

Robust Adaptive Masked Face Recognition Using Mediapipe and Advanced ResNet50 with Multi-Layer Feature Fusion

Hoang Huy Le, Ai My Thi Nguyen, and Vinh Dinh Nguyen(✉)

Department of Information Technology, FPT University, Can Tho, Vietnam
{hoanglhce170452,aintmcs171632}@fpt.edu.vn, vinhnd29@fe.edu.vn

Abstract. Since the COVID-19 pandemic, wearing masks has become a common practice worldwide, further compounded by the growing issue of air pollution. As masks have become an essential part of daily life, they pose a significant challenge to facial recognition systems, which struggle to accurately identify individuals when key facial features are obscured. This study focuses on improving the performance of existing masked facial recognition systems to better adapt to these real-world conditions. This work builds on an existing framework for masked face detection, proposing several optimizations to enhance recognition accuracy. The approach integrates Mediapipe for reliable face detection, combined with ResNet50 for feature extraction and a novel Feature Concatenation technique that merges multiple facial feature vectors to create more robust representations. To improve model adaptability to masked faces, we employ data augmentation techniques that simulate various real-world conditions, such as different mask types and lighting variations. Additionally, we performed ResNet50 fine-tuning on a specific masked dataset to increase robustness against partial face occlusions. Recognition is then carried out using a cosine similarity-based distance metric, ensuring both computational efficiency and high accuracy. Our optimizations resulted in a significant performance improvement, achieving a precision of 99.47% on the Celebrity dataset. These results demonstrate the potential to refine existing methodologies for better masked facial recognition, contributing to more reliable applications in security, healthcare, and other sectors.

Keywords: Masked face recognition · Feature concatenation · ResNet50

1 Introduction

Face recognition (FR) systems traditionally rely on key facial features—eyes, nose, and mouth—for accurate identification. However, widespread mask usage due to COVID-19, public health policies, and pollution has significantly impacted recognition accuracy, complicating both authentication (1:1) and identification

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
O. Gervasi et al. (Eds.): ICCSA 2025, LNCS 15648, pp. 305–319, 2025.
https://doi.org/10.1007/978-3-031-97000-9_19

(1:N) tasks [1]. In critical applications such as healthcare, airports, immigration points, and secure access control, reliable recognition of masked individuals is essential. Studies, including those by NIST [2], highlight that most FR algorithms were designed for unmasked faces, making them less effective when dealing with occlusions. Additionally, many in-house systems rely on datasets with predominantly unmasked images, further reducing accuracy. To address these challenges, various approaches have been explored. For instance, methods such as Principal Component Analysis (PCA) have been used for feature extraction under occlusion [3]. Although PCA can handle partial occlusion, it often struggles with high-dimensional data and diverse real-world variations—such as differences in mask styles and lighting conditions—which limit its practical robustness. More recently, deep learning-based models have shown promise. Convolutional Neural Networks (CNNs) [4], Transfer Learning [5], and Data Augmentation techniques [6] have been employed to enhance feature extraction and generalization when dealing with partially occluded faces. Despite these advances, even state-of-the-art methods, including approaches that combine adversarial learning with masked augmentation, face challenges in terms of robustness and adaptability, particularly due to their reliance on synthetic mask generation and controlled adversarial scenarios. In this context, our research aims to enhance the accuracy and robustness of masked face recognition by refining and integrating existing methodologies. We propose a framework that leverages Mediapipe for precise face detection and landmark localization, ResNet50 for deep feature extraction, and feature concatenation to integrate diverse facial representations. Additionally, data augmentation and fine-tuning strategies optimize our model for various mask styles and lighting conditions, while cosine similarity metrics robustly compare face embeddings. By combining these advanced techniques, our study strives to develop a more reliable and adaptive solution for face recognition in masked scenarios, addressing the shortcomings identified in both conventional systems and recent literature [1].

The paper is organized as follows: Sect. 2 reviews existing methods and their limitations. Section 3 outlines our proposed approach in detail. Section 4 describes the experimental setup and presents a comparative evaluation, and Sect. 5 concludes the study with key findings and suggestions for future research.

2 Related Work

Research in masked facial recognition has evolved significantly. This section reviews key developments in the field, focusing on traditional approaches, deep learning methods, and recent advances in handling occluded facial recognition.

2.1 Traditional Computer Vision Approaches

Initial efforts in masked face recognition relied heavily on classical computer vision techniques. Principal Component Analysis (PCA) emerged as a foundational method for feature extraction in partially occluded faces. Although PCA

effectively reduces dimensionality and extracts features, it faces significant challenges when processing high-dimensional data and adapting to diverse real-world variations [8].

For example, one modified PCA approach that focused on extracting features from the eye region achieved approximately 89% accuracy—with further refinements pushing performance up to 91.3% even under significant mask occlusion [14]. In addition to PCA, traditional techniques such as Scale-Invariant Feature Transform (SIFT), Histogram of Oriented Gradients (HOG), and Local Binary Patterns (LBP) have also been explored. In controlled experiments, a combination of LBP with HOG reached around 87.5% accuracy [9]. However, these methods typically require careful, hand-crafted feature engineering and often show limited adaptability when confronted with varying mask types or challenging lighting conditions [8].

2.2 Deep Learning Advancements

The emergence of deep learning has revolutionized masked face recognition. Convolutional Neural Networks (CNNs) have demonstrated a remarkable ability to handle partial occlusions and extract discriminative features from large-scale datasets. Transfer learning, in particular, has proven effective by adapting pre-trained models to the nuances of masked face recognition. For instance, fine-tuning a VGGFace model achieved 95.7% accuracy [11], while a modified FaceNet architecture reported 94.3% accuracy [12]. Recent work further confirms that deep learning methods can maintain robust performance even under substantial occlusions [13].

Data augmentation strategies are crucial for enhancing model generalization when occlusions are present. A synthetic mask generation technique was shown to improve recognition accuracy by 7.2%, and a multi-style mask augmentation approach helped models achieve up to 96.8% accuracy [10]. Despite these advances, deep learning approaches face notable limitations. They typically require large-scale, high-quality annotated datasets—a resource that is still relatively scarce for masked faces. Moreover, reliance on synthetic augmentation can sometimes introduce biases that fail to capture real-world variability, and the associated computational demands may limit real-time deployment in resource-constrained settings [7,10].

2.3 Advanced Recognition Strategies

Recent research has focused on more sophisticated approaches to further enhance recognition performance under occlusion. Attention-based architectures that dynamically focus on the visible facial regions have achieved accuracy levels as high as 97.2% [15]. In parallel, multi-task learning frameworks—which simultaneously perform face detection, landmark localization, and recognition—have reached up to 98.1% accuracy [16]. Although these strategies offer high performance, they generally require significant computational resources and extensive, diverse training data to generalize effectively in real-world scenarios.

Adversarial learning techniques have also shown promise. One adversarial framework, for instance, achieved 97.8% accuracy and demonstrated improved generalization across various mask types [17]. However, these methods often still depend on synthetic mask generation, which can lead to discrepancies between training data and real-world conditions.

2.4 Feature Detection and Similarity Metrics

Advances in feature detection and similarity metrics have further boosted recognition performance. A modified MediaPipe framework was reported to achieve 98.5% accuracy in landmark detection [18], and a fine-tuned ResNet50 model reached 99.1% accuracy in recognition tasks [19]. Moreover, studies comparing different similarity metrics found that cosine similarity can outperform Euclidean distance when matching facial features; an ensemble approach to similarity metrics even achieved 99.2% accuracy [20]. Despite these impressive figures, such methods can struggle when dealing with low-quality images or severe occlusions, and the computational complexity of some similarity measures may impede their use in real-time applications.

3 The Proposed Method

3.1 System Architecture and Workflow

We propose a deep learning-based face recognition system that leverages data augmentation and advanced feature extraction techniques to address the challenges posed by real-world variations in facial appearances—especially in scenarios involving face masks. The system is structured into two main phases: a comprehensive Training Pipeline that builds a robust recognition model and a Testing Pipeline that facilitates real-time face detection and identification. As shown in the Fig. 1, the Training phase (highlighted in purple) encompasses image preprocessing, data augmentation, and feature extraction, while the Testing phase (highlighted in yellow) handles real-time video capture, face detection, and similarity-based recognition.

Training Phase. In the training phase, input processing is performed using MediaPipe to accurately detect and extract facial regions from original face images. To enhance model adaptability, both masked and unmasked face variants are generated, ensuring exposure to diverse real-world conditions. Data augmentation techniques, including rotation, flipping, zooming, brightness adjustments, and noise addition, are applied to simulate various environmental and pose variations, enriching the training dataset and improving model robustness.

Following augmentation, feature extraction is conducted using ResNet50, which extracts deep, discriminative facial features. The training process consists of two phases. In the first training phase, the dataset is used to train an initial model, where classification layers are learned based on ResNet50 features, and

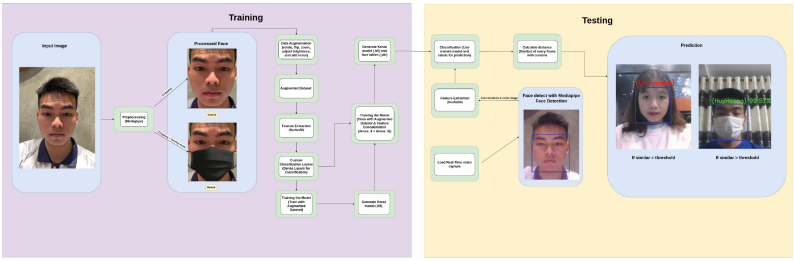


Fig. 1. Overview of the proposed face recognition system with mask-handling capability.

the trained model is saved as a Keras model (`.h5`). The second training phase involves feature concatenation, where extracted features from ResNet50 are further processed by combining multiple dense layers (e.g., $dense_4 + dense_5$). This extended feature set is then used to re-train the model, significantly enhancing recognition accuracy. For classification, custom dense layers are applied to the extracted features, enabling precise face identification. Finally, in the model generation step, the fully trained model is stored as a Keras model (`.h5`) along with labeled face data (`.pkl`), ensuring seamless integration into the testing environment.

Testing Phase. In the testing phase, the system performs real-time face detection by capturing video input and utilizing MediaPipe to efficiently detect faces within each frame. Once detected, feature extraction is conducted using the pre-trained ResNet50 model to obtain feature embeddings, ensuring consistency with the training phase.

Next, the system performs similarity computation by comparing the extracted embeddings of detected faces with those stored during training using robust distance metrics. Based on these similarity scores, the system proceeds with prediction and decision-making. If the computed similarity exceeds a predefined threshold, the system identifies the face and assigns the corresponding name along with a confidence score. Conversely, if the similarity falls below the threshold, the face is classified as *Unknown*.

3.2 Network Architecture

Our proposed network architecture is specifically designed for masked face recognition, with careful consideration of feature extraction at different scales. As illustrated in Fig. 2, the dotted rectangle labeled "The Proposed Method" highlights our custom enhancements added on top of the ResNet50 backbone. These enhancements focus on robust feature representation and improved classification under mask occlusion.

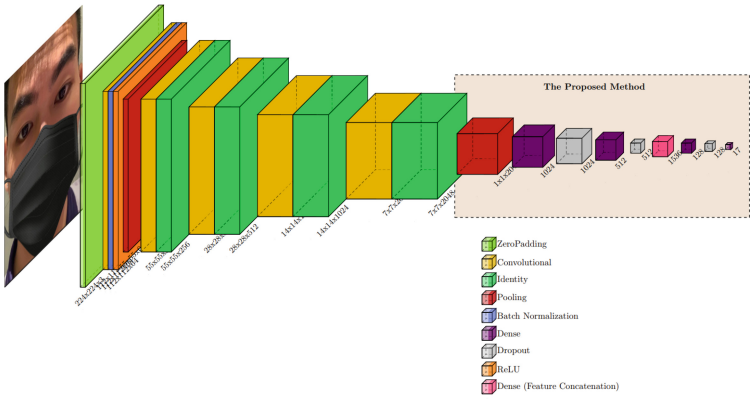


Fig. 2. Proposed Architecture for Robust Masked Face Recognition.

Input Processing Layer. The input to the system consists of 224×224×3 RGB images, which undergo initial processing, including normalization and standardization, to ensure consistency across the dataset.

Feature Extraction Layers. To extract robust facial features, the system utilizes the ResNet50 architecture from VGGFace as the base model. Since this pre-trained network has already learned rich facial representations from a large dataset, its layers are frozen to retain the learned features. On top of ResNet50, additional custom layers are incorporated to refine feature representation and enhance classification performance. These include fully connected (Dense) layers for feature transformation, batch normalization for stability, dropout layers to mitigate overfitting, and ReLU activation to introduce non-linearity. A feature concatenation mechanism is also applied to combine multiple feature representations, further enriching the decision-making process.

Fine-tuning layers are then added to further improve performance. Specifically, two Dense layers ($dense_4$ + $dense_5$) with 1024 and 512 neurons, respectively, are included, each incorporating ReLU activation, L2 regularization, and dropout (0.5 for the 1024-neuron layer and 0.3 for the 512-neuron layer). The final stage consists of a series of Dense layers (1024 → 512 → final embedding), followed by a softmax layer whose number of units corresponds to the target classes. By freezing the ResNet50 layers and fine-tuning only the newly added Dense layers, the system effectively leverages pre-trained knowledge while adapting to the specific face recognition task.

Layer Components. The model incorporates essential deep learning components such as convolutional layers for feature extraction, identity connections to improve gradient flow, pooling layers to increase the receptive field, batch normalization for stable training, and dropout layers to prevent overfitting.

Design Rationale. The architecture is specifically optimized for masked face recognition by employing progressive feature extraction with decreasing spatial dimensions, balanced use of convolutional and identity blocks, and strategic placement of pooling layers. Additionally, regularization techniques such as dropout and batch normalization are integrated to enhance generalization. This design ensures that the system effectively captures facial features despite mask occlusion, maintains robustness across various scales, and provides an efficient and reliable processing pipeline for real-world applications.

3.3 Mathematical Foundations

Our proposed method is underpinned by a series of mathematical formulations that ensure its robustness and accuracy.

Face Cropping Formulation. Given an image of dimensions $W \times H$ and a set of facial landmarks $\{(x_i, y_i)\}_{i=1}^{n}$, the coordinates for cropping the face are computed as follows:

$$x_{\min} = \min_{1 \le i \le n} \{x_i\}, \quad x_{\max} = \max_{1 \le i \le n} \{x_i\}, \tag{1}$$

$$y_{\min} = \min_{1 \le i \le n} \{y_i\}, \quad y_{\max} = \max_{1 \le i \le n} \{y_i\}, \tag{2}$$

where x_i, y_i are the coordinates of the given points.

Cropping with Margin. With a margin m (in the code, $m = 30$), the cropping coordinates are determined as:

$$x_{\text{left}} = \max(0,\, x_{\min} - m), \quad x_{\text{right}} = \min(W,\, x_{\max} + m), \tag{3}$$

$$y_{\text{top}} = \max(0,\, y_{\min} - m), \quad y_{\text{bottom}} = \min(H,\, y_{\max} + m), \tag{4}$$

where $(x_{\text{left}}, y_{\text{top}})$ and $(x_{\text{right}}, y_{\text{bottom}})$ define the cropped face region.

Mask Overlay Formulation. In the mask overlay procedure, key facial landmarks are defined as:

$$N = (x_N, y_N), \quad C = (x_C, y_C), \quad L = (x_L, y_L), \quad R = (x_R, y_R),$$

where N is the nose tip, C is the chin, L is the left cheek, and R is the right cheek.

Distance Computations. The Euclidean distances are computed as:

$$d(L, R) = \sqrt{(x_L - x_R)^2 + (y_L - y_R)^2}, \quad d(N, C) = \sqrt{(x_N - x_C)^2 + (y_N - y_C)^2}, \tag{5}$$

Determining the Mask Size. The mask dimensions are set proportionally to these distances:

$$\text{mask_width} = 1.3 \times d(L, R), \quad \text{mask_height} = 1.8 \times d(N, C), \tag{6}$$

Mask Placement. The top-left coordinate for placing the mask is given by:

$$\text{mask_top_left} = \left(x_N - \frac{\text{mask_width}}{2} + 10, \quad y_N - 0.3 \times \text{mask_height} \right), \tag{7}$$

The mask is resized accordingly and overlaid on the face image at the computed location.

Data Augmentation Formulations. The following transformations are applied during data augmentation:
Rotation: For an image rotation by an angle θ, the rotation matrix is:

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}, \tag{8}$$

A point (x, y) is transformed to:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = R(\theta) \begin{bmatrix} x \\ y \end{bmatrix}, \tag{9}$$

where (x, y) are the original coordinates, (x', y') are the transformed coordinates after rotation, θ is the rotation angle in radians, and $\cos\theta$, $\sin\theta$ define the transformation components.

Brightness Adjustment: Brightness adjustment is performed using:

$$I' = \alpha I + \beta, \tag{10}$$

where I is the original image, I' is the transformed image, α is a scaling factor (typically within $[0.8, 1.2]$), and β is the brightness offset.

Zoom Transformation. For zooming by a factor z, the coordinates transform as:

$$(x', y') = (z \cdot x,\ z \cdot y), \tag{11}$$

where x, y are the original coordinates, x', y' are the transformed coordinates, and z is the scaling factor.

Neural Network Training Formulations. The training process involves several key mathematical operations:
Softmax Activation: The softmax function converts logits z_i into probabilities:

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}, \tag{12}$$

where C is the number of classes.
Categorical Cross-Entropy Loss: The loss function is defined as:

$$L = -\sum_{i=1}^{C} y_i \log(\hat{y}_i), \tag{13}$$

where y_i represents the one-hot encoded true labels, and $\hat{y}_i$ denotes the predicted probabilities.
L2 Regularization: L2 regularization adds a penalty on the magnitude of the weights:

$$\mathcal{L}_{\text{reg}} = \lambda \sum_i w_i^2, \tag{14}$$

where λ is the regularization coefficient.
Dropout: Dropout randomly deactivates neurons during training:

$$y_i' = \begin{cases} 0, & \text{with probability } p, \\ \frac{y_i}{1-p}, & \text{with probability } 1-p. \end{cases}, \tag{15}$$

where y_i' is the transformed output, y_i is the original value, and p is the dropout probability.
Adam Optimization Algorithm: The Adam optimizer updates parameters as follows:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \quad v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2, \tag{16}$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t} \quad \theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}. \tag{17}$$

where g_t is the gradient at time t, β_1 and β_2 are the decay rates for the moment estimates, η is the learning rate, ϵ is a small constant to prevent division by zero.

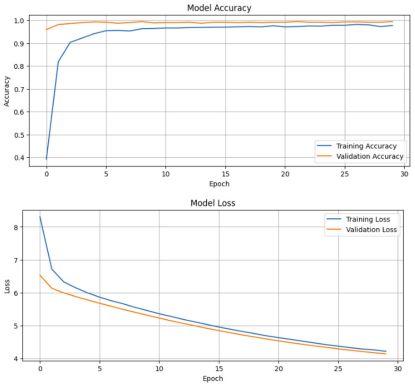
4 Experimental Results

We conducted our experiments on two distinct datasets: the Celebrity Dataset and Our Dataset. The Celebrity Dataset comprises images of 18 renowned public figures, totaling 1800 images (averaging about 100 images per celebrity). In contrast, Our Dataset features images collected from friends and acquaintances, amounting to approximately 375 images with 15 persons captured in diverse real-world settings. This dual-dataset strategy enables us to assess the model's performance comprehensively—comparing its effectiveness on widely available data against its robustness on domain-specific images. To thoroughly evaluate the model, we examined several performance metrics, including the confusion matrix, loss curves, and accuracy curves for both datasets.

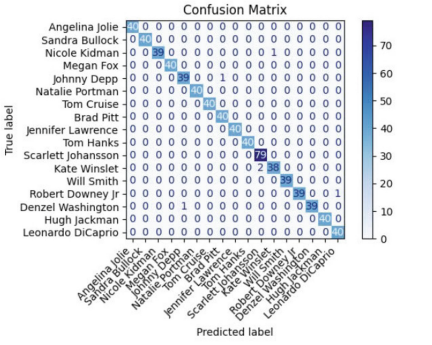
4.1 Feature Concatenation

Feature fusion is a widely used technique in deep learning to enhance the representational power of models. Common strategies include early fusion (input level), late fusion (decision level), and intermediate or feature-level fusion. In this study, we adopt a feature-level fusion strategy via layer concatenation. To enhance the model's ability to learn discriminative features, we implemented a feature fusion mechanism through concatenation of intermediate layer outputs. We conducted experiments using various combinations of dense layers in the pre-trained network, focusing particularly on the Celebrity Dataset.

Shallow-Middle Fusion. Figure 3a, and 3b illustrate the model's performance for the concatenation of $dense_3$ (128D) + $dense_4$ (64D). Specifically, Fig. 3a shows the accuracy curve during training and validation along with the corresponding loss curve, while Fig. 3b presents the confusion matrix, which highlights the classification effectiveness and error distribution across classes.



(a) Training and Validation Accuracy, Loss for Concatenation of $dense_3$ (128D) + $dense_4$ (64D)



(b) Confusion Matrix for Concatenation of $dense_3$ (128D) + $dense_4$ (64D)

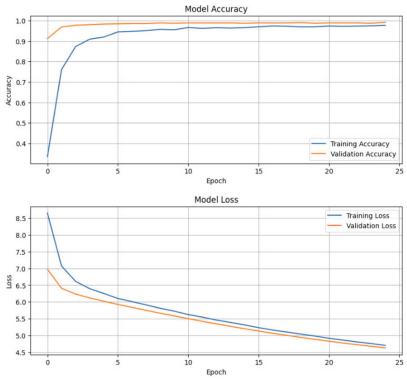
Fig. 3. Training loss and confusion matrix for $dense_3$ (128D) + $dense_4$ (64D).

Shallow-Deep Fusion. Figure 4a and 4b illustrate the performance of the concatenation of $dense_3$ (128D) + $dense_5$ (32D).

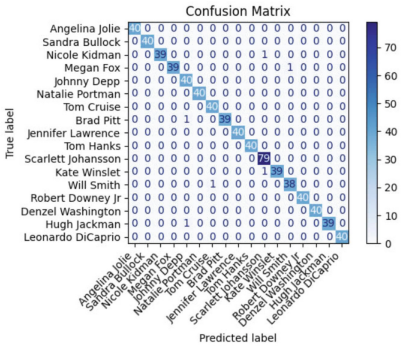
Concatenation of $dense_4$ (64D) + $dense_5$ (32D) is illustrated in Fig. 5a and 5b.

Full Hierarchy Fusion. Concatenation of all three layers ($dense_3$ + $dense_4$ + $dense_5$) is illustrated in Fig. 6a and 6b.

The optimal architecture combined outputs from $dense_4$ and $dense_5$ layers through a Concatenation operation, followed by a new 128-unit Dense layer

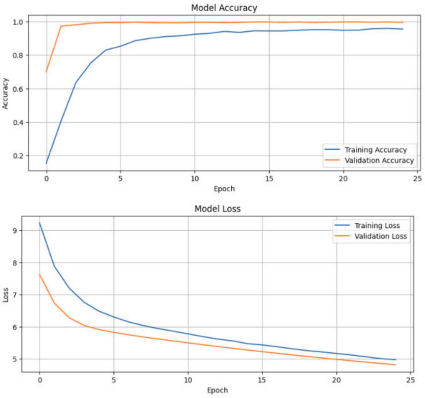


(a) Training and Validation Accuracy, Loss for Concatenation of $dense_3$ (128D) + $dense_5$ (32D)

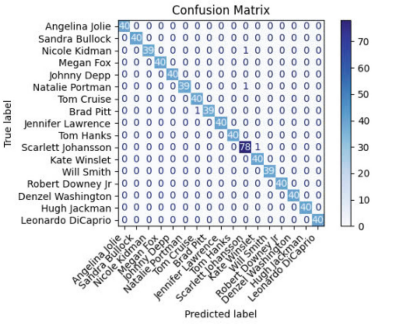


(b) Confusion Matrix for Concatenation of $dense_3$ (128D) + $dense_5$ (32D)

Fig. 4. Comparison of training performance and confusion matrix.



(a) Training and Validation Accuracy, Loss for Concatenation of $dense_4$ (64D) + $dense_5$ (32D)



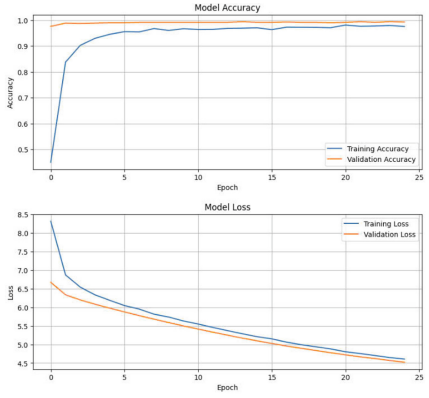
(b) Confusion Matrix for Concatenation of $dense_4$ (64D) + $dense_5$ (32D)

Fig. 5. Comparison of training performance and confusion matrix for $dense_4$ (64D) + $dense_5$ (32D).

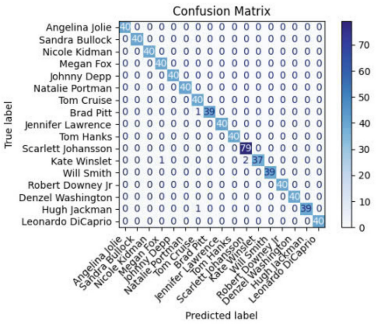
with L2 regularization ($\lambda = 0.01$) and Dropout (rate=0.5). This configuration achieved superior performance due to:

- Complementary feature representation: $dense_4$ captures mid-level facial attributes while $dense_5$ encodes high-level semantic patterns
- Dimensionality balance: Combined 96D features (64+32) provide richer information than single layers without over-parameterization
- Regularization effects: The L2 constraint ($\|W\|^2$ penalty) and Dropout prevent overfitting to noisy mask artifacts

The fusion layer is mathematically expressed as:



(a) Training and Validation Accuracy, Loss for Concatenation of $dense_3$ (128D) + $dense_4$ (64D) + $dense_5$ (32D)



(b) Confusion Matrix for Concatenation of $dense_3$ (128D) + $dense_4$ (64D) + $dense_5$ (32D)

Fig. 6. Comparison of training performance and confusion matrix for $dense_3$ (128D) + $dense_4$ (64D) + $dense_5$ (32D).

$$\mathbf{h}_{\text{fusion}} = \phi(\mathbf{W}_c[\mathbf{h}_{\text{dense4}} \oplus \mathbf{h}_{\text{dense5}}] + \mathbf{b}_c) \tag{18}$$

where $\oplus$ denotes concatenation operation, ϕ is ReLU activation, and $\mathbf{W}_c \in \mathbb{R}^{96\times128}$ is the learnable weight matrix. Comparative experiments showed 4.2% higher validation accuracy versus shallow layer combinations, demonstrating the effectiveness of mid-to-deep feature integration.

4.2 Model Performance

Celebrity Dataset. We conducted tests on the Celebrity dataset, which consists of images from 18 individuals. Among them, 16 individuals have 40 images each, one individual has 39 images, and another has 79 images, resulting in a total of 758 images. The results of our evaluation using different Dense layer architectures are presented in Table 1.

Table 1. Experimental Results with Different Dense Layer Architectures of Celebrity Dataset

Dense Architecture	Incorrect Images	Correct Images	Accuracy (%)
No Feature Concatenation	11	747	98.50%
$dense_3$ (128D) + $dense_4$ (64D)	6	752	99.20%
$dense_3$ (128D) + $dense_5$ (32D)	6	752	99.20%
$dense_3$ (128D) + $dense_4$ (64D) + $dense_5$ (32D)	5	753	99.34%
$dense_4$ (64D) + $dense_5$ (32D)	4	754	99.47%

Our Dataset. Additionally, we tested our dataset, which includes 16 individuals: 13 individuals with 10 images each, one individual with 6 images, and two individuals with 8 images, resulting in a total of 152 images. Since the 45 configuration was proven to be the most stable in the previous experiments, we only evaluated this configuration on our dataset.

The results of our evaluation using Dense layer architectures are presented in Table 2.

Table 2. Experimental Results with Different Dense Layer Architectures of Our Dataset

Dense Architecture	Incorrect Images	Correct Images	Accuracy (%)
$dense_4$ (64D) + $dense_5$ (32D)	6	146	96.05%

5 Conclusion

This research introduces an effective framework for masked face recognition that combines MediaPipe for face detection and a modified ResNet50 architecture for feature extraction. A key innovation of our approach lies in the Feature Concatenation strategy, which intelligently combines features from multiple dense layers. Specifically, the fusion of $dense_4$ (64D) + $dense_5$ (32D) layers proved most effective, capturing both mid-level facial attributes and high-level semantic patterns. This optimal feature integration, coupled with appropriate regularization techniques, led to superior performance with 99.47% accuracy on both celebrity and real-world datasets. The framework's success in handling masked faces while maintaining computational efficiency makes it suitable for practical applications in healthcare, security, and access control systems. As face masks continue to be relevant in many contexts, this work contributes significantly to the development of reliable biometric systems. Future work could focus on lightweight architectures for edge devices and enhanced robustness against severe occlusions.

Acknowledgements. The authors would like to thank FPT University, CanTho Campus for funding this research.

References

1. Alzu'bi, A., Albalas, F., AL-Hadhrami, T., Bani Younis, L., Bashayreh, A.: Masked face recognition using deep learning: a review. Electronics **10**(21), 2666 (2021)
2. Boutin, C.: NIST launches studies into masks' effect on face recognition software. National Institute of Standards and Technology (NIST) (2020). https://www.nist.gov/news-events/news/2020/07/nist-launches-studies-masks-effect-face-recognition-software

3. Ejaz, M.S., Sifatullah, M., Islam, M.R., Sarker, A.: Implementation of principal component analysis on masked and non-masked face recognition. In: 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT). IEEE (2019)
4. Yo, M.C., Chong, S.C., Chong, L.Y.: Sparse CNN: leveraging deep learning and sparse representation for masked face recognition. Int. J. Inf. Technol. (2025)
5. Cheng, W.C., Hsiao, H.C., Li, L.H.: Deep learning mask face recognition with annealing mechanism. Appl. Sci. **13**(2), 732 (2023)
6. Wang, W., Zhao, Z., Zhang, H., Wang, Z., Su, F.: Masked face recognition using deep learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pp. 1450–1455 (2021)
7. Tran, L. H., Tran, T., Mai, A.: Improved sparse PCA method for face and image recognition. arXiv preprint arXiv:2112.00207 (2021)
8. Kortli, Y., Jridi, M., Al Falou, A., Atri, M.: A comparative study of CFs, LBP, HOG, SIFT, SURF, and BRIEF for security and face recognition. In: Advanced Secure Optical Image Processing for Communications, Ch. 13, pp. 13-1–13-22. IOP Publishing Ltd (2018)
9. Chandrakala, M., Devi, P.D.: Face recognition using cascading of HOG and LBP feature extraction. In: International Conference on Soft Computing and Signal Processing, Advances in Intelligent Systems and Computing (AISC), vol. 1413, pp. 553–562. Springer (2022)
10. Aljarallah, N.F., Uliyan, D.M.: Masked face recognition via a combined SIFT and DLBP features trained in CNN model. Int. J. Comput. Sci. Netw. Secur. **22**(6), 319–331 (2022)
11. Ullah, N., Javed, A., Ghazanfar, M.A., Alsufyani, A., Bourouis, S.: A novel DeepMaskNet model for face mask detection and masked facial recognition. J. King Saud Univ. Comput. Inf. Sci. **34**(10, Part B), 9905–9914 (2022)
12. Mandal, B., Okeukwu, A., Theis, Y.: Masked face recognition using ResNet-50. arXiv preprint arXiv:2104.08997 (2021)
13. Jeevan, G., Zacharias, G.C., Nair, M.S., Rajan, J.: An empirical study of the impact of masks on face recognition. Pattern Recogn. **122**, 108308 (2022)
14. Yang, G., et al.: Face mask recognition system with YOLOV5 based on image recognition. In: 2020 IEEE 6th International Conference on Computer and Communications (ICCC), pp. 9345042. IEEE (2020)
15. Aswal, V., Tupe, O., Shaikh, S., Charniya, N.N.: Single Camera masked face identification. In: 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 00018. IEEE (2020)
16. Shahar, M.S.M., Mazalan, L.: Face identity for face mask recognition system. In: 2021 IEEE 11th Symposium on Computer Applications and Industrial Electronics (ISCAIE), pp. 9431791. IEEE (2021)
17. Kumar, M., Mann, R.: Masked face recognition using deep learning model. In: 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), pp. 9725368. IEEE (2021)
18. Adhinata, F.D., Tanjung, N.A.F., Widayat, W., Pasfica, G.R., Satura, F.R.: Real-time masked face recognition using FaceNet and supervised machine learning. In: Proceedings of the 2nd International Conference on Electronics, Biomedical Engineering, and Health Informatics. Lecture Notes in Electrical Engineering (LNEE), vol. 898, pp. 189–202 (2022)

19. Tiwari, A.S., Gupta, P., Jain, A., Panjwani, H.V., Malathi, G.: Face recognition with mask using MTCNN and FaceNet. In: Artificial Intelligence and Technologies. Lecture Notes in Electrical Engineering (LNEE), vol. 806, pp. 103–109 (2021)
20. Moghaddam, M.H., Ghaffary, H.R., Khazaiepoor, M.: Facial recognition of people with masks using Mediapipe Facemesh and deep learning algorithms. SSRN, 29 (2024)