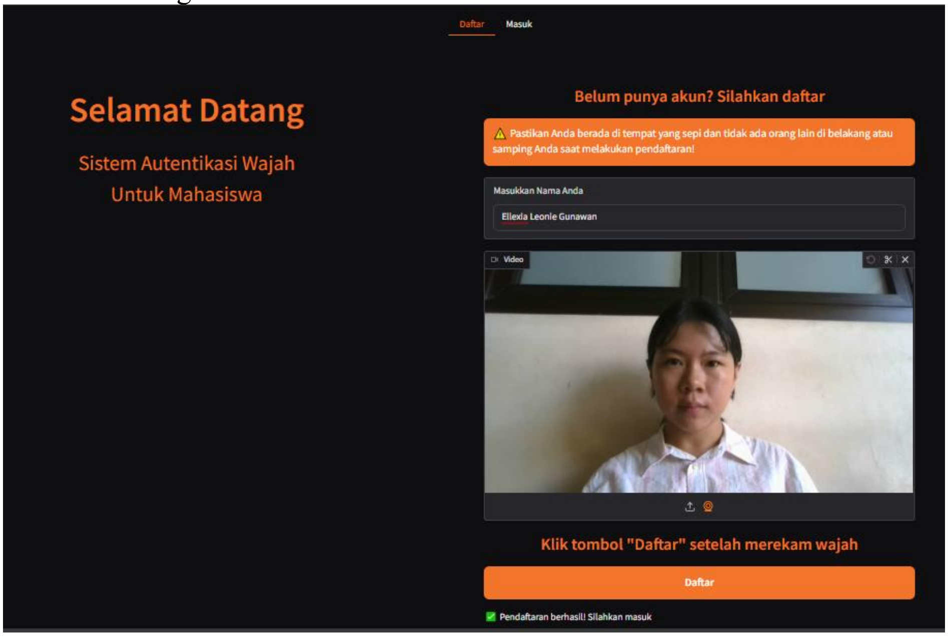
	InceptionResNetV1-VGGFace2	**InceptionResNetV1-CASIA-WebFace**
Akurasi	0.9875	0.8625
Precision	0.9853	0.9828
Recall	1	0.8507
F1 Score	0.9926	0.9119

Based on the results above, the pretrained VGGFace2 model demonstrated superior performance compared to CASIA-WebFace, especially in terms of recall, which reached 100%, indicating that the model did not miss any faces that should have been recognized. On the other hand, the CASIA-WebFace model had a lower recall (85.1%), meaning there were still faces that could not be recognized properly.

In terms of precision and specificity, both models had nearly similar performance, indicating that both were fairly reliable in recognizing faces that were not in the database. However, the significant difference in recall indicates that VGGFace2 is more suitable for applications requiring high sensitivity, such as face-based authentication systems.

d. User Interface of the Developed System

To support ease of use in the facial authentication system, a user interface (GUI) was developed using the Gradio platform. Gradio allows for system implementation and testing locally without the need for additional installation or external servers, and it offers a simple and easy-to-use interface. The GUI consists of two main features: Register and Login, which represent the user registration and login authentication flows.



Selamat Datang

Sistem Autentikasi Wajah
Untuk Mahasiswa

Belum punya akun? Silahkan daftar

⚠ Pastikan Anda berada di tempat yang sepi dan tidak ada orang lain di belakang atau samping Anda saat melakukan pendaftaran!

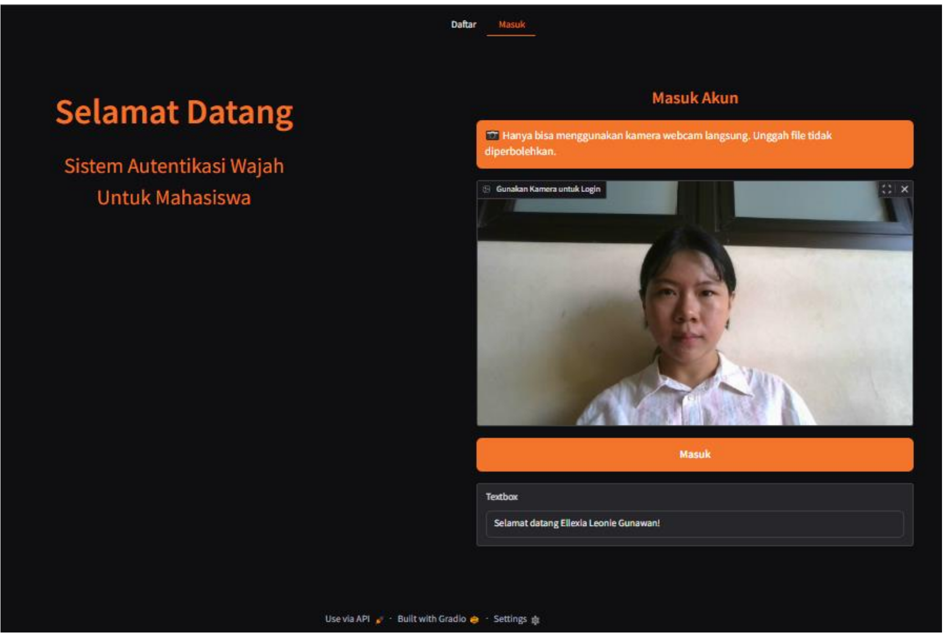
Masukkan Nama Anda: Ellexia Leonie Gunawan

Klik tombol "Daftar" setelah merekam wajah

✓ Pendaftaran berhasil! Silahkan masuk

Figure 7. "Register" Feature Display on the Gradio-Based User Interface

Figure 7 shows the interface of the Register feature. In this feature, users are asked to enter their name and record their face using the camera. The captured facial image is then extracted into an embedding and stored in the local database. Once the registration process is successful, the system provides a text-based notification.


Daftar
Masuk
Gunakan Kamera untuk Login
Masuk
Use via API · Built with Gradio · Settings

Selamat Datang

Sistem Autentikasi Wajah
Untuk Mahasiswa

Masuk Akun

📷 Hanya bisa menggunakan kamera webcam langsung. Unggah file tidak diperbolehkan.

Textbox

Selamat datang Ellexia Leonie Gunawan!

Figure 8. "Login" Feature Display on the Gradio-Based User Interface

Figure 8 shows the interface of the Login feature. Users capture their facial image directly using the camera, after which the system detects the face, extracts its features, and compares it with the stored embeddings in the database using Euclidean Distance calculation. If the face is recognized, the system displays the user's name as a successful authentication result.

With the implementation of this GUI, the entire process from registration to authentication runs in an integrated and intuitive flow. This demonstrates that the system is not only technically reliable in terms of the model used, but also feasible for real-world applications.

5 CONCLUSION

This study aimed to optimize a facial recognition-based authentication system by utilizing the InceptionResNetV1 model retrained using the VGGFace2 dataset. The system was designed to enhance login security by accurately identifying registered users through facial features. Evaluation results showed excellent performance, achieving 67 True Positives (TP), 0 False Negatives (FN), 12 True Negatives (TN), and only 1 False Positive (FP), resulting in an overall accuracy rate of 98.75%. The absence of False Negatives indicates high sensitivity and the system's reliability in recognizing registered users. These findings reinforce that the VGGFace2 pretrained model is capable of generating robust and discriminative facial embeddings, contributing significantly to the optimization of facial recognition processes.

For future development, this system can be further tested in real-time scenarios using live camera input to evaluate its performance under dynamic conditions. Additional tests involving variable conditions such as wearing masks, glasses, different lighting, and facial expressions can be conducted to improve system robustness. Implementation of this system on mobile or web platforms is also an essential step toward practical deployment. Future research may also explore other deep learning architectures or large-scale pretrained datasets to further improve accuracy and generalization across a broader population.

REFERENCES

[1] P. S. B. Bele, G. S. Band, S. S. Kawade, A. D. Udimkar, and R. N. Khandare, "The Role of Biometric and Authentication in Security," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 12,

no. November, pp. 1997–1999, 2024, [Online]. Available: https://www.ijraset.com/best-journal/the-role-of-biometric-and-authentication-in-security Doi: https://doi.org/10.22214/ijraset.2024.65520

[2] Divya Khandekar and Shailesh Bendale, “Biometrics Security System,” *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 3, no. 1, pp. 219–224, Mar. 2023, doi: 10.48175/IJARSCT-8588. Doi: https://doi.org/10.48175/IJARSCT-8588

[3] D. Tribuana, Hazriani, and A. L. Arda, “Face recognition for smart door security access with convolutional neural network method,” *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 22, no. 3, pp. 702–710, 2024, doi: 10.12928/TELKOMNIKA.v22i3.25946. Doi: https://doi.org/10.12928/telkomnika.v22i3.25946

[4] M. A. N. U. Ghani et al., “Enhancing Security and Privacy in Distributed Face Recognition Systems through Blockchain and GAN Technologies,” Comput. Mater. Contin., vol. 79, no. 2, pp. 2609-2623, 2024, doi: https://doi.org/10.32604/cmc.2024.049611 https://doi.org/10.32604/cmc.2024.049611

[5] K. Alhanaee, M. Alhammadi, N. Almenhali, and M. Shatnawi, “Face recognition smart attendance system using deep transfer learning,” *Procedia Comput. Sci.*, vol. 192, pp. 4093–4102, 2021, doi: 10.1016/j.procs.2021.09.184. Doi: https://doi.org/10.1016/j.procs.2021.09.184

[6] M. A. Turk and A. P. Pentland, “Face recognition using eigenfaces,” in *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Comput. Sco. Press, 1991, pp. 586–591. doi: https://doi.org/10.1109/CVPR.1991.139758

[7] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces vs. Fisherfaces: recognition using class specific linear projection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997, doi: https://doi.org/10.1109/34.598228

[8] S. Swapna Rani, “Enhancing Facial Recognition Accuracy in Low-Light Conditions Using Convolutional Neural Networks,” *J. Electr. Syst.*, vol. 20, no. 5s, pp. 2140–2148, 2024, doi: https://doi.org/10.52783/jes.2559

[9] T. Q. Chung, H. C. Huyen, and D. V. Sang, “A Novel Generative Model to Synthesize Face Images for Pose-invariant Face Recognition,” in *2020 International Conference on Multimedia Analysis and Pattern Recognition (MAPR)*, IEEE, Oct. 2020, pp. 1–6. doi: https://doi.org/10.1109/MAPR49794.2020.9237763

[10] W. Setiawan, “Perbandingan Arsitektur Convolutional Neural Network Untuk Klasifikasi Fundus,” *J. Simantec*, vol. 7, no. 2, pp. 48–53, 2020, doi: https://doi.org/10.21107/simantec.v7i2.6551

[11] T. A. Kadhim, N. S. Zghal, W. Hariri, and D. Ben Aissa, “Face Recognition in Multiple Variations Using Deep Learning and Convolutional Neural Networks,” in *2022 IEEE 9th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, IEEE, May 2022, pp. 305–311. doi: https://doi.org/10.1109/SETIT54465.2022.9875530

[12] J. J. Sundi, H. Kumar, Bharti, and R. Bedi, “Real-Time Facial Expression Recognition Using Convolutional Neural Networks for Adaptive User Interfaces,” in *2024 5th International Conference for Emerging Technology (INCET)*, IEEE, May 2024, pp. 1–6. doi: https://doi.org/10.1109/INCET61516.2024.10593062

[13] T. M. Fahrudin and I. Z. A. Illah, “SkinMate: Mobile-Based Application for Detecting Multi-Class Skin Diseases Classification Using Pre-Trained MobileNetV2 on CNN Architecture,” in *2023 IEEE 9th Information Technology International Seminar (ITIS)*, IEEE, Oct. 2023, pp. 1–6. doi: https://doi.org/10.1109/ITIS59651.2023.10420370

[14] R. Liu, J. Jia, Y. Zhou, Y. Zhou, and Y. Liu, “Training Deep Neural Networks with Large-scale Datasets on Sunway High Performance Computer,” in *2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, IEEE, Jun. 2022, pp. 466–471. doi: https://doi.org/10.1109/ICAICA54878.2022.9844581

[15] T. Li, B. He, and Y. Zheng, “Research and Implementation of High Computational Power for Training and Inference of Convolutional Neural Networks,” *Appl. Sci.*, vol. 13, no. 2, 2023, doi: https://doi.org/10.3390/app13021003

[16] S. Ansari, “Computer Vision Modeling on the Cloud,” in *Building Computer Vision Applications Using Artificial Neural Networks*, Berkeley, CA: Apress, 2023, pp. 457–512. doi: https://doi.org/10.1007/978-1-4842-9866-4_10

[17] W. S. J. Saputra, E. Y. Puspaningrum, W. F. Syahputra, A. P. Sari, Y. V. Via, and M. Idhom, “Car Classification Based on Image Using Transfer Learning Convolutional Neural Network,” in *2022 IEEE 8th Information Technology International Seminar (ITIS)*, IEEE, Oct. 2022, pp. 324–327. doi: https://doi.org/10.1109/ITIS57155.2022.10010073

[18] A. H. Ali, M. G. Yaseen, M. Aljanabi, S. A. Abed, and C. GPT, “Transfer Learning: A New Promising Techniques,” *Mesopotamian J. Big Data*, no. February, pp. 29–30, 2023, doi: https://doi.org/10.58496/MJBD/2023/004

[19] L. Yu, N. Jiang, H. Xu, and Z. Qi, “Facial Expression Recognition Based on Improved VGG-face Model and Transfer Learning,” in *Proceedings of the 2023 International Conference on Computer, Vision and Intelligent Technology*, New York, NY, USA: ACM, Aug. 2023, pp. 1–7. doi: https://doi.org/10.1145/3627341.3630376

[20] M. Usgan, R. Ferdiana, and I. Ardiyanto, “Deep learning pre-trained model as feature extraction in facial recognition for identification of electronic identity cards by considering age progressing,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1115, no. 1, p. 012009, Mar. 2021, doi: https://doi.org/10.1088/1757-899X/1115/1/012009

[21] T. H. Tsai and P. T. Chi, “A single-stage face detection and face recognition deep neural network based on feature pyramid and triplet loss,” 2022. doi: https://doi.org/10.1049/ipr2.12479

[22] M. Cardaioli, M. Conti, G. Orazi, P. P. Tricomi, and G. Tsudik, “BLUFADER: Blurred face detection & recognition for privacy-friendly continuous authentication,” Pervasive Mob. Comput., vol. 92, p. 101801, 2023, doi: https://doi.org/10.1016/j.pmcj.2023.101801. https://doi.org/10.1016/j.pmcj.2023.101801

[23] S. Anwarul, T. Choudhury, and S. Dahiya, “A novel hybrid ensemble convolutional neural network for face recognition by optimizing hyperparameters,” *Nonlinear Eng.*, vol. 12, no. 1, Jun. 2023, doi: https://doi.org/10.1515/nleng-2022-0290

[24] P. A. Riyantoko, Sugiarto, and K. M. Hindrayani, “Facial Emotion Detection Using Haar-Cascade Classifier and Convolutional Neural Networks,” *J. Phys. Conf. Ser.*, vol. 1844, no. 1, p. 012004, Mar. 2021, doi: https://doi.org/10.1088/1742-6596/1844/1/012004

[25] I. G. S. M. Diyasa, A. Fauzi, M. Idhom, and A. Setiawan, “Multi-face Recognition for the Detection of Prisoners in Jail using a Modified Cascade Classifier and CNN,” *J. Phys. Conf. Ser.*, vol. 1844, no. 1, p. 012005, Mar. 2021, doi: https://doi.org/10.1088/1742-6596/1844/1/012005

[26] E. Al Daoud and G. Samara, “Improving the Face Recognition Performance Using Gabor and VGGFace2 Features Concatenation,” in *2022 6th International Conference on Information Technology (InCIT)*, IEEE, Nov. 2022, pp. 187–190. doi: https://doi.org/10.1109/InCIT56086.2022.10067669

[27] S. C. Hidayati, J. Tasyanita, C. Fatichah, and Y. Anistyasari, “Understanding Human Face Shape via Inception-ResNet Neural Network Architecture,” in *2023 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)*, IEEE, Nov. 2023, pp. 631–636. doi: https://doi.org/10.1109/ICAMIMIA60881.2023.10427897

[28] O. Elharrouss, N. Almaadeed, and S. Al-Maadeed, “LFR face dataset:Left-Front-Right dataset for pose-invariant face recognition in the wild,” in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, IEEE, Feb. 2020, pp. 124–130. doi: https://doi.org/10.1109/ICIoT48696.2020.9089530

[29] Shahjahan, N. Sheikh, A. Rehman, and M. Anjum, “Development of Face Recognition System Using Support Vector Machine Algorithm,” *Eur. J. Appl. Sci. Eng. Technol.*, vol. 2, no. 3, pp. 214–227, May 2024, doi: https://doi.org/10.59324/ejaset.2024.2(3).20

[30] H. R. Maier *et al.*, “On how data are partitioned in model development and evaluation: Confronting the elephant in the room to enhance model generalization,” *Environ. Model. Softw.*, vol. 167, p. 105779, Sep. 2023, doi: https://doi.org/10.1016/j.envsoft.2023.105779

[31] N. S. Vemulapalli, P. Paladugula, G. S. Prabhat, S. Abhishek, and A. T, “Face Detection

with Landmark using YOLOv8," in *2023 3rd International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET)*, IEEE, Dec. 2023, pp. 1–5. doi: https://doi.org/10.1109/ICEFEET59656.2023.10452204.

[32] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu, and L. Shao, "Normalization Techniques in Training DNNs: Methodology, Analysis and Application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 10173–10196, Aug. 2023, doi: https://doi.org/10.1109/TPAMI.2023.3250241

[33] I. Y. Susrama, I. G. S. M. D., Putra, A. H., Ariefwan, M. R. M., Atnanda, P. A., Trianggraeni, F., & Purbasari, "Feature Extraction for Face Recognition Using Haar Cascade Classifier," 2022. doi: https://doi.org/10.11594/nstp.2022.2432

[34] K. Cao, "Novel coronavirus pneumonia CT image classification based on inception ResNet," *IET Conf. Proc.*, vol. 2024, no. 21, pp. 134–138, Jan. 2025, doi: https://doi.org/10.1049/icp.2024.4214

[35] M. I. Hossain, Sama-E-Shan, and H. Kabir, "An efficient way to recognize faces using mean embeddings," *Proc. 2021 1st Int. Conf. Adv. Electr. Comput. Commun. Sustain. Technol. ICAECT 2021*, no. October, 2021, doi: https://doi.org/10.1109/ICAECT49130.2021.9392401

[36] N. Mardiana, R. D. Dana, Faisal, I. Farida, A. G. Azwar, and Nurwathi, "Similarity Measures Implementation on Face Authentication using Indonesian Citizen ID Card," in *2023 17th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*, IEEE, Oct. 2023, pp. 1–5. doi: https://doi.org/10.1109/TSSA59948.2023.10366880

[37] H. Wu, Y. Cao, H. Wei, and Z. Tian, "Face Recognition Based on Haar like and Euclidean Distance," *J. Phys. Conf. Ser.*, vol. 1813, no. 1, 2021, doi: https://doi.org/10.1088/1742-6596/1813/1/012036

[38] S. M. Sami, J. McCauley, S. Soleymani, N. Nasrabadi, and J. Dawson, "Benchmarking human face similarity using identical twins," *IET Biometrics*, vol. 11, no. 5, pp. 459–484, 2022, doi: https://doi.org/10.1049/bme2.12090