

## ORIGINAL RESEARCH

# A lightweight deep learning model for real-time face recognition

Zong-Yue Deng<sup>1</sup> | Hsin-Han Chiang<sup>2</sup> | Li-Wei Kang<sup>1</sup> | Hsiao-Chi Li<sup>3</sup>

<sup>1</sup>Department of Electrical Engineering, National Taiwan Normal University, Taipei, Taiwan

<sup>2</sup>Department of Vehicle Engineering, National Taipei University of Technology, Taipei, Taiwan

<sup>3</sup>Department of Electrical Engineering, National Taipei University of Technology, Taipei, Taiwan

## Correspondence

Hsiao-Chi Li, National Taipei University of Technology, 1, Sec. 3, Zhongxiao E. Rd., Taipei 10608 Taiwan.

Email: hcli@mail.ntut.edu.tw

## Funding information

National Science and Technology Council,

Grant/Award Numbers:

109-2221-E-027-124-MY3,

111-2221-E-027-147-MY2

## Abstract

Lightweight deep learning models for face recognition are becoming increasingly crucial for deployment on resource-constrained devices such as embedded systems or mobile devices. This paper presents a highly efficient and compact deep learning (DL) model that achieves state-of-the-art performance on various face recognition benchmarks. The developed DL model employs one- or few-shot learning to obtain effective feature embeddings and draws inspiration from FaceNet with significant refinements to achieve a memory size of only 3.5 MB—about 30 times smaller than FaceNet—while maintaining high accuracy and real-time performance. The study demonstrates the model's effectiveness through extensive experiments, which include testing on public datasets and the model's ability to recognize occluded faces in uncontrolled environments using grayscale input images. Compared to the state-of-the-art lightweight models, the proposed model requires fewer FLOPs (0.06G), has a smaller number of parameters (1.2 M), and occupies a smaller model size (3.5 MB) while achieving a competitive level of recognition accuracy and real-time performance. The results show that the model is well-suited for deployment in embedded domains, including live entrance security checks, driver authorization, and in-class attendance systems. *The entire code of FN8 is available on [GitHub](#).*

## 1 | INTRODUCTION

Deep learning (DL) techniques have become a promising method for achieving state-of-the-art face recognition, thanks to their impressive performance on public benchmarks. To extract valuable features from facial databases, evolving architectures with adequate loss functions are crucial for robust face recognition. However, the accuracy of DL algorithms heavily depends on large-scale databases with well-annotated training data, which can make deep face recognition models large, complex, and difficult to deploy on computational resource-limited embedded platforms. To address this challenge, researchers have focused on developing efficient and lightweight face recognition methods, including FaceNet, which has received attention for its high accuracy and robustness to various factors [1].

So far, there have been many research works to advance the specific applications of FaceNet and other state-of-the-art models [2–5], however, they often rely on sophisticated GPU and large databases, which can limit their performance. Additionally, many DL-based algorithms focus on improving

recognition accuracy without considering model size reduction and algorithmic scale. To achieve a balance between state-of-the-art performance and model size, it is essential to develop a lightweight DL model that can achieve high levels of accuracy and compactness.

In the present study, we have addressed the above issues by developing an exceptionally lightweight and efficient DL model for the practical solution of accurate and robust face recognition without compromising performance bottlenecks. The substantive contributions of this work are summarized as follows:

- The study proposes an extremely lightweight DL model for face recognition, with a size of only 3.5 MB, which is 2.3 times lighter than MobileFaceNet. The model's performance is evaluated in terms of accuracy, efficiency, and robustness for occluded faces even under uncontrolled environments.
- The concepts presented in this study are critical for achieving state-of-the-art model performance with much smaller computational resources. These include face embeddings made by modifying the structure of Inception ResNet,

This is an open access article under the terms of the [Creative Commons Attribution](#) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2023 The Authors. *IET Image Processing* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

correspondences enforced through Global Average Pooling (GAP), and network output design.

- The study analyzes and compares the deployment capacity of the proposed lightweight model with the state-of-the-art in terms of model size, number of parameters, Giga floating point operations per second (GFLOPs), and hardware resources.
- In comparison to existing state-of-the-art models, the developed model has a smaller model size and lower computational cost while the recognition performance reaches a competitive accuracy and real-time capability. The proposed model can also be trained on real-world applications for the facial recognition community and can be effectively deployed on resource-constrained devices.

The manuscript is structured as follows. First, Section 2 introduces an overview of related work. Then, Section 3 presents the specific implementation for establishing the proposed lightweight recognition model. Section 4 provides an extensive experimental evaluation and comparison with the selected lightweight face models. Finally, Section 5 gives a final remark as conclusions.

## 2 | RELATED WORKS

Recently, many researchers have demonstrated the reliability of their face recognition models in identifying peoples' identities. These models can be applied to a range of real-world scenarios, such as security checks and monitoring, gated access control, verification of proper face-mask usage, and identification of masked faces [6–10]. While executing these programs on high-performance-computing (HPC) devices poses no issue, they often encounter limited computational resources. This section summarizes existing face recognition solutions and related advancements in lightweight structured frameworks.

### 2.1 | Face recognition

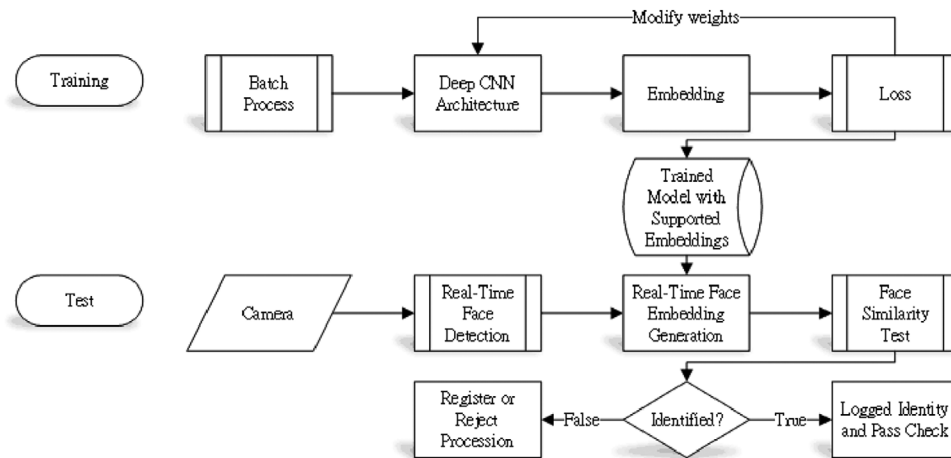
Face recognition aims to identify the identity of an individual. The challenges include occlusion, low-quality images, posture changes, head-angle variations, and illumination conditions. Many research studies focused on and intensively discussed three subdomains over the past decades, including face detection, face feature extraction, and face matching, for constructing such recognition tasks [7, 11–14]. Instead of fulfilling face recognition by these traditional multi-stage frameworks separately, convolutional neural network (CNN) techniques advanced the research field to accomplish the extraction and classification task in a single network without the necessity for predefined facial features. DeepFace [15] with AlexNet backbone, DeepID [16], and VGG-face [17] with VGGNet as its backbone, were proposed to extract facial features using deep convolutional neural networks with fully connected layers performed as a classifier. Despite that, others believe a deeper network can deliver more detailed features, but

it may not be good for discerning subtle inter-class differences and great intra-class variations. Therefore, applying state-of-the-art deep CNNs, such as AlexNet, GoogleNet, ResNet, etc., is a solution for dealing with the different factors of occlusions, poses, or illuminations, along with a conventional classifier, support vector machine (SVM), to achieve higher accuracy [18].

The late structure introduces a few-shot learning strategy that generates feature embeddings via convolutional layers and performs face identification by similarity with the stored faces. FaceNet, proposed by the Google research group, is one of the most representative models in the literature. With its deep network, FaceNet can retrieve facial features as an embedding and perform precise recognition that showed great success and achieved 99% accuracy on the Labeled Faces in the Wild database (LFW) dataset with its embedding generation process learned and supervised by triplet loss. But despite that, FaceNet has a defect in its heavy model size. Hence, many face recognition network variants were modified and inherited from FaceNet, such as OpenFace [19], MobiFace [20], and ShuffleFaceNet [21], to tackle this computational burden. Besides, SphereFace [22], CosFace [23], and ArcFace [24] focused on improving recognition in various poses, illumination, and occlusion by finding suitable margins using angular distance.

### 2.2 | Lightweight deep learning applications

Despite the exceptional accuracy achieved by recognition models that incorporate DL into their frameworks, their computational inefficiency renders them unsuitable for real-world applications. This is due to the fact that embedded devices cannot sustain their DL applications in real time while retaining low latency, low power usage, and high precision under such computational resource constraints. Thus, a practical strategy is necessary as a tradeoff between the model generalization and the model performance. To address this issue, several researchers have proposed lightweight and efficient recognition models. Duong et al. and Martínez-Díaz et al. proposed MobiFace and ShuffleFaceNet, respectively, to maintain identification accuracy with lighter structures to FaceNet. Another lightweight architecture is VarGFaceNet [25], which downscaled VarGNet by alternating the use of blocks with a squeeze and excitation (SE) block. MobileFaceNet [26] comprised MobileNet and ArcFace loss to attain a result for lightweight face recognition. Kocacinar et al. [27] took advantage of MobileNet as its pre-trained model to enhance performance on face-mask identification. ELANet [28], inspired by MobileFaceNet, tackles the challenge of face variations by incorporating spatial attention mechanisms and integrating the multi-scale pyramid module to capture features at different scales. PocketNet, a student model, learns from the ResNet-100 as its teacher model using the differential architecture search (DARTS) algorithm [29]. ConvFaceNet utilizes an enhanced ConvNeXt (ECN) block with depthwise convolution in the first layer [30] to achieve lower FLOPs and reduce the number of parameters. Martínez-Díaz



**FIGURE 1** Overall process of the proposed deep face recognition for training phase and on real-time test phase.

et al. [31] applied and assessed three lightweight face recognition models for the face-mask recognition challenge. Zhao et al. [32] presented a lightweight CNN that incorporates a densely connected convolutional layer with model compression techniques for emotion and expression recognition. Agbo-Ajala and Viriri [33] offered a lightweight model for real-age and apparent age estimation and deployed it on mobile devices. Sun et al. [10] introduced a lightweight solution using CNN for detecting head pose in video rate. Wang et al. [14] degenerated CNN by alternating convolutional layers with Local Binary Pattern (LBP) and Principal Component Analysis (PCA) to eliminate the demand of the long training time in training deep CNNs for multi-face recognition.

### 3 | PROPOSED METHOD

The framework of the proposed method comprises two main components: a series of image preprocessing steps and a recognition model. This research aims to eliminate model redundancy and optimize model performance while maintaining a precise recognition rate. To accomplish this objective, the architecture of the proposed recognition model, based on CNN, the optimization algorithms, image preprocessing steps, and the selection of loss functions, will be introduced. Additionally, this section provides details on the training process for the proposed model. Ultimately, this approach is expected to facilitate the onboard processing of real-time data, as illustrated in Figure 1.

#### 3.1 | Preprocessing

The objective of this study is to reduce the size of the recognition network for face recognition without compromising its accuracy. To achieve this goal, the paper proposes a model that takes grayscale images as its input. To extract the essential features of the target face, several image processing steps are applied, including face registration/detection

**TABLE 1** Proposed FN8 model architecture for face embeddings.

Layer/stride	Kernel	Output size	Parameters
Image	—	160×160×1	—
Conv / 2	3×3	80×80×32	320
Conv / 2	5×5	40×40×64	51.26K
Conv_Sep / 2	3×3	20×20×128	8.896K
Inception ResNet A	—	20×20×128	754.43K
Inception ResNet B	—	20×20×128	344.576K
Conv_Sep / 2	3×3	9×9×196	25.92K
Conv_Sep / 2	3×3	6×6×256	51.136K
GAP	6×6	1×1×125	0
Total	—	—	1.24 M

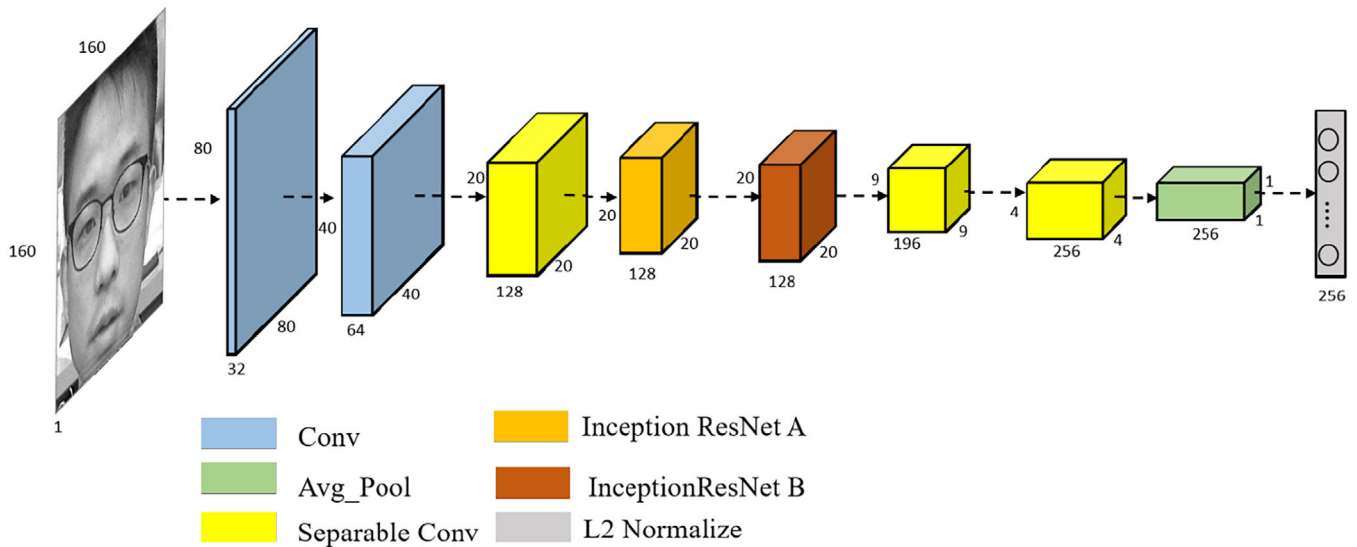
using Haar-like features, histogram equalization for contrast enhancement, enhancement of face contour/edge, and image data augmentation. The augmentation involves processes such as whitening, random rotation, random scaling, and random flipping.

#### 3.2 | Recognition model

This study builds upon the framework of FaceNet and employs a series of simplification strategies to develop a downscaled recognition model called FN8.

##### 3.2.1 | Model structure

Figure 2 depicts the proposed FN8 model for face recognition with its detailed structure presented in Table 1. The model primarily consists of convolutional layers, separable convolutions, and refined Inception ResNets to extract facial features and generate face embeddings for classification.



**FIGURE 2** The proposed lightweight model for face recognition.

To reduce the size of feature maps, a stride of 2 is used as an alternative to pooling. Separable convolutions are employed in the 3<sup>rd</sup>, 6<sup>th</sup>, and 7<sup>th</sup> layers to minimize the number of parameters in the convolution mask. Within the convolutional layers of the refined Inception ResNet-A (Figure 3), filters with heterogeneous sizes ( $3 \times 5$ ,  $5 \times 3$ , and  $1 \times 1$ ) are designed to capture various combinations of facial features. The spatial information of the Inception input is max-pooled to gather the primary facial features. The input of the Inception process is then skip-connected to the resulting Inception features. To avoid over-corrections, the resulting Inception features are rescaled by a factor of 0.1 prior to skip-connection.

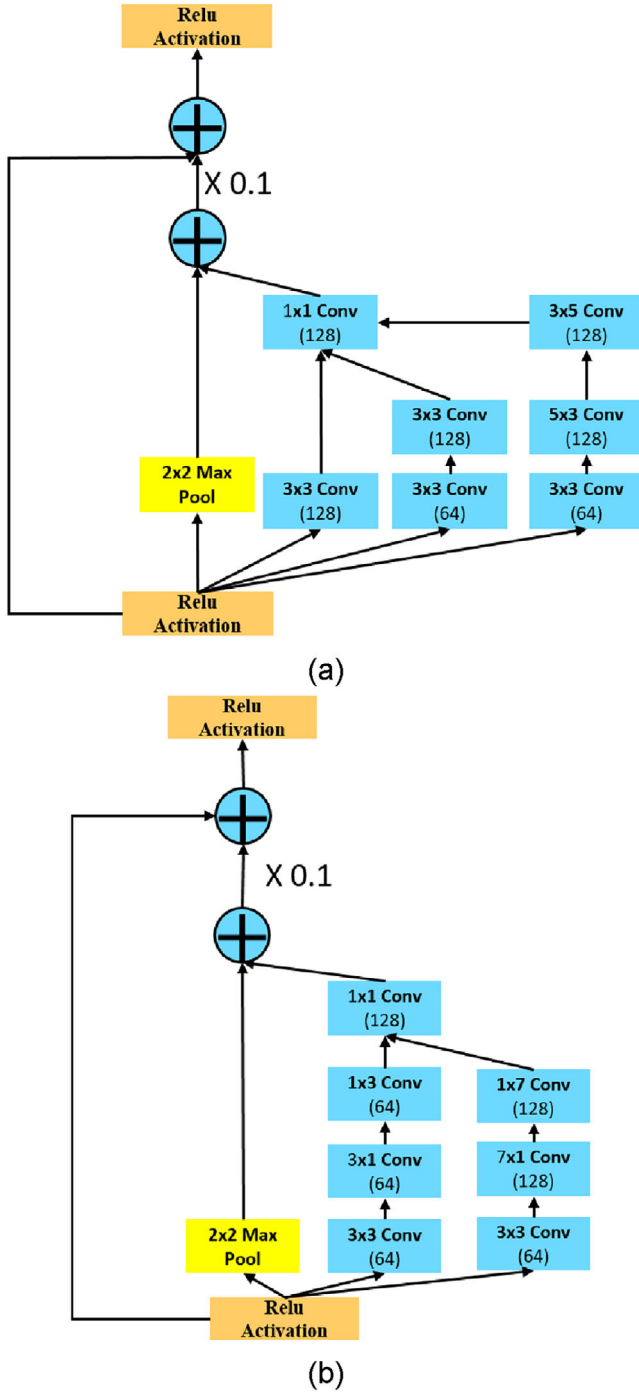
Additionally, the spatial features captured by the modified Inception ResNet-A are passed to the second structure, which is modified from Inception ResNet-B. Max pooling is utilized in these two Inception modules to prevent over-fitting. By incorporating the Inception ResNet-B, FN8 can extract detailed eigen-features that may have been missed. Apart from these modifications, FN8 includes only one GAP layer, which replaces multiple conventional fully connected layers, for extracting 3-dimensional image features as an embedding vector.

### 3.2.2 | Objective function

In recent years, CNN has become the most popular tool for building a discrimination network due to its ability to extract features without manual interventions through the convolution process. While a conventional CNN can easily discriminate between objects in different classes using simple loss functions like softmax loss, face recognition poses a more challenging task. Even though softmax can differentiate multiple faces,

the differences among various classes are often inadequate. This insufficiency may cause image recognition to fail when dealing with difficult tasks. Additionally, the use of the conventional loss setting for CNN, which is composed of the softmax and the cross-entropy, presents several unresolved issues. First, it requires an overloaded computational effort for training a model. Second, it has a limited capability for extracting features during the fully connected layer, especially when it comes to discriminating numerous face classes. Last, the model is not reusable when appending new face classes for the recognition system, which is a crucial requirement for face recognition tasks, especially in real-world applications. Additionally, having an instant response without the need for model retraining is also essential, especially for security access control systems that frequently register new users. Retraining the recognition network for the identity registration of a few new users becomes impractical.

Figure 4a depicts the features for each face class via a model with softmax loss applied. Due to wide variations within a sample class, the resulting model may have a higher misclassification rate, especially when introducing new face classes. Under such circumstances, the model needs an update. Therefore, the proposed model replaces the conservative setting in the loss function by the center loss [34] jointly supervised with softmax to lessen the variety of features within each face class. And simultaneously, class centers are updated based on Fisher's optimization criterion. Under the joint supervision of the softmax loss and the center loss, the same-class samples stay closer to their centers while the various face classes keep apart from the others. Consequently, it not only reduces variations of features within a class but also keeps the differences in between-class features. Figure 4b depicts an example with features in two-dimensional space and illustrates the combination of center loss with softmax loss to train facial feature embedding.



**FIGURE 3** Refined inception structures. (a) Inception ResNet-A. (b) Inception ResNet-B.

To characterize the within-class variation, we start from the center loss function, formulated as

$$L_c = \frac{1}{2} \sum_{i=1}^m \|x_i - c_{y_i}\|_2^2. \quad (1)$$

In the above function,  $m$  is the batch size,  $x_i$  denotes the extracted embedding feature of the  $i^{\text{th}}$  image, and  $c_{y_i}$  repre-

sents the center of class  $y_i$ . During the process of the center update, the model performance can be more efficient while only using the current batches and the corresponding classes. From the gradient of  $L_c$  with respect to  $x_i$  as defined by  $\partial L_c / \partial x_i = x_i - c_{y_i}$ , the update function  $\Delta c_j$  for class  $c_j$  can then be derived as

$$\Delta c_j = \frac{\sum_{i=1}^m \delta(y_i = j) \cdot (c_j - x_i)}{1 + \sum_{i=1}^m \delta(y_i = j)}, \quad (2)$$

where  $\delta(\cdot)$  is the condition function that will be satisfied as  $y_i = j$ . Further, the softmax loss function can be formulated as

$$L_s = - \sum_{i=1}^m \sum_{j=1}^n \delta(y_i = j) \log \frac{e^{\mathbf{W}^T x_i + b}}{\sum_{l=1}^n e^{\mathbf{W}^T x_i + b}}, \quad (3)$$

where  $n$  is the number of individuals stored in the database,  $\mathbf{W}$  is the corresponding weight of the last connected layer of the model, and  $b$  is the bias. Overall, the objective function of collaborating the center loss and the softmax loss of our proposed FN8 model can be formulated as follows:

$$L = L_s + \lambda L_c, \quad (4)$$

where  $\lambda$  denotes the weighting influence of the class-wise center loss.

### 3.2.3 | Batch normalization and optimizer

The proposed FN8 model can reduce a massive number of parameters and calculations. Despite that, there are two common issues affecting training efficiency. First, the model is sensitive to the initials of model parameters and the setting of learning rates. And the second is the impact of calculations in terms of the numeric precision in the DL model. As the designed network gets deeper and deeper, every tiny calculation fluctuation in layers would continue to enlarge. Hence, mini-batch normalization [35] is applied to sort out such issues. It can normalize each layer input with the Internal Covariate Shift via its mean and variance with the mini-batch of size  $m$ , denoted as  $x_i$ , by (5) as

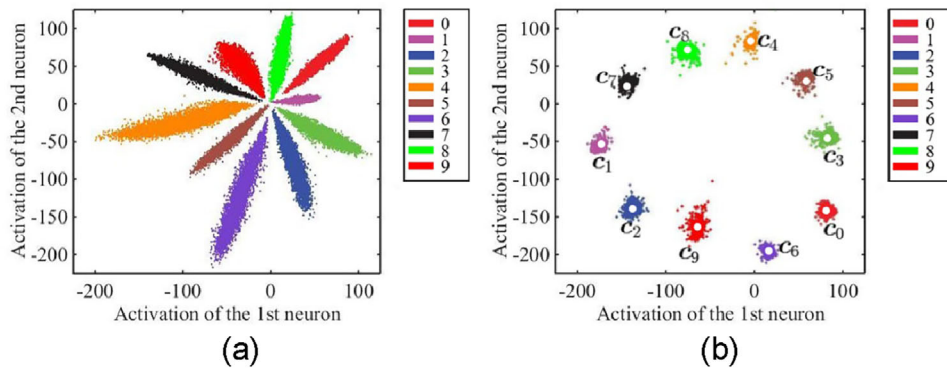
$$\hat{x}_i^{(k)} = \frac{x_i^{(k)} - \mu_B^{(k)}}{\sqrt{\sigma_B^{(k)2} + \epsilon}}, \quad (5)$$

$$\mu_B^{(k)} = \frac{1}{m} \sum_{i=1}^m x_i^{(k)},$$

and

$$\sigma_B^{(k)2} = \frac{1}{m} \sum_{i=1}^m (x_i^{(k)} - \mu_B^{(k)})^2,$$

where  $k$  represents the number of mini-batch,  $\mu_B$  and  $\sigma_B^2$  are the mean and variance of the layer input of each mini-batch,



**FIGURE 4** An illustration of comparing embedding vector learned from the model with/without introducing center loss to concur with softmax jointly; (a) without center loss and (b) with center loss. Points with different colors represent features for various classes.

respectively, and  $\epsilon$  is a constant. To keep its non-linearity characteristics, it can activate the normalized features of a mini-batch by introducing  $\gamma$  and  $\beta$  as parameters to adjust each feature dimension and represent the layer output via the following equation:

$$y_i^{(k)} = \gamma \hat{x}_i^{(k)} + \beta. \quad (6)$$

Some remarkable advantages are worth noting with the introduction of batch normalization in model training.

### 1. Resolve the issue of exploding/vanishing gradients

In the training process, gradients are getting smaller and smaller. At this point, the tuning of parameters becomes more significant to the model. Introducing a mini-batch rescales the parameters and diminishes the reliance on gradients.

### 2. Stable convergence toward the global optimal

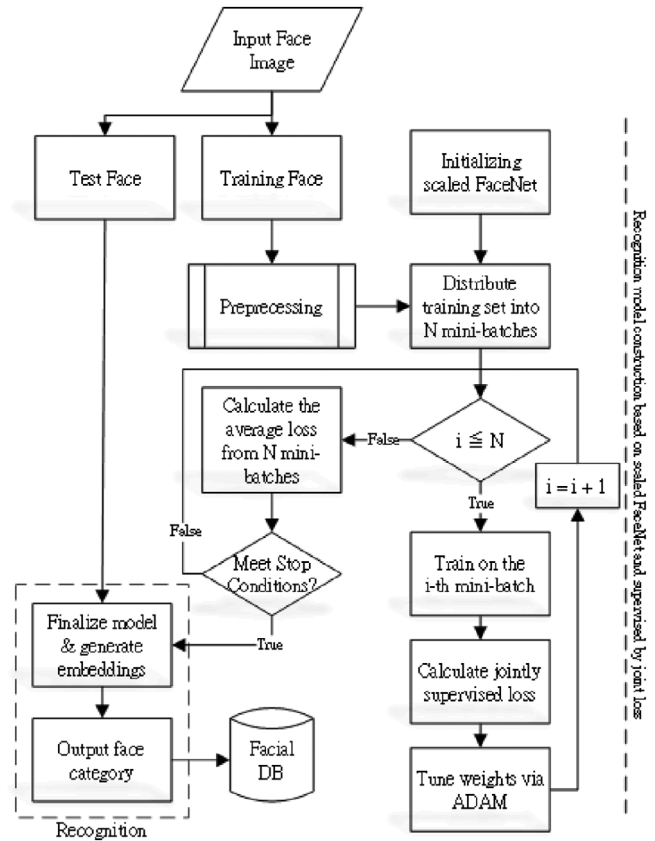
Bringing forward the mini-batch can lead to more stable convergence where the calculated gradients are the average over multi-samples with less noise.

### 3. Avoidance of overfitting

Empirical observations suggest that smaller batch sizes can lead to faster training dynamics. Additionally, smaller batch sizes tend to enhance the model's generalization ability on the test dataset compared to larger batch sizes. This phenomenon can be attributed to the notion that small batches can be seen as noisy representations of the entire dataset. As a result, using small batch sizes introduces a “tug-of-war” dynamic, which prevents the neural network from overfitting on the training set and subsequently avoids poor performance on the test set.

## 3.3 | Model training

The proposed model focuses on grayscale face determination for efficiently identifying faces among multiple face classes. First, a grayscale transformation is applied to generate a single-channel image as the model input. The preprocessed facial images by the forenamed preprocessing steps were then input



**FIGURE 5** The processing flowchart for the proposed face recognition framework.

to the proposed FN8 for model training as the processing flowchart depicted in Figure 5. FN8 has a batch input layer similar to FaceNet but followed by the proposed scaled neural network in a significant lightweight structure. Thus, unlike FaceNet, which has a high demand for computer resources in training, the proposed model can avoid over-fitting under the joint supervision of the center loss and the softmax loss. The proposed model updates its parameters via a backward propagation approach with an ADAM optimizer, and the training process repeatedly performs until the average loss converges

and meets the preset criterion or achieves the number of training epochs. Thus, the trained model can generate a face embedding with the size of 256 entries corresponding to the input facial image while requiring minimal parameter tuning in the training process. After that, the Mahalanobis distance (MD) is adopted for the recognition model to quantify the similarity and sense the difference among various face classes. Through the optimal threshold determined by AUROC for the recognition classifier, it can determine if there exists the input facial image in the database. Once the existence is confirmed, the recognition continues to identify the target's identity that inputs; proceeds to compare the distance of embeddings between the input and stored faces. Ranking by the distance measure as similarity, the model counters a face class with less value as a higher rank. Last, the recognition classifier will assign the face to a class with the highest-ranking number.

The proposed network is designed based on a few-shot learning strategy [36]. Unlike the conventional recognition model, which would generate the possibilities of a subject into every possible class, it embeds all feature information into a vector, referred to as face embedding, instead, and let the discrimination stage work afterward. Here, face embedding plays an essential role in precisely recognizing a subject, and the model particularly needs no retraining for any incoming unseen face.

Since FN8 utilizes multiple loss functions for training different from FaceNet, it is not applicable to measure the similarity of face embeddings as FaceNet does. FaceNet, using triplet loss, can ensure that the distance of feature vectors within a class would be smaller than the distance of feature vectors between various class pairs. Therefore, FaceNet does not need to take care of the correlation among features. Diverse from FaceNet, FN8 devotes itself to separating class samples as far as possible while minimizing the within-class variations. With the choice of MD measure as an alternative to Euclidean distance (ED), FN8 can exclude correlations among features.

### 3.3.1 | Mahalanobis distance

MD is a distance measurement that can effectively quantify the similarity of two samples. Taking the inverse of the sample covariance matrix and data centralization, MD measures the distance of correlated face-embeddings by

$$md(f_{input}, f_c) = \left[ (f_{input} - f_c)^T \Sigma^{-1} (f_{input} - f_c) \right]^{\frac{1}{2}}, \quad (7)$$

where  $f_{input}$  is the face embedding of the input facial image,  $f_c$  is the centroid of all embeddings generated from the training samples, and  $\Sigma$  is the sample covariance matrix of training faces. Due to the characteristics of a symmetric positive semi-definite matrix, the process of eigen-decomposition can decompose the sample covariance matrix as

$$\Sigma = U \Lambda U^T, \quad (8)$$

where the column vectors  $u_i$  of  $U$  are the unitary eigenvectors and  $\Lambda$  is the diagonal matrix consisting of eigenvalues. So that,  $\Sigma^{-1}$  can be further derived by

$$\begin{aligned} \Sigma^{-1} &= (U^T)^{-1} \Lambda^{-1} U^{-1} \\ &= U \Lambda^{-1} U^T, \text{ since } U^{-1} = U^T \\ &= U \Lambda^{-\frac{1}{2}} \Lambda^{-\frac{1}{2}} U^T = \left( U \Lambda^{-\frac{1}{2}} \right) \left( U \Lambda^{-\frac{1}{2}} \right)^T \\ &= \begin{bmatrix} \lambda_1^{-\frac{1}{2}} u_1 & \lambda_2^{-\frac{1}{2}} u_2 & \dots & \lambda_D^{-\frac{1}{2}} u_D \end{bmatrix} \\ &\quad \cdot \begin{bmatrix} \lambda_1^{\frac{1}{2}} u_1 & \dots & \lambda_D^{\frac{1}{2}} u_D \end{bmatrix} \\ &= \sum_{i=1}^D \lambda_i^{-1} u_i u_i^T \end{aligned} \quad (9)$$

In Equation (9),  $D$  represents the feature dimension of the generated face-embeddings. Now, the similarity of two face-embeddings,  $f_1$  and  $f_2$ , can be measured by

$$\begin{aligned} md^2(f_1, f_2) &= (f_1 - f_2)^T \left( \sum_{i=1}^D \lambda_i^{-1} u_i u_i^T \right) (f_1 - f_2) \\ &= \sum_{i=1}^D \lambda_i^{-1} (f_1 - f_2)^T u_i u_i^T (f_1 - f_2) \\ &= \sum_{i=1}^D \lambda_i^{-1} (u_i^T f_1 - u_i^T f_2)^T (u_i^T f_1 - u_i^T f_2) \\ &= \sum_{i=1}^D \lambda_i^{-1} (u_i^T f_1 - u_i^T f_2)^2 \end{aligned} \quad (10)$$

As shown in Equation (10), the correlations among features are reduced by decomposing the sample covariance matrix. Thus, the MD between two face-embeddings can be calculated effectively by taking the unitary eigenvectors.

## 4 | EXPERIMENTAL RESULTS AND EVALUATION

This study primarily sets the experiments on two different computing environmental devices, a general personal computer, and an industrial computer, for implementing the proposed face recognition models. Therefore, the conducted experimental results are ready for verifying their performance. The experiments divide into two parts. The first part focuses on the settings of the model parameters during the training stage, while the second part devotes to the model evaluation in the test phase. For fair comparison, the experiment initiated FN8 with training in a GPU-available device, and further tested on devices without GPU support.

### 4.1 | Dataset

Accurately recognizing human faces in real-world applications can be challenging due to various environmental factors, includ-

**TABLE 2** Model comparison in accuracy of using various dimension number of face embeddings.

# Dimension	Accuracy	F1-score
128	92.82%	0.9405
256	98.40%	0.9465
512	92.37%	0.938

ing lighting conditions, temperature variations, and age-related physical changes. To evaluate the effectiveness of the proposed model in addressing these challenges, it is crucial to use a carefully curated dataset. The Labeled Faces in the Wild database (LFW) [37] is a widely used dataset for face recognition tasks. It comprises a diverse collection of 13,233 facial images, featuring variations in pose, expression, and illumination, sourced from 5,749 individuals.

Additionally, the AgeDB dataset [38] provides age-invariant face verification and includes 16,488 facial images of 568 individuals. This dataset offers age annotations that are free from any noise and comprises 3,000 positive and 3,000 negative face pairs. It encompasses four protocols, each involving different age gaps of 5, 10, 20, and 30 years, allowing for a comprehensive evaluation of age-related face verification techniques. For our evaluation, we selected the AgeDB-30 protocol as it presents the most significant age gap, posing a more challenging scenario compared to the other three protocols.

To demonstrate the discriminative ability of the proposed FN8 model, a subset of the LFW test set was used to perform a similarity test. The similarity scores range from  $-1$  to  $1$ , where a value close to  $1$  represents the two samples that are more likely to belong to the same identity. Scores close to  $-1$  may occur when the two samples belong to the same identity but with opposite postures. As shown in Figure 6, most of the samples share limited relations, with the value approaching zero. The highest similarity value among two different identities (in non-diagonal entries) is about  $0.26$  (Abdel versus Abbas). This is due to the two identities sharing a similar signature, sunglasses-wearing, in their sample figures. The recognition model must rely on the other extracted features forming a discerned face embedding to perform better discrimination on disguised samples. Additionally, Figure 6a demonstrates that FN8 has a good ability in separating various faces since most of their similarity scores are close to zero.

To generate a face embedding for each facial image, the proposed FN8 model was used to produce a 256-dimensional embedding. The choice of 256 dimensions was made based on empirical analysis, as discussed in subsection C. From Table 2, it can be seen that the FN8 model with 256-dimensional face embeddings achieves higher accuracy than those with 128-dimensional or 512-dimensional face embedding outputs.

## 4.2 | Results and evaluation

In this section, we provide the experimental results and performance evaluation of the proposed FN8 model. Initially,

FN8 was trained using the CASIA-WebFace [39], and its performance was assessed using the LFW dataset and AgeDB dataset. Subsequently, FN8's performance was evaluated in a real-time scenario using our self-collected dataset. To further demonstrate the superior computing efficiency of FN8, two state-of-the-art lightweight models were used as baselines to test the self-collected data.

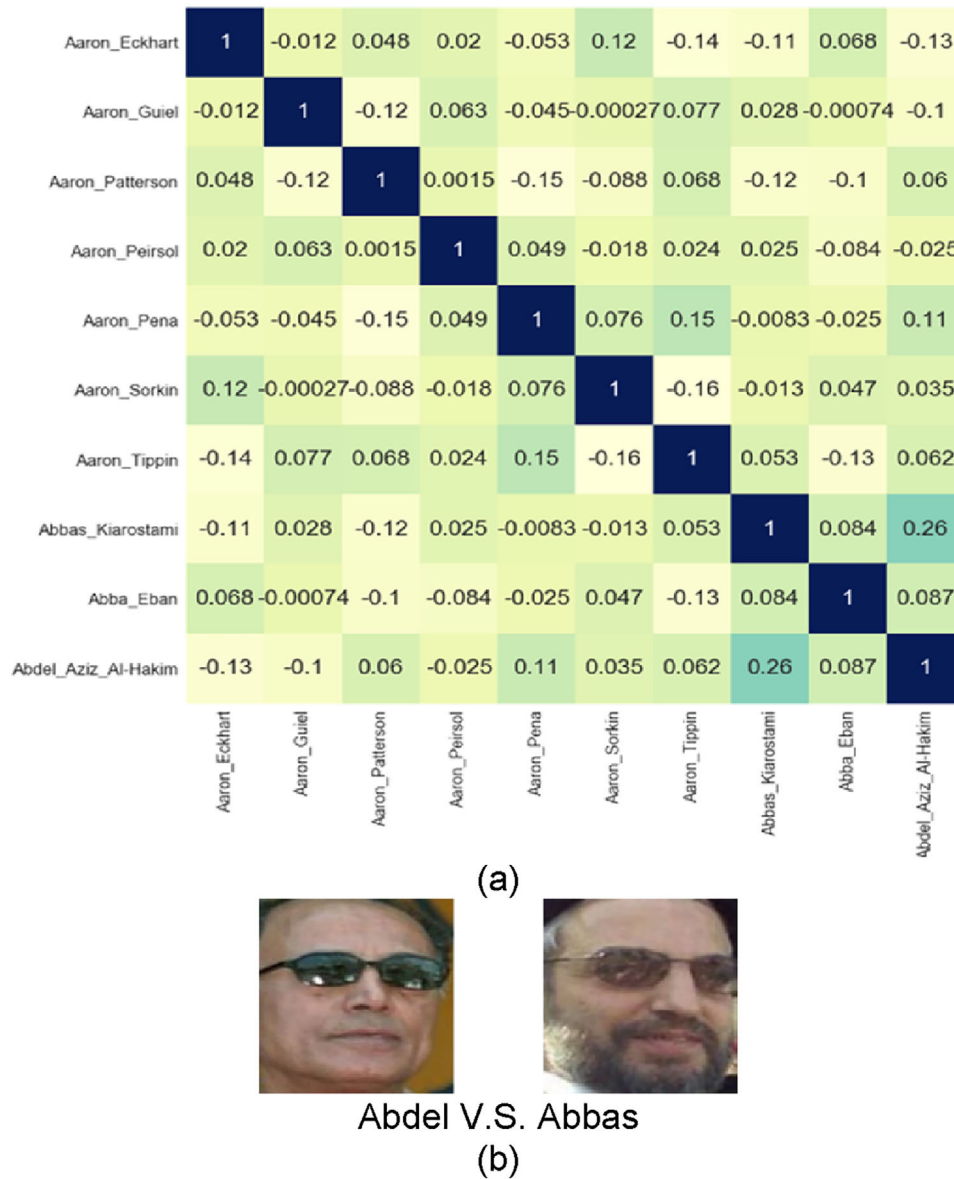
### 4.2.1 | Performance comparison on public datasets

FN8 was trained on the CASIA-WebFace dataset, utilizing 128 batches for batch normalization. The model achieved an impressive 99% accuracy for LFW verification, showcasing its effectiveness despite its lightweight structure. Figures 7 and 8 illustrate the accuracy and the trend of loss of, the model, respectively, throughout the training process with the center loss being jointly induced by the softmax loss. The use of the ADAM optimizer contributed to the improvement in both accuracy and loss. Notably, a significant drop in the loss trend is observed in Figures 8 and 9 when the learning rate was set to  $10^{-3}$  at the 100,000<sup>th</sup> iteration.

In a performance comparison using the LFW dataset, FN8 outperformed FaceNet in terms of precision and model efficiency on both computationally resourceful and relatively low computational resource devices. As tabulated in Table 3, FN8 maintained high accuracy while significantly reducing the model size to about 30 times lighter than FaceNet. Additionally, FN8 demonstrates high accuracy in identifying individuals without sacrificing reliability, as measured by the F1-score. By employing a low-shot learning strategy [40] and utilizing grayscale images, FN8 achieved higher accuracy by incorporating additional stored face samples, without significantly increasing the processing time. The results demonstrate that FN8 exhibited superior real-time performance in terms of frame-per-second (fps) on both the computationally resourceful PC (i7-2600 processor) and the relatively low computational resource device (N3710 processor). Notably, with the computationally resourceful PC, FN8 improved the recognition accuracy from 98.4% (1-sample) to 99.2% (3-sample) with a negligible increase of only 0.002 seconds in processing time.

Furthermore, the comparison results presented in Table 4 demonstrate that the developed lightweight DL model surpasses selected state-of-the-art lightweight models as well as recently proposed lightweight variations in terms of storage space (model size in MB), compactness (number of parameters), and computing efficiency (GFlops). Among the three new models, ELANet achieves the highest accuracy in LFW verification at 99.68%; nevertheless, it has the largest storage space ( $>1$  M).

ConvFaceNext, which was trained with a similar-sized dataset to FN8, exhibits a 0.9% higher accuracy than FN8 on the LFW dataset but a 3.4% lower accuracy on the AgeDB-30 dataset. On the other hand, PocketNetS-256, trained with the



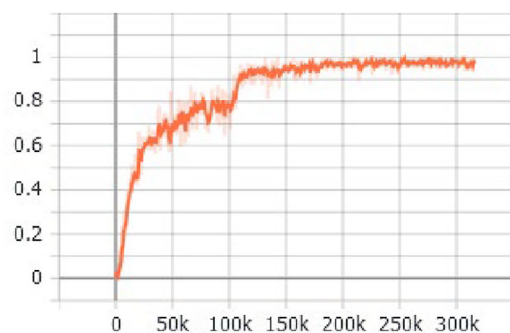
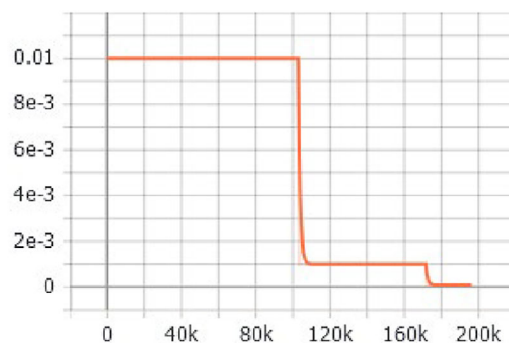
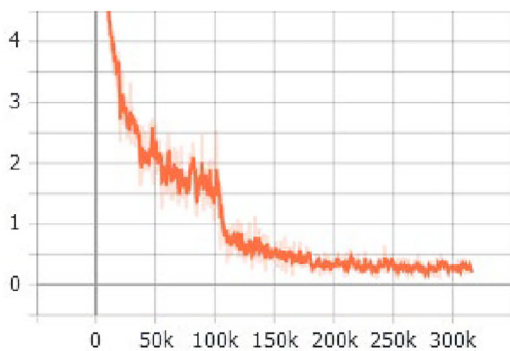
**FIGURE 6** (a) The similarity test for LFW dataset using FN8 generated face embeddings; (b) the two identities with face figures sharing the highest correlation in (a).

**TABLE 3** Model performance comparison between FN8 and FaceNet on LFW.

	Training dataset, #Face Images	Loss function	Model size (MB)	#Param. (M)	GFlops	Accuracy	F1- score	Inference time (s) (CPU i7-2600)	Inference time (s) (CPU N3710)	fps
FaceNet	VGGFace2, 100–200 M	Triplet	95	7.5	1.6	99.6%	—	0.082	0.840	1
FN8 (1-sample)	CASIA, 0.49 M	Center + Softmax	3.5	1.2	0.06	98.4%	0.9405	0.028	0.220	4
FN8 (3-sample)						99.2%	0.9756	0.030	0.232	4

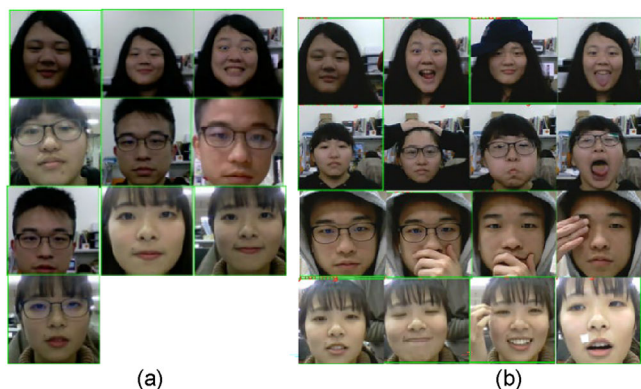
**TABLE 4** Model performance comparison among FN8, state-of-the-art lightweight models, and other lightweight variants.

	Training dataset, #Face Images	Loss function	Model size (MB)	#Param. (M)	GFlops	LFW Acc.	AgeDB-30 Acc.	Inference time (s) (CPU i7-11800H)	fps
MobileFaceNet [26]	CASIA, 0.49 M	ArcFace	4	0.99	0.44	99.28%	93.05%	0.058	17
MobileFaceNets [26]	Cleaned MS-Celeb-1 M, 3.8 M	ArcFace	4	0.99	0.44	99.55%	96.07%	—	—
MobileFaceNet [5]	MS1M- RetinaFace, 5.1 M	ArcFace	8.2	2.0	0.93	99.7%	97.6%	—	—
ShuffleFaceNet 1.5x [21]	MS1M- RetinaFace, 5.1 M	ArcFace	10.5	2.6	0.58	99.67%	97.3%	0.043	23
VarGFaceNet [27]	MS1M- RetinaFace, 5.1 M	Angular Distillation	20.0	5.0	1.02	99.68%	97.5%	—	—
ELANet [28]	MS1M- RetinaFace, 5.1 M	ArcFace	—	1.6	0.55	99.68%	—	0.073	13
PocketNetS-256 [29]	MS1M-ArcFace, 5.8 M	ArcFace	3.9	0.99	0.59	99.66%	96.35%	0.081	12
ConvFaceNeXt_PS [30]	UMD, 0.37 M	ArcFace	4.11	0.96	0.39	99.3%	92.88%	0.035	28
FN8 (1-sample)	CASIA, 0.49 M	Center + Softmax	<b>3.5</b>	1.2	<b>0.06</b>	98.4%	96.26%	<b>0.033</b>	<b>30</b>
FN8 (3-sample)						99.2%	—	0.035	28

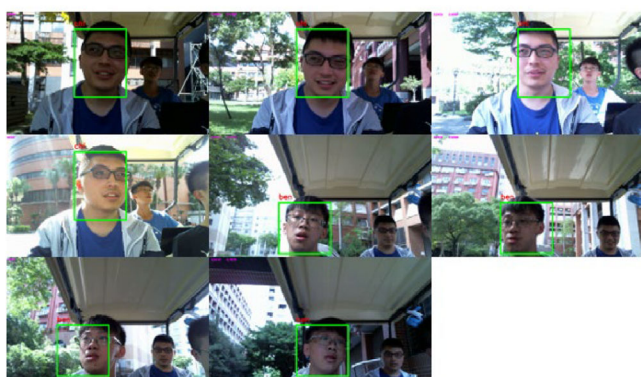
**FIGURE 7** Model accuracy in training.**FIGURE 9** The trend of learning rate.**FIGURE 8** The trend of training loss.

larger MS1M-ArcFace dataset (5.8 M), demonstrates the best performance on both datasets but requires more than double ( $>2\times$ ) the computing time of FN8.

Despite being trained on a relatively small dataset (CASIA-WebFace, 0.49 M), FN8 achieves a competitive accuracy on the LFW dataset compared to the other models and even surpasses MobileFaceNet on the AgeDB-30 dataset. This suggests that the proposed model structure of FN8, jointly supervised by center loss and softmax loss, provides sufficient information for better convergence in training without sacrificing generality. This is particularly evident when evaluating model efficiency on the AgeDB-30 dataset. As AgeDB only consists of one



**FIGURE 10** Real-time recognition for disguised faces; (a) the registered faces, (b) successfully recognized faces.



**FIGURE 11** Recognition test under the uncontrolled outdoor environment with the authorized access to autonomous vehicles.

positive sample and one negative sample, the experimental results only report FN8 using a one-shot strategy. Remarkably, FN8 outperforms MobileFaceNet trained on CASIA-WebFace in this scenario. Under identical testing conditions and environmental settings, the shorter training time and faster processing speed underscore the effectiveness of our methods used in the development of a lightweight DL model.

#### 4.2.2 | Performance comparison in real-time scenarios

An anti-environmental test examines FN8's robustness with disguised-conditioned face samples. Figure 10 illustrates the recognition result of registered faces under disguised conditions in real-time processing. This newly collected dataset consists of 10 subjects with several still images and video sequences. The model with few-shot learning techniques can correctly recognize their identities in a controlled indoor environment. Not only did FN8 exhibit high accuracy, but it also performed fast and robust recognition against varying sizes of face images, facial expression variations, and partial face occlusions. Figure 11 demonstrates a real-world in-vehicle application with

two subjects riding on a golf cart for driver identification and authorization through FN8. In addition to containing complicated backgrounds, the challenging problems in this outdoor environment include the significant changes in illumination, head pose, and camera shake during vibrations caused by cart moving. With only grayscale facial images, our FN8 model successfully overcomes interference in the uncontrolled outdoor surroundings and can recognize the registered faces by executing the similarity test for driver identification and monitoring system.

To demonstrate the superiority of FN8 over other state-of-the-art lightweight face recognition models, this study evaluated its performance on a small self-collected dataset consisting of 12 subjects and 240 images for model training and testing. We selected ShuffleFaceNet and MobileFaceNet as baseline models and ensured that all models were trained and validated under identical conditions on a single computer with the identical hardware (intel i7 processor, 2.3 GHz, 16 GB RAM) and software version without GPU support. The same initial learning rate, fixed number of epochs, as well as maximum iterations, was set for all models. The training set comprises 200 images, and the remaining 40 images were used for testing. As shown in Tables 5 and 6, FN8 took about the same time to complete the training process as the other models. Notably, despite its reduced model size and the number of model parameters, FN8 achieved a recognition rate of 30 fps for a single identity without sacrificing validation accuracy. This makes it an ideal candidate for rapid deployment in real-time face recognition applications, such as live entrance security checks, driver authorization, and in-class attendance systems. Overall, the experimental results demonstrate that the developed lightweight DL model outperforms selected state-of-the-art lightweight models in terms of storage space, compactness, and computing efficiency which makes it more practical for deployment in embedded applications.

### 4.3 | Discussion

#### 4.3.1 | Number of embedding dimensions

This research work investigates the proper size of the embeddings via PCA. By applying PCA to the LFW training set, some featured eigenfaces can be predetermined and ranked according to the degree of variations in each dimension. As the various sizes of embedding are chosen for FN8 to generate, the distribution of embedding vectors can be analyzed and compared. Figure 12 shows that the recognition model messed up easily in distinguishing different class samples with the embedding vectors of size 128. While with a larger embedding size, it is easier to categorize the among-class differences. However, the oversize issue of face embeddings would result in an irregular representation, as shown in Figure 12c. As it concludes, FN8 with 256-dimensional face embeddings can achieve better recognition among face classes, so it sets the size of 256 dimensions for generating embeddings in this work.

**TABLE 5** Comparisons in training on single CPU with state-of-the-art lightweight models.

	Image size	Learning rate	Epochs	Elapsed time	iterations	Validation accuracy
ShuffleFaceNet	112×112	0.001	30	11'37"	100	100%
MobileFaceNet	112×112	0.001	30	14'25"	100	100%
FN8	160×160	0.001	30	13'19"	100	100%

**TABLE 6** Test result comparison with state-of-the-art lightweight models.

	Accuracy	Inference time (s)	fps
ShuffleFaceNet	100%	0.043"	23
MobileFaceNet	100%	0.058"	17
FN8	100%	<b>0.033"</b>	<b>30</b>

### 4.3.2 | Performance bottleneck analysis

This subsection conducts the performance bottleneck analysis on the proposed model for face recognition, and the result shows in Figure 13. The horizontal axis represents the learned weights in each layer node, and the vertical axis shows the iteration number in the model learning stage. A healthy learning model is characterized by the presence of spreading values in the graph during each iteration. Upon analyzing the weights in FN8, it is evident that they display a centralized distribution throughout the initial 100,000 iterations. As the number of iterations increases, the distribution of model weights gets more evenly. Hence, the proposed FN8 can be considered a healthy network and can be used to validate the efficiency of model parameters.

### 4.3.3 | Critical cases

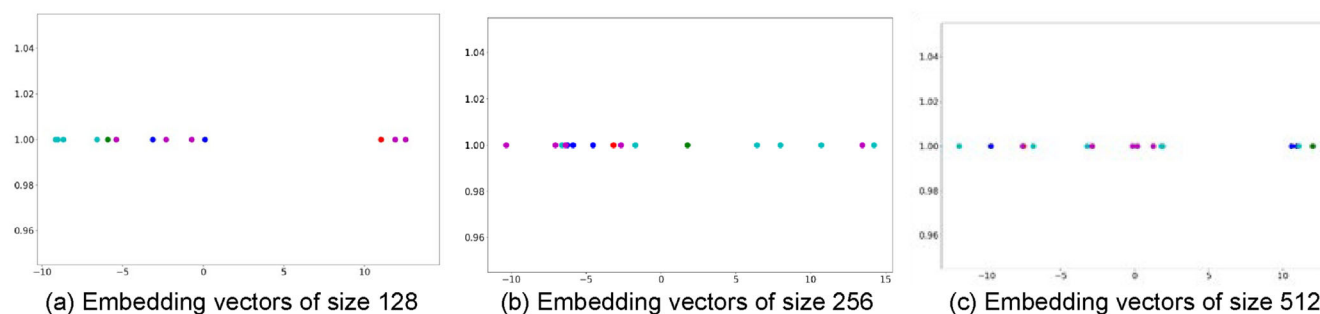
This study also investigates a critical challenge for the discrimination task of twins and conducts the test result with the collected twin figures. Figures 14 and 15 show the two pairs of twins used for the twin discrimination challenge. Figure 16 demonstrates the similarity map for the twin discrimination via FaceNet and FN8, respectively. As can be found from the

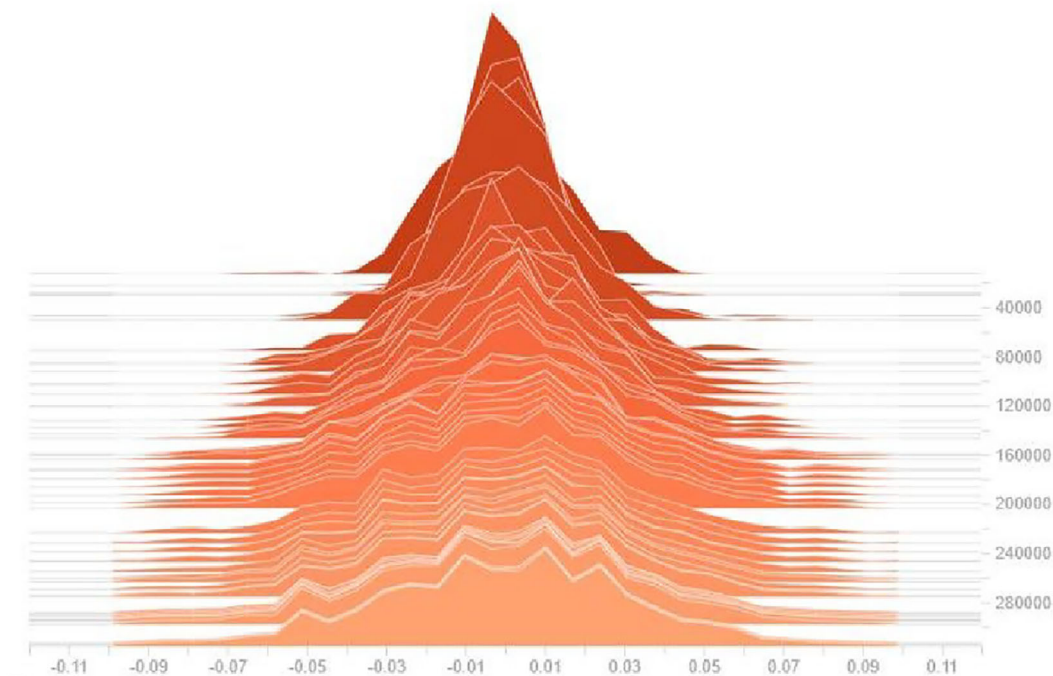
results, the differences between the two twins are not enough while the two subjects stay in the same position; they have even higher differences than the same subject in different directions. Digging into the shown values, FN8 with MD for distance measurements results in a better resolution (range of values) in discrimination. The results reveal its potential ability to distinguish critical cases or even non-identical twins by increasing the number of sample cases. Although FN8 cannot differentiate the twins, its satisfying performance in routine situations still exhibits a high discern rate.

### 4.3.4 | Ablation study—Effect of modified model structures

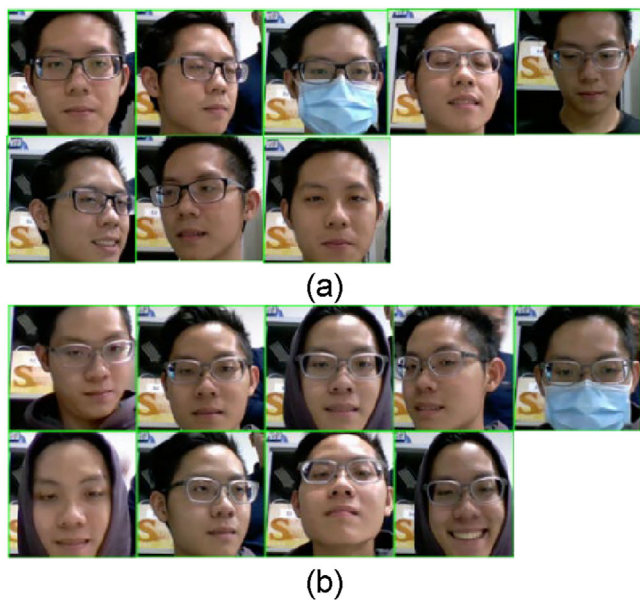
This section aims to investigate the impact of using the refined Inception ResNets in the proposed framework. As a baseline, the model retains all Inception ResNets (A, B, and C) used in FaceNet, achieving an equivalent high accuracy of 99%. To assess the significance of each net, we conducted ablation experiments by removing one layer at a time. The results showed that without Inception ResNet A, the accuracy dropped to 86.67%; without Inception ResNet B, the accuracy dropped to 89.25%; and without Inception ResNet C, the accuracy dropped to 92%. These findings suggest that removing Inception ResNet C significantly reduces the model size and number of parameters with only a slightly sacrificing accuracy, offering a trade-off.

To further enhance the model's recognition rate, modifications were made to Inception ResNet A and B to compensate for the precision loss caused by the reduction process. By introducing the aforementioned simplification strategies to modifying Inception ResNets, the model's accuracy was increased to 97%. Additionally, by incorporating the refined Inception

**FIGURE 12** PCA analysis on the number of embedding dimensions. The dot points with various colors represent the different face classes. (a) Embedding vectors of size 128. (b) Embedding vectors of size 256. (c) Embedding vectors of size 512.



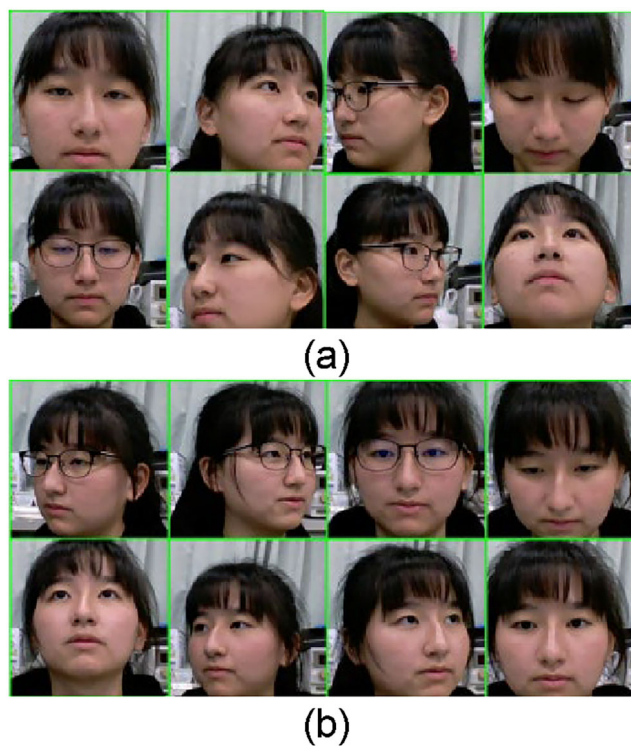
**FIGURE 13** Performance bottleneck analysis in layer weights.



**FIGURE 14** Subjects of male twins; (a) Subject 1 and (b) Subject 2.

ResNets with a  $2 \times 2$  Max Pooling, the model achieved an accuracy of 98.4%. The simplified and refined model structures of FN8 enable support for few-shot learning while maintaining real-time performance demand. In the case of utilizing 3 grayscale shots, FN8 can achieve a high-level accuracy of 99.2%.

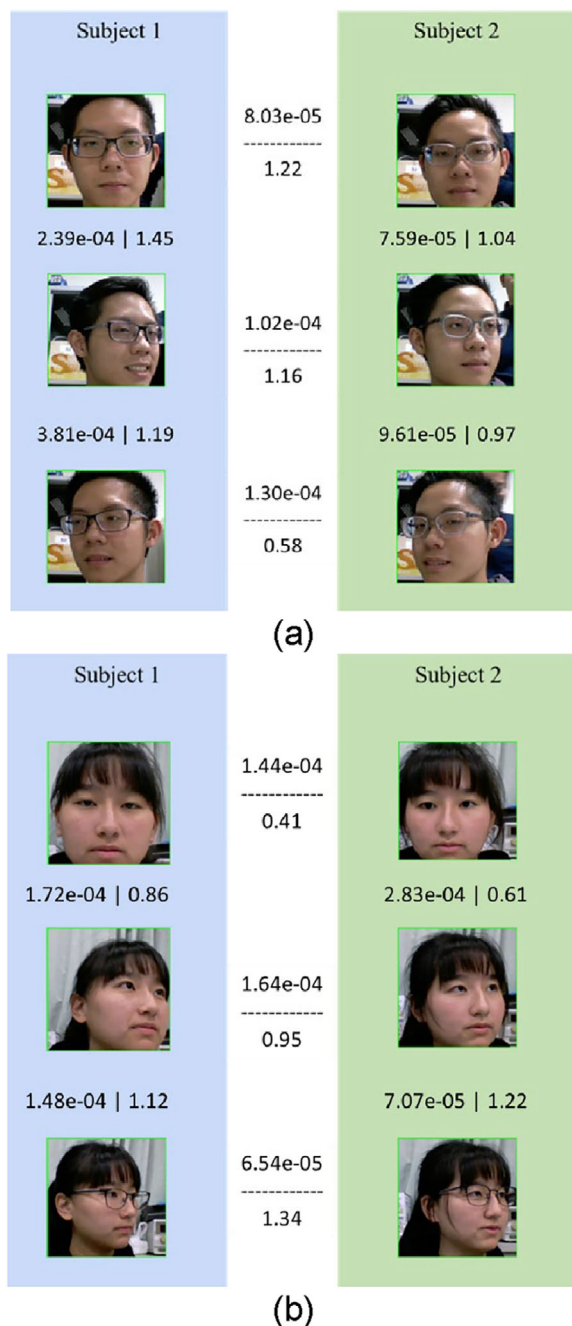
The above findings highlight the effectiveness of the modified model structures in improving accuracy while maintaining efficiency and meeting the demands of real-time performance.



**FIGURE 15** Subjects of female twins; (a) Subject 1 and (b) Subject 2.

## 5 | CONCLUSIONS

In this study, a novel DL model called FN8 with an exceptionally lightweight architecture was proposed and tested for deep face



**FIGURE 16** Similarity map between twin pair subjects at various positions; (a) male twins and (b) female twins. Between images are the distance measured by FaceNet and FN8, respectively.

recognition. The experimental results showed that the proposed model outperformed state-of-the-art lightweight face models in terms of accuracy and robustness. The reduced complexity of the proposed lightweight model makes it particularly attractive due to its low memory requirement of only 3.5 MB and a small volume of parameters at the cost of losing very few in accuracy. By using low-shot learning from grayscale facial images as the model input to generate face embeddings, FN8 can still preserve the strong facial features to discern different identities and identify an undisclosed one. Extensive experimental evaluations

were conducted to test the proposed approach in complex and uncontrolled situations, which confirmed the effectiveness of the model. This lightweight face recognition model is feasible for real-time deployment on resource-constrained devices and exhibits practical significance and applied potential.

## AUTHOR CONTRIBUTIONS

Conceptualization, Z.-Y. Deng, H.-H. Chiang, L.-W. Kang, and H.-C. Li; Methodology, Z.-Y. Deng and H.-H. Chiang; Software, Z.-Y. Deng; analysis and validation, Z.-Y. Deng, and H.-C. Li; Investigation, H.-H. Chiang, and L.-W. Kang; Resources, Z.-Y. Deng, and L.-W. Kang; Data curation, H.-H. Chiang, L.-W. Kang, and H.-C. Li; Writing—original draft preparation, Z.-Y. Deng and H.-C. Li; Writing—review and editing, H.-H. Chiang and H.-C. Li; Project administration, H.-H. Chiang and H.-C. Li; Funding acquisition, H.-H. Chiang. All authors have read and agreed to the published version of the manuscript.

## CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

## DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

## PERMISSION TO REPRODUCE MATERIALS FROM OTHER SOURCES

None.

## REFERENCES

- Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: A unified embedding for face recognition and clustering. In: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 815–823. IEEE, Piscataway (2015)
- Li, H.-C., Deng, Z.-Y., Chiang, H.-H.: Lightweight and resource-constrained learning network for face recognition with performance optimization. *Sensors* 20(21), 6114 (2020). <https://doi.org/10.3390/s20216114>
- Zhong, Y., Deng, W., Fang, H., Hu, J., Zhao, D., Li, X., Wen, D.: Dynamic training data dropout for robust deep face recognition. *IEEE Trans. Multim.* 24, 1186–1197 (2022). <https://doi.org/10.1109/TMM.2021.3123478>
- Shi, X., Chai, X., Xie, J., Sun, T.: MC-GCN: A multi-scale contrastive graph convolutional network for unconstrained face recognition with image sets. *IEEE Trans. Image Process.* 31, 3046–3055 (2022). <https://doi.org/10.1109/TIP.2022.3163851>
- Martínez-Díaz, Y., Nicolás-Díaz, M., Méndez-Vázquez, H., Luevano, L.S., Chang, L., González-Mendoza, M., Sucar, L.E.: Benchmarking lightweight face architectures on specific face recognition scenarios. *Artif. Intell. Rev.* 54, 6201–6244 (2021). <https://doi.org/10.1007/s10462-021-09974-2>
- Liu, S., Song, Y., Zhang, M., Zhao, J., Yang, S., Hou, K.: An identity authentication method combining liveness detection and face recognition. *Sensors* 19(21), 4733 (2019). <https://doi.org/10.3390/s19214733>
- Lee, H., Park, S.-H., Yoo, J.-H., Jung, S.-H., Huh, J.-H.: Face recognition at a distance for a stand-alone access control system. *Sensors* 20(3), 785 (2020). <https://doi.org/10.3390/s20030785>
- Wang, Z., Zhang, X., Yu, P., Duan, W., Zhu, D., Cao, N.: A new face recognition method for intelligent security. *Appl. Sci.* 10(3), 852 (2020). <https://doi.org/10.3390/app10030852>
- Song, Z., Nguyen, K., Nguyen, T., Cho, C., Gao, J.: Spartan face mask detection and facial recognition system. *Healthcare* 10(1), 87 (2022). <https://doi.org/10.3390/healthcare10010087>

10. Sun, J., Lu, S.: An improved single shot multibox for video-rate head pose prediction. *IEEE Sens. J.* 20(20), 12326–12333 (2020). <https://doi.org/10.1109/JSEN.2020.2999625>
11. Abdi, H., Williams, L.J.: Principle component analysis. Wiley Interdiscip. Rev. Comput. Stat. 2, 433–459 (2010)
12. Lu, J., Plataniotis, K.N., Venetsanopoulos, A.N.: Face recognition using LDA-based algorithms. *IEEE Trans. Neural Netw.* 14(1), 195–200 (2003). <https://doi.org/10.1109/TNN.2002.806647>
13. Santaji, G., Ghorpade, J., Shamla, M.: Pattern recognition using neural networks. *Int. J. Comput. Sci. Inf. Technol. Res. (IJCSIT)* 2(6), 92–98 (2010)
14. Wang, F., Xie, F., Shen, S., Huang, L., Sun, R., Le Yang, J.: A novel multi-face recognition method with short training time and lightweight based on ABASNet and H-Softmax. *IEEE Access* 8, 175370–175384 (2020). <https://doi.org/10.1109/ACCESS.2020.3026421>
15. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: DeepFace: Closing the gap to human-level performance in face verification. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708. IEEE, Piscataway (2014). <https://doi.org/10.1109/CVPR.2014.220>
16. Sun, Y., Chen, Y., Wang, X., Tang, X.: Deep learning face representation by joint identification-verification. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, vol. 2, pp. 1988–1996. ACM, New York (2014)
17. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition. In: British Machine Vision Conference (BMVC), pp. 41.1–41.12. Springer-Verlag, London (2015)
18. Almabdy, S., Elrefaei, L.: Deep convolutional neural network-based approaches for face recognition. *Appl. Sci.* 9(20), 4397 (2019) <https://doi.org/10.3390/app9204397>
19. Amos, B., Ludwiczuk, B., Satyanarayanan, M.: Openface: A general-purpose face recognition library with mobile applications. *CMU School Comp. Sci.* 6(2), 20 (2016)
20. Duong, C.N., Quach, K.G., Jalata, I., Le, N., Luu, K.: MobiFace: A lightweight deep learning face recognition on mobile devices. In: 2019 IEEE 10th International Conference on Biometrics Theory, Applications, and Systems (BTAS), pp. 1–6. IEEE, Piscataway (2019) <https://doi.org/10.1109/BTAS46853.2019.9185981>
21. Martínez-Díaz, Y., Luevano, L.S., Méndez-Vázquez, H., Nicolás-Díaz, M., Chang, L., González-Mendoza, M.: ShuffleFaceNet: A lightweight face architecture for efficient and highly-accurate face recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, Piscataway (2019)
22. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: SphereFace: Deep hypersphere embedding for face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 212–220. IEEE, Piscataway (2017)
23. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: CosFace: Large margin cosine loss for deep face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5265–5274. IEEE, Piscataway (2018)
24. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: ArcFace: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4690–4699. IEEE, Piscataway (2019)
25. Yan, M., Zhao, M., Xu, Z., Zhang, Q., Wang, G., Su, Z.: VarGFaceNet: An efficient variable group convolutional neural network for lightweight face recognition. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 2647–2654. IEEE, Piscataway (2019) <https://doi.org/10.1109/ICCVW.2019.00323>
26. Chen, S., Liu, Y., Gao, X., Han, Z.: MobileFaceNets: Efficient CNNs for accurate real-time face verification on mobile devices. In: Biometric Recognition CCBR 2018, 10996. Springer, Cham (2018) [https://doi.org/10.1007/978-3-319-97909-0\\_46](https://doi.org/10.1007/978-3-319-97909-0_46)
27. Kocacinar, B., Tas, B., Akbulut, F.P., Catal, C., Mishra, D.: A real-time CNN-Based lightweight mobile masked face recognition system. *IEEE Access* 10, 63496–63507 (2022). <https://doi.org/10.1109/ACCESS.2022.3182055>
28. Zhang, P., Zhao, F., Liu, P., Li, M.: Efficient lightweight attention network for face recognition. *IEEE Access* 10, 31740–31750 (2022). <https://doi.org/10.1109/ACCESS.2022.3150862>
29. Boutros, F., Siebke, P., Klemm, M., Damer, N., Kirchbuchner, F., Kuijper, A.: PocketNet: Extreme lightweight face recognition network Using Neural Architecture Search and Multistep Knowledge Distillation. *IEEE Access* 10, 46823–46833 (2022). <https://doi.org/10.1109/ACCESS.2022.3170561>
30. Hoo, S.C., Ibrahim, H., Suandi, S.A.: ConvFaceNeXt: Lightweight networks for face recognition. *Mathematics* 10(19), 3592 (2022). <https://doi.org/10.3390/math10193592>
31. Martínez-Díaz, Y., Méndez-Vázquez, H., Luevano, L.S., Nicolás-Díaz, M., Chang, L., González-Mendoza, M.: Towards accurate and lightweight masked face recognition: An experimental evaluation. *IEEE Access* 10, 7341–7353 (2022). <https://doi.org/10.1109/ACCESS.2021.3135255>
32. Zhao, G., Yang, H., Yu, M.: Expression recognition method based on a lightweight convolutional neural network. *IEEE Access* 8, 38528–38537 (2020). <https://doi.org/10.1109/ACCESS.2020.2964752>
33. Agbo-Ajala, O., Viriri, S.: A lightweight convolutional neural network for real and apparent age estimation in unconstrained face images. *IEEE Access* 8, 162800–162808 (2020). <https://doi.org/10.1109/ACCESS.2020.3022039>
34. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. In: European Conference on computer vision (ECCV), vol. 9911. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46478-7\\_31](https://doi.org/10.1007/978-3-319-46478-7_31)
35. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on Machine Learning, vol. 37, pp. 448–456. PMLR, New York (2015)
36. Koch, G., Zemel, R., Salakhutdinov, R.: Siamese neural networks for one-shot image recognition. In: Proceedings of 2015 International Conference on Machine Learning (ICML). International Machine Learning Society, Madison, WI (2015)
37. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst (2007)
38. Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I., Zafeiriou, S.: Agedb: The first manually collected, in-the-wild age database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 51–59. IEEE, Piscataway (2017)
39. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch. arXiv preprint, arXiv:1411.7923 (2014)
40. Wu, Y., Liu, H., Fu, Y.: Low-Shot Face Recognition with Hybrid Classifiers. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), pp. 1933–1939. IEEE, Piscataway (2017). <https://doi.org/10.1109/ICCVW.2017.228>

**How to cite this article:** Deng, Z.-Y., Chiang, H.-H., Kang, L.-W., Li, H.-C.: A lightweight deep learning model for real-time face recognition. *IET Image Process.* 17, 3869–3883 (2023). <https://doi.org/10.1049/ipr2.12903>