Feature Fusion Strategies for Enhanced 2.5D Face Recognition Using EfficientNet-B4

Min-Er Teo2, a), Lee-Ying Chong1, 2, b), Siew-Chin Chong1, 2, c)

*1Faculty of Information and Technology, Multimedia University, Jalan Ayer Keroh Lama, 75450 Bukit Beruang, Melaka, Malaysia  
2Centre for Advanced Analytics, COE of Artificial Intelligence, Multimedia University, Jalan Ayer Keroh Lama, 75450 Bukit Beruang, Melaka, Malaysia*

*b) Corresponding author: lychong@mmu.edu.my*

*a)1171200304@student.mmu.edu.my*

c)*chong.siew.chin@mmu.edu.my*

**Abstract.** Face recognition is emerging as one of the most widely used biometrics technologies for verifying an individual's identity across several applications. To enhance the precision and robustness of such systems, the 2.5D face recognition approach leverages depth information from 2.5D data (depth images), which provides additional discriminative features. This paper proposes a feature fusion 2.5D face recognition system using EfficientNet-B4 and numerous optimizers. The system's performance can be further improved by integrating feature fusion techniques with deep learning methods. Feature fusion allows for the combination of features extracted from depth images, which contain rich discriminative information that enhances recognition accuracy. Evaluation results show combining two distinct feature sets led to an improved recognition rate. Among the nine fusion techniques tested, the *dy\_fd* feature fusion method achieved the highest recognition rate of 98.10% by implementing the RMSprop optimizer after five fine-tuning epochs.

# Introduction

Generally, 2.5D face recognition represents a simple form of 3D face recognition, which leverages multiple 2.5D partial data points retrieved from several positions to construct a face model. A depth image refers to a 2D matrix structure that carries *z*-coordinate, indicating depth value [1]. However, depth images encounter several challenges similar to those faced by 2D images, including occlusions, noise within development, perspective distortion, and a shortage of 3D connection, as they share certain characteristics with 2D images [2]. Additionally, 2.5D face recognition still faces similar challenges as 2D face recognition, for instance, the distinction of facial posture and face expressions [3]. Due to the 2.5D face is known as the simplified type of a 3D face, it uses data from only a single viewpoint, resulting in a limited face model that provides only partial 3D information [2,4]. On the other sides, although 2.5D has the above-mentioned drawbacks, it also contains some advantages. Besides the depth value (*z*-coordinate), numerous types of geometric features can be retrieved from the depth image. The hidden features that can be extracted from the depth image contain distinct features which can increase the effectiveness of the system [5]. Examples of geometric features that can be extracted from the depth image include Coefficients of the Fundamental Forms, Derivatives, and others [6].

Thus, a 2.5D face recognition system with the use of feature fusion methods, which produce different types of geometric features is proposed to enhance the performance of the system in this paper. The 2.5D face recognition framework proposed in this research utilises the EfficientNet-B4 model with transfer learning to reduce computational complexity and enhance the efficiency of the model. A number of optimizers have been used to alter the parameters of the system, including, but not limited to learning rates, weights, and others, to improve training performance. The most suited optimizer of the system is found through evaluation of different optimizers. Several evaluations are carried out to discover the best geometric features with the best optimizers of EfficientNet that boost the recognition rate of the system.

# Related works

Feature fusion is the process of combining different features from different parts or areas of a structure. In a face recognition system, feature fusion means the merging and fusing of two different algorithms, hence it improves the system efficiency and reliability [7]. Furthermore, deep-learning models like CNN are typically executed alongside the application of feature fusion techniques to promote the performance in terms of accuracy.

Dutta et al. [8] introduced a computational framework for facial features derived from depth points captured via the depth image. The approach segments various facial images into four main components ('CC'), potentially generating 36 extra hybrid segments through the integration of a mathematical model and a straightforward data-level combining method. The 'CC face space' is a novel facial image space created from these 40 faces. A two-stage PSO method enhances recognition outcomes by minimizing the features required to depict the range of a face image. The proposed method has been tested successfully on three prominent 3D face databases, yielding encouraging results. Additionally, AlFawwaz et al. [9] proposed a method for MDCT merging known as the FFLFRM, which encompasses face identification, feature extraction, feature merging, and face classification. The MDCT fusion method was used initially, and then the classification was performed using Artificial Neural Network (ANN). Achieving a 97.70% accuracy rate, the MDCT-based method displayed promising recognition outcomes, highlighting its effectiveness and robustness in overcoming challenges

A fusion-based 2.5D face recognition that utilising ELM classifier and GRCM feature descriptor is recommended by Chong et al. [1]. Several feature fusions are used for generating the covariance matrix in the block-based GRCM. After that, ELM is utilised for classification once the covariance matrix has been converted into the feature vector by employing manifold flattening. In order to improve the accuracy rate, feature fusions employing 2.5D and 2D images have also been tested. Additionally, Xu et al. [10] suggested a unique DMDNet leveraging the DIIF to reduce noise and increase the resolution of face-depth images in low-quality 3D technologies. Furthermore, they constructed a LDNFNet, utilising multibranch fusion blocks to discover matching properties between depth and normal images. Experimental findings demonstrate that the suggested approach, which makes use of the Adam optimizer, attains an 88.94% accuracy rate using the Lock3DFace database.

Based on the above literature review, although many studies have achieved promising results in 3D face recognition, the feature fusion procedures they employed often involve multiple steps and configurations, making them complex and difficult to implement. Moreover, the overall processes are computationally intensive and take longer computation times due to the complexity of the feature fusion methods. To address these challenges, this study proposes a 2.5D face recognition system that leverages feature fusion techniques to collect distinct forms of geometric features from depth images. Given the simplicity of extracting geometric features from depth images, the proposed method offers a more efficient and less time-consuming workflow compared to the aforementioned approaches.

# METHODOLOGY

Figure 1 illustrates the proposed methodology, which consists of five main steps: processing the data, feature fusion, image augmentation, feature extraction, as well as EfficientNet model training. First, raw face data is prepared in the data preprocessing step to generate a depth (range) image. Next, in the feature fusion step, new features are formed from the depth image. Following this, the fused data goes through the feature extraction and classification phase to extract distinctive features and classify individuals into specific categories. Finally, the recognition rate is then computed using the outcomes generated by the classification stage.

A diagram of a data processing process

AI-generated content may be incorrect.

**Figure 1.** The proposed fusion-based 2.5D face recognition using EfficientNet-B4

## Data Preprocessing

The partial 2.5D facial image is preprocessed in this suggested technique to reduce noise, standardise size, as well as generate zero-mean normalisation. To generate a standardised depth image, the 2.5D data preprocessing steps include coordinate acquisition, depth image computation, face cropping, face normalising, and face standardising. Initially, the *x*, *y*, and *z* coordinates obtained from the 3D point coordinates of a person's face are defined as the 2.5D data. Once the extraneous background setup is eliminated, a facial area is obtained utilising the 2.5D information. The depth image is created by using interpolation to project the depth value onto the *x*-*y* plane. Subsequently, the image is trimmed based on the locations of the eyes and mouth to create a standard depth image. Ultimately, a standard modification has been carried out to standardize the depth image, guaranteeing that both the variance and mean equal zero.

## Feature Fusion

Feature fusion exists as a type of approach that merges and combines the features or traits from different portions of an object [11]. It is often utilised in face recognition to boost the performance of the system by integrating the different unique features of a person. The depth image captures a large quantity of face geometrical information, which can be used to derive traits for fusion [12]. The fusion between different geometric features derived from the depth image is employed as one of the fusion methods in this paper. There are nine different fusion methods have been explored to determine which fusion approach outperforms the depth image. Table 1 below shows the description of the feature fusions.

**TABLE 1.** Description of feature fusion

|  |  |
| --- | --- |
| **Feature Fusion** | **Description** |
| *dy\_fd* | Fusion that is formed by retrieving the first derivative of *z* with respect to *y*-direction of the depth image. |
| *dxdy\_wav\_mean* | Fusion that is produced by extracting the wavelet of *dx\_fd* and *dy\_fd* and calculating the mean wavelet pixel values of the *dx\_fd* and *dy\_fd* features. |
| *dxdy\_sum* | Fusion that is formed by summing up the *dx\_fd* and *dy\_fd’s* pixel values. |
| *dxdy\_swt\_min* | Fusion that is created by acquiring the Stationary Wavelet Transform (SWT) of *dx\_fd* and *dy\_fd* and subsequently choosing the minimum SWT pixel values from *dx\_fd* and *dy\_fd*. |
| *dx\_fd* | Fusion that is formed by extracting the first derivative of *z* with respect to *x*-direction of the depth image. |
| *dxdy\_swt\_max* | Fusion that is produced by retrieving the Stationary Wavelet Transform (SWT) of *dx\_fd* and *dy\_fd* and selecting the maximum SWT pixel values from *dx\_fd* and *dy\_fd.* |
| *dxdy\_max* | Fusion that is formed by choosing the largest pixel values from the *dx\_fd* and *dy\_fd.* |
| *dxdy\_swt\_mean* | Fusion that is created by obtaining the Stationary Wavelet Transform (SWT) of *dx\_fd* and *dy\_fd* and calculating the mean SWT pixel values of *dx\_fd* and *dy\_fd.* |
| *dxdy\_swt\_abs\_max* | Fusion that is formed by retrieving the Stationary Wavelet Transform (SWT) of *dx\_fd* and *dy\_fd* and choosing the absolute maximum SWT pixel values from *dx\_fd* and *dy\_fd.* |

## Image Augmentation

Image augmentation is a strategy that generates additional data for model training by altering existing images. In order to expand the quantity of data used for model training and improve the efficiency of the system, it is crucial to implement the image augmentation technique [13]. This research consisted of a variety of image augmentation techniques, which involved image flipping, rotation, and shifting. The training data's images are randomly flipped horizontally by applying the horizontal flip strategy. The zoom range method randomly changes how much a picture zooms in on training data. The rotation range is used to alter the pictures at random through various degrees. The aforementioned strategies improve model reliability and diversity by allowing alternative image processing procedures to be applied to the same picture.

## Feature Extraction

Feature extraction serves as an essential process that gathers both vital and minor traits of a raw image with the goal to attain fine-grained image recognition, especially when employing the techniques of deep learning. Transfer learning is implemented in this research for applying and transferring previously determined features from a huge database to a specific task. By gathering high-level attributes collected from the source image, this framework works similarly as a feature extractor. These features are subsequently displayed as convolutional layers of learnt illustrations. The following layers can subsequently utilise these representations to extract traits for categorisation. The feature extraction method is employed in combination with the EfficientNet-B4 pre-trained model to retrieve distinctive characteristics of the face data.

## EfficientNet-B4

Tan and Le [14] developed EfficientNet, a CNN architecture and scaling approach that leverages a compound coefficient strategy for equally scaling each dimension of resolution, width, and depth. Unlike the conventional process, which modifies network width, resolution, and depth at random, the EfficientNet scaling method uniformly raises these parameters by using a set of predetermined scaling factors. By employing a set of predetermined scaling variables, the compound scaling technique is the primary and potent innovation that can reliably adjust the entire framework in three dimensions (resolution, depth, width). By adjusting these settings, EfficientNet could accomplish different levels of effectiveness and model size. EfficientNet-B4 serves as the EfficientNet structure's members [15]. Compound scaling approaches make up EfficientNet-B4 structures, which equitably expand the entire network’s depth, resolution, and width. As a result, it enables exceptional precision at a minimal processing expense [16]. The EfficientNet-B4 framework first specifies its base model, which consists of the B4 model utilising distinct arguments. The B4 model is rebuilt after the top layer is removed, and the max pooling layer is implemented to identify the feature map’s highest values. To avoid the overfitting issue, the framework is also enhanced with the BatchNormalization layer. The output layer of the EfficientNet-B4 model employs the SoftMax activation function. In this research, 469 layers were frozen to add new features, and it was noted that this quantity of frozen layers produced the optimal results. In the end, the result is assessed by unfreezing and re-educating each layer. The design of EfficientNet-B4 that is suggested can be observed in Figure 2. The evaluation of performance involves unfreezing and retraining each layer.

A diagram of a diagram

AI-generated content may be incorrect.

**Figure 2.** The procedure of the recommended EfficientNet-B4 architecture

## Recognition Rate

The precision of each biometric system is a crucial feature to evaluate as it indicates the reliability and efficiency of a system. Various assessment methods can be employed to gauge the efficiency of the system. True positive (TP) refers to the total instances where the system accurately identifies the identical individual in two different images. Conversely, the overall count of the system that accurately identifies two distinct individuals in the pictures is referred to as the true negative (TN). Additionally, the algorithm identifies a false positive (FP) when it recognizes two distinct individuals as being the same person. Conversely, the total in which the system mistakenly identified one individual as two distinct persons is referred to as the false negative (FN).

(1)

The formula for the recognition rate can be obtained from (1) above. The recognition rate is a measurement that shows how effectively an individual can be detected as a legitimate user of the system. The precision is typically calculated by taking the sum of correctly identified images (TP and TN) and dividing it by the total number of photos.

# EXPERIMENTS AND DISCUSSIONS

This study utilises the Face Recognition Grand Challenge version 2 (FRGC v2.0) to perform different experiments [17]. This study also utilised a subgroup from the FRGC v2.0 dataset, which consists of 254 individuals. The collection consists of 3,860 images, all of which have been cropped and normalized to 72×60 pixels. The intensity of pixels in each image has been normalized. The dataset has been separated into three categories: testing, training, and validation through the implementation of the stratified k-fold cross-validation method. This approach is particularly useful in situations of class imbalance, where certain groups contain more data than others. It ensures that the class distribution in each fold closely resembles that of the original dataset. Since every collection includes a typical sample from each class, this approach minimizes the chances of bias during model assessment. Given that every subject in the FRGC v2.0 database contains a minimum of eight images, a k-fold value of 8 was applied to split the dataset in this study.

Additionally, the data augmentation method is utilised to boost the number of images in the dataset, guaranteeing that the system possesses sufficient data for training. The training set comprises augmentation techniques like 10% zoom, 20% rotation of images, and horizontal flipping. Prior to fitting the dataset in the training phase, multiple parameters need to be established. In this experiment, a batch size of 32 was applied. This study utilised different optimizers—RMSprop, Nadam, Adam, and Adamax to identify which one works best for the system. All optimizers were set with the same learning rate of 0.001. The system performs a multiclass classification task using the categorical cross-entropy loss function. The EfficientNet-B4 model was fine-tuned by primarily freezing the first 469 layers. It then underwent training for several epochs—specifically 5, 10, 20, 30, 40, and 50—to identify the optimal number of epochs that allow the system to effectively learn the dataset's features. To enable further learning from previously acquired features, all layers were subsequently unfrozen and retrained with a larger number of epochs, set to 100. The model also employs early stopping to avoid overfitting by monitoring the validation loss. Training is automatically terminated when the validation loss ceases to decrease.

Table 2 shows the recognition rate of feature fusion after fine-tuning with various values of epochs along with numerous types of optimizers. Depth images act as the baseline of the system. Four different types of optimizers such as RMSprop, Adamax, Nadam, and Adam are utilised in all the experiments. It can be observed that among the feature fusions, the *dy\_fd* gains the highest recognition rate of 98.10% using the RMSprop optimizer with 5 fine-tuning epochs. This may be caused by the features *dy* that retrieved from the depth image contains more distinctive traits that can increase the recognition rate of the system than using only the original depth image. Moreover, the *dxdy\_wav\_mean* represents the feature fusion that achieved the second-highest accuracy rate of 97.94 with the RMSprop optimizer via 10 fine-tuning epochs among the other feature fusions. This feature fusion that incorporates the integration between the wavelet features extracted from both *dx* and *dy* features contains unique characteristics to increase the effectiveness of the system. As the feature fusion technique incorporates two different features, it can help to mix up two different features, hence the system’s recognition rate can be improved than using only a sole depth image.

**TABLE 2.** Recognition rate (%) of feature fusions of the proposed method using EfficientNet-B4 model with different no. of epochs respect to various optimizers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Geometric** | **No. of epochs** | **Recognition Rate (%)** | | | |
| **features** | **(fine-tuning)** | **Adam** | **Nadam** | **Adamax** | **RMSprop** |
| **depth image** | 5 | **97.93** | 95.85 | 95.44 | 96.89 |
|  | 10 | 96.06 | 96.27 | 92.74 | 96.68 |
|  | 20 | 95.85 | 95.64 | 93.15 | 95.85 |
|  | 30 | 94.81 | 96.06 | 89 | 95.64 |
|  | 40 | 96.47 | 95.23 | 92.74 | 97.1 |
|  | 50 | 96.27 | 93.57 | 93.98 | 97.51 |
| ***dy\_fd*** | 5 | 94.81 | 92.95 | 89.21 | **98.1** |
|  | 10 | 95.02 | 93.15 | 83.4 | 95.85 |
|  | 20 | 95.44 | 93.98 | 84.65 | 96.26 |
|  | 30 | 92.94 | 93.16 | 83.82 | 95.43 |
|  | 40 | 95.86 | 94.4 | 84.64 | 94.82 |
|  | 50 | 92.74 | 96.47 | 88.38 | 96.06 |
| ***dxdy\_wav\_mean*** | 5 | 90.87 | 94.19 | 86.1 | 94.4 |
|  | 10 | 90.04 | 93.78 | 85.48 | **97.94** |
|  | 20 | 90.46 | 91.05 | 88.17 | 93.15 |
|  | 30 | 91.29 | 90.46 | 84.65 | 91.7 |
|  | 40 | 90.66 | 93.36 | 81.33 | 94.19 |
|  | 50 | 91.18 | 92.53 | 87.76 | 94.81 |
| ***dxdy\_sum*** | 5 | 90.66 | 93.98 | 89.83 | 94.4 |
|  | 10 | 92.95 | 93.57 | 80.91 | **96** |
|  | 20 | 91.08 | 91.91 | 88.38 | 94.19 |
|  | 30 | 92.53 | 95.23 | 90.46 | 94.39 |
|  | 40 | 83.82 | 92.74 | 88.59 | 94.4 |
|  | 50 | 91.29 | 92.53 | 88.58 | 93.57 |
| ***dxdy\_swt\_min*** | 5 | 90.66 | 93.15 | 88.17 | 94.19 |
|  | 10 | 94.61 | 94.2 | 88.38 | 94.4 |
|  | 20 | 94.39 | 91.29 | 82.99 | **95.85** |
|  | 30 | 91.91 | 94.81 | 85.06 | 92.12 |
|  | 40 | 90.87 | 93.98 | 83.61 | 92.53 |
|  | 50 | 92.54 | 92.32 | 85.89 | 93.78 |
| ***dx\_fd*** | 5 | 89.21 | 91.29 | 84.23 | **95.23** |
|  | 10 | 92.12 | 91.49 | 80.91 | 93.36 |
|  | 20 | 91.49 | 92.12 | 84.23 | 93.15 |
|  | 30 | 91.08 | 91.49 | 81.33 | 93.36 |
|  | 40 | 89.21 | 91.07 | 79.88 | 93.83 |
|  | 50 | 89.42 | 89.63 | 73.44 | 94.19 |
| ***dxdy\_swt\_max*** | 5 | 90.25 | 90.87 | 86.51 | 91.29 |
|  | 10 | 88.8 | 91.29 | 82.37 | **95** |
|  | 20 | 85.68 | 91.3 | 78.63 | 94.4 |
|  | 30 | 90.46 | 89.83 | 81.33 | 90.04 |
|  | 40 | 90.87 | 92.32 | 84.02 | 94.39 |
|  | 50 | 89.42 | 89.21 | 82.37 | 91.91 |
| ***dxdy\_max*** | 5 | 88.17 | 91.08 | 79.25 | 92.12 |
|  | 10 | 91.49 | 89 | 82.78 | 89.83 |
|  | 20 | 82.78 | 84.65 | 84.02 | 92.53 |
|  | 30 | 87.97 | 87.76 | 80.49 | **94.81** |
|  | 40 | 87.96 | 91.91 | 83.2 | 92.74 |
|  | 50 | 89.42 | 88.59 | 81.33 | 82.37 |
| ***dxdy\_swt\_mean*** | 5 | 91.49 | 92.95 | 86.93 | 92.94 |
|  | 10 | 90.04 | 91.91 | 83.33 | 93.98 |
|  | 20 | 90.25 | 91.9 | 86.1 | 90.66 |
|  | 30 | 89.21 | 90 | 85.89 | **94.61** |
|  | 40 | 89.63 | 90.25 | 86.27 | 94.6 |
|  | 50 | 91.08 | 93.36 | 84.85 | 92.74 |
| ***dxdy\_swt\_abs\_max*** | 5 | 86.31 | 87.34 | 74.9 | 92.95 |
|  | 10 | 91.91 | 86.93 | 84.23 | 91.29 |
|  | 20 | 86.72 | 90.46 | 78.42 | 90 |
|  | 30 | 90.04 | 91.07 | 81.74 | 93.15 |
|  | 40 | 89.63 | 91.7 | 84.02 | **93.98** |
|  | 50 | 85.48 | 83.2 | 79.05 | 91.9 |

A graph with black and white lines

AI-generated content may be incorrect.

**Figure 3.** Computation time (s) for depth image and *dy\_fd*

# CONCLUSION

This paper proposed a feature fusion 2.5D face recognition using an EfficientNet-B4 pre-trained model with various number of epochs and optimizers. In the suggested method, the feature fusion that integrates two different features will help to increase and increase the system’s accuracy rate. In addition, the deep learning method that retrieves the distinctive features from the fused data can help to enhance the performance of the system. Furthermore, several optimisers are used to modify the system's characteristics, such as weights and learning rates, with the aim of improving training efficacy. Based on the experimental results, *dy\_fd* feature fusion achieved the highest recognition rate of 98.10% by using the RMSprop optimizer with 5 fine-tuning epochs. Additionally, the RMSprop optimizer has been shown to be a dependable and powerful optimiser for deep learning approaches that improve system performance. In future work, additional face databases, like Texas3DFR, the GavabDB, and others, will be taken into consideration for use in further research.

# References

1. L.-Y. Chong, and S.-C. Chong, “Fusion of 2.5D Face Recognition through Extreme Learning Machine via Manifold Flattening,” in *2021 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, (IEEE, Kuala Terengganu, Malaysia, 2021), pp. 18–23.
2. Strachan, Euan, The Application of Range Imaging for Improved Local Feature Representations (Ph.D. Thesis), University of Glasgow, 2013.
3. L.Y. Chong, T.S. Ong, and A.B.J. Teoh, “Feature fusions for 2.5D face recognition in Random Maxout Extreme Learning Machine,” Applied Soft Computing **75**, 358–372 (2019).
4. J.C. Monteiro, T. Freitas, and J.S. Cardoso, “Multimodal Hierarchical Face Recognition using Information from 2.5D Images,” UPjeng **2**(2), 39–54 (2018).
5. M.-E. Teo, L.-Y. Chong, and S.-C. Chong, “Fusion-Based 2.5D Face Recognition System,” JTDE **12**(1), 19–38 (2024).
6. E. Vezzetti, F. Marcolin, and G. Fracastoro, “3D face recognition: An automatic strategy based on geometrical descriptors and landmarks,” Robotics and Autonomous Systems **62**(12), 1768–1776 (2014).
7. L.-Y. Chong, A.B.J. Teoh, T.-S. Ong, and S.-C. Chong, “2.5D Face Recognition under Tensor Manifold Metrics,” in *Neural Information Processing*, edited by C.K. Loo, K.S. Yap, K.W. Wong, A.T. Beng Jin, and K. Huang, (Springer International Publishing, Cham, 2014), pp. 653–660.
8. K. Dutta, D. Bhattacharjee, M. Nasipuri, and O. Krejcar, “Complement component face space for 3D face recognition from range images,” Appl Intell **51**(4), 2500–2517 (2021).
9. B.M. AlFawwaz, A. AL-Shatnawi, F. Al-Saqqar, and M. Nusir, “Multi-Resolution Discrete Cosine Transform Fusion Technique Face Recognition Model,” Data **7**(6), 80 (2022).
10. R. Xu, K. Wang, C. Deng, M. Wang, X. Chen, W. Huang, J. Feng, and W. Deng, “Depth Map Denoising Network and Lightweight Fusion Network for Enhanced 3D Face Recognition,” Pattern Recognition **145**, 109936 (2024).
11. H. Kaur, D. Koundal, and V. Kadyan, “Image Fusion Techniques: A Survey,” Arch Computat Methods Eng **28**(7), 4425–4447 (2021).
12. E. Vezzetti, and F. Marcolin, “3D human face description: landmarks measures and geometrical features,” Image and Vision Computing **30**(10), 698–712 (2012).
13. M. Xu, S. Yoon, A. Fuentes, and D.S. Park, “A Comprehensive Survey of Image Augmentation Techniques for Deep Learning,” Pattern Recognition **137**, 109347 (2023).
14. M. Tan, and Q.V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” (2020).
15. I.N. Alam, I.H. Kartowisastro, and P. Wicaksono, “Transfer Learning Technique with EfficientNet for Facial Expression Recognition System,” RIA **36**(4), 543–552 (2022).
16. M.-E. Teo, L.-Y. Chong, S.-C. Chong, and P.-Y. Goh, “2.5D Face Recognition System using EfficientNet with Various Optimizers,” JOIV : Int. J. Inform. Visualization **8**(4), 2388 (2025).
17. P.J. Phillips, P.J. Flynn, T. Scruggs, K.W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, “Overview of the Face Recognition Grand Challenge,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, (IEEE, San Diego, CA, USA, 2005), pp. 947–954.