International Journal of Electrical and Computer Engineering (IJECE)
Vol. 16, No. 3, June 2026, pp. 1213~1226
ISSN: 2088-8708, DOI: 10.11591/ijece.v16i3.pp1213-1226

Optimization of transfer learning for facial emotion classification on the FER-2013 dataset

Nida Muhliya Barkah, Shofwatul 'Uyun
Department of Informatics, Universitas Islam Negeri Sunan Kalijaga Yogyakarta, Indonesia

Article Info

Article history:

Received Oct 15, 2025
Revised Jan 30, 2026
Accepted Mar 16, 2026

Keywords:

Deep learning
Emotion
Emotion recognition
Image classification
Transfer learning

ABSTRACT

Facial expressions play a key role in non-verbal communication by naturally reflecting human emotions. Facial emotion recognition (FER) using computer vision has gained attention with advances in deep learning. However, deep learning models require large datasets to perform well, posing a challenge for FER tasks with limited data. Transfer learning is a promising approach to address this issue, but a standardized method for FER is yet to be established. This study optimizes three transfer learning models ResNet-50, Inception V3, and Xception on the FER-2013 dataset. Experiments include testing input image sizes, hyperparameter tuning, data augmentation, layer addition, and training methods. Results show each model requires different input sizes for best accuracy. Hyperparameter tuning improves accuracy by 6.35%, 4.69%, and 1.04% for ResNet-50, Inception V3, and Xception, respectively. Augmenting only the disgust class yields better accuracy than augmenting all classes. The freeze fine-tuning method is less effective than fine-tuning alone on datasets with thousands of samples but outperforms the freeze layer method. The best accuracies achieved are 64.89% (ResNet-50), 65.83% (Xception), and 66.40% (Inception V3). These findings provide insights into freeze fine-tuning limitations and guidance for optimizing transfer learning in FER with limited data.

Corresponding Author:

Nida Muhliya Barkah
Department of Informatics, Universitas Islam Negeri Sunan Kalijaga Yogyakarta
Jl. Laksda Adisucipto, Depok, Sleman, 55281, Daerah Istimewa Yogyakarta
Email: nidamuhliya35@gmail.com

1. INTRODUCTION

Emotions or facial expressions are crucial in non-verbal communication and are one of the most natural ways to convey internal human feelings in personal interactions [1]. Emotions are generated from the movements of facial muscles and can be used to convey a range of signals in communication, from indicators of danger to subtle communicative signals [2]. Most psychological studies show that half of the information conveyed in a conversation comes from the emotions displayed [1]. Paul Ekman, a leading psychologist and researcher in the field of emotion psychology and facial expressions, discovered that humans universally express their emotions through seven distinct facial expression patterns: happiness, sadness, anger, fear, surprise, disgust, and neutrality [3].

Facial emotion recognition is one of the research branches of face recognition. Facial recognition is a biometrics technology that uses face recognition algorithms to recognize an individual by comparing an input facial image and a database of faces to find the face that best matches the input image [4]. Information about emotions or facial expressions obtained through image processing allows the system to operate similarly to how the human brain processes and recognizes information. The availability of this rich

information provides crucial clues for the system to interpret human intentions. Therefore, facial expression recognition has broad applications in various fields and is becoming a crucial component in the development of affective computing [1]. Emotion recognition is influenced by factors such as lighting, pose, background, viewing angle, the camera used to capture the image, and occlusion, which occurs when the face is blocked by an object [5]. Emotion recognition can be performed using facial image datasets, which then go through pre-processing and analysis stages using pattern recognition and machine learning techniques such as artificial intelligence (AI), computer vision, and deep learning [6].

Deep learning is a branch of machine learning where algorithms are inspired by the structure of the human brain, leading to the creation of artificial neural networks (ANN). Deep learning encompasses several algorithms, including long short term memory networks (LSTM), recurrent neural networks (RNN), self-organizing maps (SOM), and convolutional neural networks (CNNs) [7]. CNNs are a type of deep learning artificial neural network widely used in digital image analysis, proving to be highly effective in various pattern recognition and image classification tasks [8]. Transfer learning is an approach in deep learning that utilizes existing knowledge from one task to assist in model training on a different task. It can reduce the reliance on large amounts of labeled data and training costs, allowing the adaptation of trained models to new tasks [9].

Most early research on facial emotion recognition used laboratory-developed datasets, such as JAFFE [10] and CK+ [11]. Such laboratory datasets have the disadvantage of being too uniform, as they typically include only positive expressions without occlusion, which makes them less applicable to complex real-life situations. To overcome this problem, many facial emotion recognition studies have developed datasets collected in unrestricted, real-world environments [12], [13]. Among all datasets, the FER-2013 dataset is one of the most commonly used ones, as it contains a large number of facial images captured in an unconstrained setting but still has drawbacks such as low image resolution (only 48×48 pixels), imbalanced class distribution, and expressions can vary greatly between individuals. This makes the emotion classification process more difficult. Transfer learning is very effective in addressing this challenge, as it allows models pre-trained on large datasets to be applied to smaller datasets like FER-2013, improving performance despite limited data.

Several previous studies have shown that transfer learning is superior to building a model from scratch in facial emotion recognition. A study by Hung *et al.* [14] using Dense_FaceLiveNet with a two-stage transfer learning approach on the JAFFE, KDEF, and FER-2013 datasets successfully increased the accuracy to 91.93%, or 12.9% higher than without transfer learning. Another study comparing two approaches on CNN AlexNet and VGG16 with the RaFD dataset also showed that transfer learning not only produced higher accuracy (98.33%) but also significantly accelerated the training time [15]. The use of Inception V3 and MobileNet-V2 pretrained models on the Emognition dataset also produced better performance than models built from scratch, with an accuracy of 96% and an F1-score of 0.95 [16]. In addition, the development of the lightweight RS-Xception model showed that transfer learning improved efficiency and accuracy on various datasets, including FER2013 and RAF-DB [17]. These findings confirm that transfer learning is a more effective approach for facial emotion recognition tasks, especially under limited data and complex environments.

Some research on emotion recognition in the FER-2013 dataset using has been conducted previously. This includes a study by Yen and Li [1], which aims to determine the importance of using transfer learning for facial emotion recognition (FER) and the effect of training datasets and training types on transfer learning. The results of research on five transfer learning models show that class weight is the optimal technique for balancing data, the freeze+fine-tuning training method proposed in this study can improve accuracy without being affected by data size, and multi-stage training significantly improves model accuracy on the FER-2013 dataset. Research conducted by Shahzad *et al.* [18] aims to improve FER performance with a zoning-based method (ZFER) that extracts and divides facial reference points into four regions, using VGG-16 and fully connected neural network (FCNN) models for emotion classification. As a result, the method achieved 98.4% accuracy on the CK+ dataset and 65% on FER-2013, with zoning increasing the accuracy from 98.47% to 98.74% on CK+. Research conducted by Urnisha *et al.* [19] focuses on improving FER using the transfer learning method with MobileNetV2 architecture. The results showed an accuracy of 99% on random images and video clips, and an accuracy value of 61% on the FER-2013 dataset, showing progress in real-time facial expression recognition. Several previous studies have proven that transfer learning and the use of multiple training models can improve accuracy on the FER 2013 dataset. Summary of previous research related to facial recognition can be seen in Table 1.

This study proposes an optimization of three transfer learning: ResNet-50, Inception V3 and Xception for facial emotion recognition on FER 2013 dataset by testing various strategies, such as image resizing, hyperparameter tuning, data augmentation, and the addition of dropout layers and batch normalization. In addition, training methods such as fine-tuning, freezing, and freeze fine-tuning are applied to the pre-trained models to identify the best combination for achieving the highest accuracy. Unlike previous

studies that considered freeze fine-tuning to be superior, the findings reveal that this method is less effective on small datasets like FER-2013. These results provide new insights into the limitations of freeze fine-tuning and offer practical guidance for optimizing transfer learning in facial emotion recognition tasks.

Table 1. Summary of previous research

No	Authors (Year)	Method	Dataset	Input type	Accuracy/Notes
1	[14]	Dense_FaceLiveNet with two-stage transfer learning	JAFFE, KDEF, FER-2013	Image, Video	Accuracy 91.93%, 12.9% higher than without transfer learning
2	[15]	CNN (AlexNet, VGG16) with transfer learning	RaFD	Image	Transfer learning: Accuracy 98.33%, faster training time than from scratch
3	[16]	Inception V3, MobileNet V2 (pretrained)	Emognition	Video to Image	Accuracy 96%, F1-score 0.95, outperformed optimized models from scratch
4	[17]	Lightweight RS-Xception (transfer learning-based)	FER-2013, CK+, Bigfer2013, RAF-DB	Image	Transfer learning improved accuracy and efficiency across datasets
5	[1]	Xception, EfficientNet-B0, Inception, ResNet-50, DenseNet-121	FER-2013	Image	Freeze + fine-tuning improves accuracy; class weights effective for balancing
6	[18]	ZFER with VGG-16 and FCNN	CK+, FER-2013	Image	98.4% (CK+), 65% (FER-2013); zoning improved CK+ accuracy to 98.74%
7	[19]	MobileNetV2 (transfer learning)	FER-2013	Image, Video	99% (random images/video), 61% (FER-2013); effective for real-time recognition
8		This study (Proposed)	ResNet-50, Inception V3, Xception	FER-2013	Image Optimization through multiple strategies: i) input size testing, ii) hyperparameter tuning, iii) augmentation schemes, iv) additional architectural layers, and v) evaluation of three training methods

2. BACKGROUND STUDY

2.1. Emotion recognition

Facial emotions and expressions are crucial in non-verbal communication and serve as a natural way to convey human internal feelings in personal interactions [1]. Emotions arise from the movements of facial muscles and can communicate various signals in communication, from warnings to subtle cues. For instance, raising eyebrows or furrowing the brow during a conversation can convey messages without words [2], Such expressions help clarify emotions, intentions, or feelings and can strengthen or complement verbal communication. Studies in psychology indicate that about half of the information exchanged in conversations comes from displayed emotions [1]. For example, a conversation accompanied by a happy or sad face can significantly influence how the listener receives the message. Renowned psychologist Paul Ekman discovered that humans universally express emotions through seven similar facial expressions: happiness, sadness, anger, fear, surprise, disgust, and neutrality [3].

Emotion recognition is influenced by factors like lighting, pose, background, perspective, camera quality, and occlusion, where a face is partially obstructed by another object. The accuracy of emotion recognition is largely dependent on the processing capabilities of the visual recognition system, supported by how information is understood and processed [5]. Emotion recognition can be performed using facial image datasets, which undergo preprocessing and analysis through pattern recognition techniques and machine learning methods like computer vision, artificial intelligence, and deep learning [6]. Recognizing emotions is essential as it enhances the quality of interactions between humans and computers, applicable in various fields. In tourism, AI-based facial emotion recognition can help assess tourists' satisfaction or dissatisfaction in real-time by analyzing facial expressions, allowing for more accurate response adjustments, such as offering additional solutions when negative emotions are detected to enhance customer experience [20]. In healthcare, emotion recognition technology aids in monitoring patients' emotional conditions, especially those with facial paralysis, facilitating diagnosis and treatment adjustments [21]. In education, facial emotion recognition helps assess student engagement in real-time, enabling educators to adjust teaching methods and materials to improve effectiveness and academic results [22]. Overall, this technology enables computer systems to respond to user emotions more humanely.

2.2. Transfer learning

Transfer learning is a machine learning approach that enables the use of a model trained for one specific task as a starting point for addressing another related task. This method is especially advantageous when existing procedures that tackle the main issue are available, and the new task demands a large amount of data [23], It is frequently utilized across various model types, particularly CNNs in image recognition tasks. In transfer learning, the features from the pre-trained model are extracted, which means that training

can commence without starting from scratch. Typically, transfer learning models are trained on extensive datasets, and the parameters derived from these models can be utilized in specialized neural networks for other similar tasks. This allows the model to be directly employed for making predictions in new tasks or for training in related applications.

2.2.1. ResNet-50

ResNet-50 [24] is a popular CNN model designed to address network degradation in deep architectures. Its skip connections allow inputs to skip certain layers, helping to reduce vanishing gradients and overfitting. This design contributed to its success in winning the 2015 ImageNet challenge.

2.2.2. Xception

Xception [25] is a modern CNN model that combines concepts from GoogleNet and ResNet. Unlike those two models, Xception uses a separable convolution layer that separates spatial and cross-channel patterns. This reduces computational complexity, number of parameters, and memory usage while improving model performance.

2.2.3. Inception

Inception [26] or GoogleNet, was developed by Google to optimize parameter usage in networks. The model utilizes an efficient inception module, minimizing the number of parameters without sacrificing accuracy. Inception V3 is one of the most frequently used variants in facial emotion recognition.

3. METHOD

This study aims to optimize transfer learning for facial emotion classification on the FER-2013 dataset by utilizing three pre-trained models: ResNet-50, Inception V3, and Xception. The experiments were conducted directly on Kaggle notebooks with P100 GPU accelerator. The local machine used for preprocessing and minor computations had the following specifications: Intel(R) Core(TM) i5-8265U, 4 GB RAM and running in Windows 11. All models were implemented using TensorFlow with Keras API, and additional Python libraries included NumPy, Pandas and Scikit-Learn. The optimization process is carried out in several stages and applied to each model. The first stage involves testing three variations of input image sizes to determine the most suitable resolution for each model. The second stage focuses on tuning four key hyperparameters: learning rate, batch size, optimizer, and epochs, aiming to find the best combination for each architecture. The third stage applies data augmentation to address the imbalance of the dataset with two types of schemes, applied only to the minority class and applied to all classes. In the fourth stage, architectural modifications are made by adding dropout layers and a combination of dropout layers and batch normalization to improve model generalization and stability. The final stage tests three different training methods to evaluate their impact on model performance. The best results from each stage serve as the basis for the next stage, ensuring that the optimal combination of settings and architectures is achieved for each model.

All experiments were implemented using the TensorFlow and Keras frameworks in Python. The experiments were conducted using Kaggle Notebooks with the GPU P100 accelerator, supported by a personal laptop with an Intel(R) Core(TM) i5-8265U processor, 4 GB RAM, and Windows 11 operating system. Identical data splits and preprocessing steps were applied across all models to ensure fair comparison of performance. Figure 1 illustrates the overall research stages and experimental setup used in this study, providing a clear depiction of the workflow implemented for optimizing transfer learning on the FER-2013 dataset.

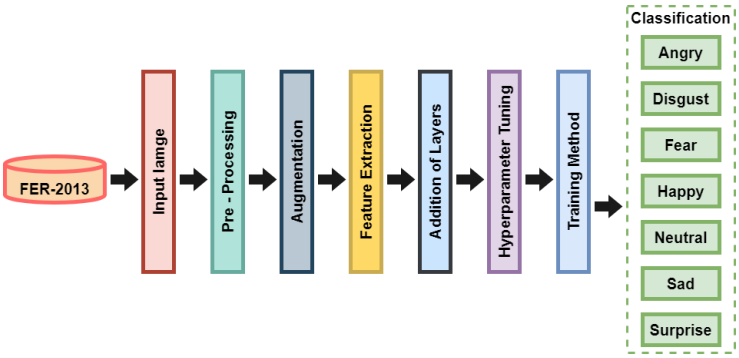

Figure 1. Experimental setup and research stages of proposed method

3.1. Dataset

FER-2013 is a dataset introduced at an international conference on machine learning in 2013 [27]. It can be accessed for free through the Kaggle website. The FER-2013 dataset consists of 35,887 images with grayscale mode and an image size of 48×48 pixels. The FER-2013 dataset is divided into training data (27,809 images) and testing data (7,178 images), with each set further categorized into seven emotion classes: angry, disgust, fear, happy, neutral, sad, and surprise. A sample of FER-2013 facial emotion images is shown in Figure 2 and the distribution of images in the training and testing data can be seen in Table 2.

Figure 2. Sample facial emotion images from the FER-2013

Table 2. Dataset description

Class	Training	Testing
Angry	3.995	958
Disgust	436	111
Fear	4.097	1.024
Happy	7.215	1.774
Neutral	4.965	1.233
Sad	4.830	1.247
Surprise	3.171	831
Total	27.809	7.178

3.2. Data pre-processing

The data pre-processing stage includes changing the image mode to RGB and resizing the image to 150×150 pixels, 200×200 pixels, and 224×224 pixels. This resizing aims to determine the most suitable input image size for each transfer learning model (ResNet-50, Inception V3, and Xception) to achieve the highest accuracy. All pixel values are normalized by scaling them to the range [0, 1] to standardize the input distribution. At this stage, testing experiments are carried out on the three transfer learning models to find the most suitable input image size for each transfer learning model so that it produces the highest accuracy value. This test was conducted using Adam optimizer, learning rate 0.001, batch size 64, and epoch 10 for each model. It is important to note that no data augmentation is applied at this stage, augmentation is implemented later in a separate experimental phase to assess its specific contribution to performance and class imbalance mitigation.

3.3. Feature extraction

Feature extraction in this study applies three transfer learning models that have proven to excel in the image recognition process: ResNet-50, Inception V3, and Xception. Each transfer learning model utilizes a deep convolutional architecture to extract visual patterns from facial images in the FER-2013 dataset. The feature extraction process involves using the initial layers of these models to obtain relevant representations, which are then used in the classification stage. The architectural structure of ResNet-50, Inception V3, and Xception used as the baseline models in this study is illustrated in Figure 3.

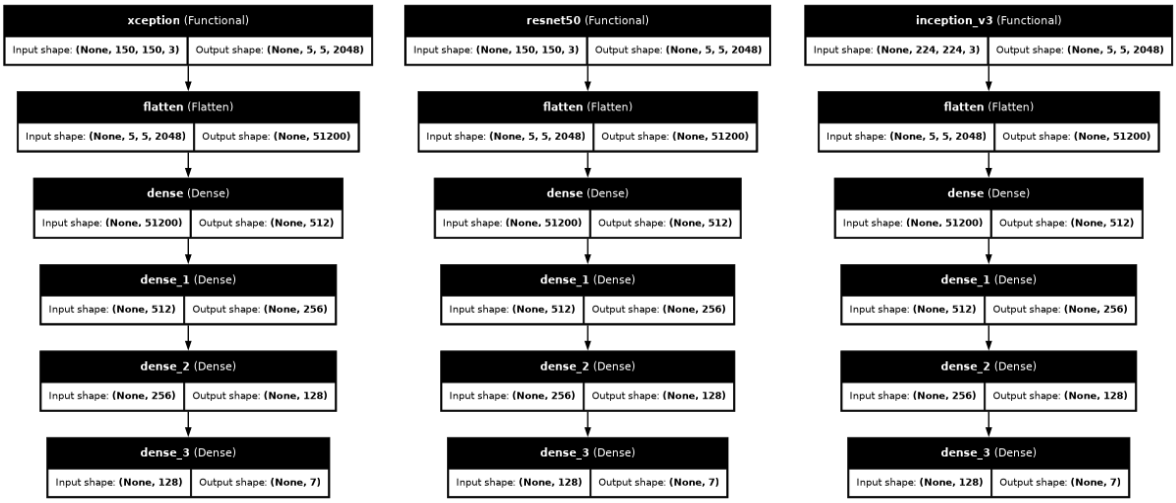

Figure 3. Transfer learning architecture

3.4. Hyperparameter tuning

At this stage, the hyperparameter tuning test scenario will be conducted using the input image that achieved the best accuracy in previous tests for each model. ResNet-50 and Xception will use an input image size of 150×150 pixels, while Inception v3 will use an input image size of 224×224 pixels. Hyperparameter tuning testing is conducted to identify the optimal combination of optimizer, batch size, learning rate, and epoch for the three models that provide the best performance characterized by the highest accuracy value. The optimizers tested are Adam, RMSProp, and SGD. The batch sizes to be tested are 32, 64, and 128. The learning rates to be evaluated are 0.01, 0.001, and 0.0001. The number of epochs to be tested are 10, 50, and 100. The default initial parameters used for all models are Adam optimizer, learning rate of 0.001, batch size of 64, and 10 epochs.

3.5. Augmentation

To address the issue of class imbalance, particularly the limited number of *disgust* class images, two augmentation strategies are evaluated in a separate experimental phase. The first involves targeted augmentation applied only to the *disgust* class using OpenCV transformations (e.g., rotation, zoom, flipping). The second approach applies augmentation to all classes using the Augmentor library to maintain balanced representation. These methods aim to enhance training diversity and assess which augmentation technique yields the best accuracy across different models. Oversampling or weighted loss functions are not used in this study; augmentation is the primary strategy for handling imbalance. This testing stage uses the optimal parameters obtained from the previously conducted hyperparameter tuning scenarios. Augmentation parameters for the disgust class alone, as well as for all classes, are shown in Table 3.

Table 3. Augmentation parameters only for disgust class and entire class

Augmentation	Library	Parameter
Only disgust class	OpenCV (cv2)	Rotate 90 Clockwise Rotate 90 Counterclockwise Rotate 180 Flip Horizontal Flip Horizontal + Rotate 90 Degrees Clockwise Flip Horizontal + Rotate 90 Degrees Counterclockwise Flip Vertical Flip Vertical + Rotate 90 Degrees Clockwise Flip Vertical + Rotate 90 Degrees Counterclockwise
Entire class (Including disgust class)	Augmentor	Rotate (Probability=0.5, Max_Left_Rotation=20, Max_Right_Rotation=20) Rotate_Random_90 (Probability=0.5) Width_Shift_Range = 0.2 Height_Shift_Range = 0.2 Flip_Left_Right(Probability=0.5) Flip_Top_Bottom(Probability=0.5) Fill_Mode = 'Nearest' Sample (7215)

3.6. Addition of layers

Dropout and batch normalization are crucial techniques in neural networks that enhance performance and training stability. Dropout reduces overfitting by randomly disabling some neurons during training, while batch normalization normalizes each mini-batch's input to accelerate training and mitigate vanishing gradient issues. Both techniques improve the generalization and efficiency of neural networks. Experiments with adding dropout and batch normalization layers aim to assess their impact on accuracy when augmenting the disgust class in each transfer learning model. The architecture of each model when adding dropout layers and batch normalization is presented in Figures 4 and 5.

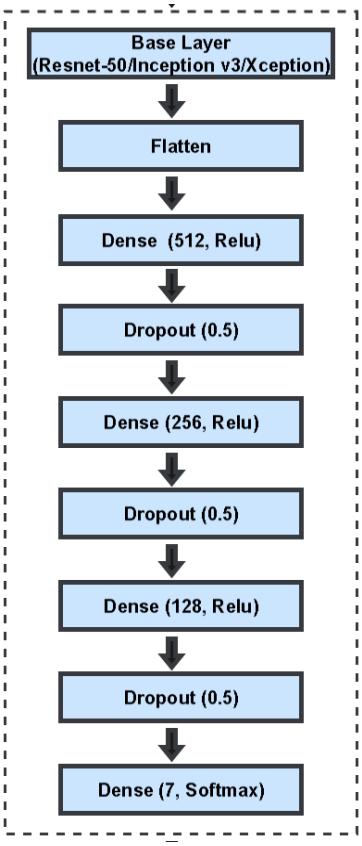

Figure 4. Dropout layer architecture

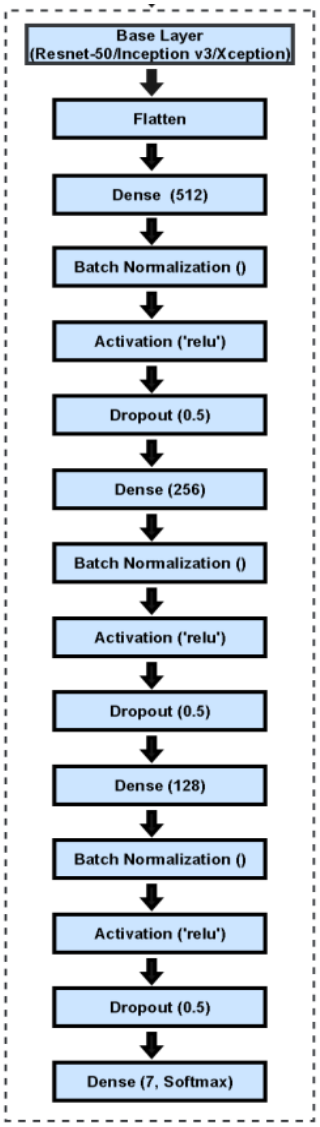

Figure 5. Dropout and batch normalization layer architecture

3.7. Training method

Based on several previous studies, common training methods in computer vision research include fine-tuning, freeze layer, and freeze+fine-tuning. Fine-tuning involves taking a pre-trained model and adapting it to a new, more specific dataset. This process updates the weights of some or all network layers, allowing the model to learn features that are more relevant to the new data.

Freeze layer is a strategy where certain layers in a neural network are kept frozen and not updated during the fine-tuning process. This means that the weights and parameters in these layers remain unchanged while only the other layers are allowed to adapt to new data. This technique helps prevent the waste of

computing resources and training time during the fine-tuning process. It is commonly used when applying transfer learning to extract features, as freezing the feature-extracting layers allows Fine-Tuning to focus on adapting more complex features relevant to new tasks.

Freeze+fine-tuning is a combination of the two previous strategies and represents a new method introduced by Yen and Li [1]. In the initial stage, some or all layers (base layers/pre-trained method) are frozen to prevent updates during several training iterations. Once these frozen layers stabilize or converge, they are then unfrozen, and the entire network is retrained (fine-tuned) to better adapt to the new dataset. This approach combines the advantages of transfer learning with specific adjustments to new data, enhancing the method's adaptability to changing conditions or data characteristics.

The training method test was conducted to determine which method performed best, as indicated by the highest accuracy value. Each method was tested without augmentation and with augmentation applied to the disgust class, incorporating dropout or batch normalization layers based on the best accuracy values obtained from previous tests.

3.8. Evaluation

The evaluation of the study was conducted by applying a multi-class confusion matrix to calculate the accuracy parameters for each transfer learning model. The evaluation used 7,178 facial emotion images as test data.

4. RESULTS AND DISCUSSION

4.1. Results

4.1.1. Data pre-processing

The results of the input image size testing show that the best accuracy for the ResNet-50 model is achieved at an image size of 150×150 pixels, with an accuracy value of 58.5400%. For the Inception V3 model, the best accuracy obtained is 61.3402% at an image size of 224×224 pixels. The Xception model demonstrates the highest accuracy at an image size of 150×150 pixels, reaching an accuracy of 64.8231%. Based on the image size testing results, the Xception model outperforms the other two models as it has the highest accuracy. The results of the data pre-processing testing can be seen in Table 4.

Table 4. Results of input image size experiments for each model

Model	Image size	Accuracy	Loss
ResNet-50	**150×150**	**58.5400%**	**1.4291**
	200×200	56.7150%	1.7919
	224×224	56.3945%	1.5979
Inception V3	150×150	59.0276%	1.1698
	200×200	60.4068%	1.2950
	224×224	**61.3402%**	**1.1147**
Xception	**150×150**	**64.8231%**	**1.3942**
	200×200	62.3433%	1.7233
	224×224	59.0137%	1.9313

4.1.2. Hyperparameter tuning

In hyperparameter tuning, four scenarios were conducted. In the first scenario, three types of optimizers: Adam, SGD, and RMSProp, were tested using the following fixed parameters: learning rate = 0.001, batch size = 64, and epochs 10. The test results showed that the ResNet-50 model experienced an increase in accuracy when using the RMSProp optimizer, while the highest accuracy for the other two models was achieved with the Adam optimizer. The lowest accuracy across all models was observed when using the SGD optimizer. In the second scenario, three learning rate values were tested: 0.01, 0.001, and 0.0001, with each model using the optimizer that produced the best accuracy in the first scenario. The remaining parameters were set as follows: batch size = 64 and epochs 10. The test results showed that ResNet-50 and Inception V3 achieved increased accuracy with a learning rate = 0.0001, while Xception reached the highest accuracy with a learning rate = 0.001. All models exhibited the lowest accuracy with a learning rate of 0.01.

In the third scenario, three batch sizes: 32, 64, and 128, were tested with each model using the best optimizer and learning rate from the previous test scenario, and the remaining parameter was set to epochs = 10. The test results showed that the ResNet-50 model achieved higher accuracy with a batch size of 128, while the highest accuracy for the other two models was obtained with a batch size of 64. The lowest accuracy for ResNet-50 and Xception was observed with a batch size of 32, while Inception V3 achieved its lowest accuracy with a batch size of 128. The test results indicate that Inception V3 and Xception achieved

increased accuracy with 50 epochs, whereas ResNet-50 showed improved accuracy with 100 epochs. In this scenario, all models experienced enhanced performance. The results of the hyperparameter tuning are presented in Table 5 to Table 8. The optimal hyperparameter tuning scenario for each model, which produced the best accuracy, is presented in Table 9.

Table 5. Optimizer test results

Model	Scenario 1: Optimizer		
	Optimizer	Accuracy	Loss
ResNet-50	Adam	58.5400%	1.4291
	SGD	58.3728%	1.5121
	RMSprop	**59.4455%**	**1.6206**
Inception V3	**Adam**	**61.3402%**	**1.1147**
	SGD	54.1655%	1.4524
	RMSprop	59.0694%	1.3616
Xception	**Adam**	**64.8231%**	**1.3942**
	SGD	53.3157%	1.2768
	RMSprop	60.2257%	1.6826

Table 6. Learning rate test results

Model	Scenario 2: Learning rate		
	Learning rate	Learning rate	Learning rate
ResNet-50	0.01	0.01	0.01
	0.001	0.001	0.001
	0.0001	**0.0001**	**0.0001**
Inception V3	**0.01**	**0.01**	**0.01**
	0.001	0.001	0.001
	0.0001	0.0001	0.0001
Xception	**0.01**	**0.01**	**0.01**
	0.001	0.001	0.001
	0.0001	0.0001	0.0001

Table 7. Batch size test results

Model	Scenario 3: Batch size		
	Batch size	Accuracy	Loss
ResNet-50	32	58.8047%	3.1573
	64	60.9083%	2.8245
	128	**64.3912%**	**2.9262**
Inception V3	32	64.3773%	1.6329
	64	**64.8091%**	**1.7979**
	128	62.2597%	1.8830
Xception	32	59.6963%	1.4543
	64	**64.8231%**	**1.3942**
	128	61.5492%	1.4642

Table 8. Epoch test results

Model	Scenario 4: Epoch		
	Epoch	Epoch	Epoch
ResNet-50	10	10	10
	50	50	50
	100	**100**	**100**
Inception V3	**10**	**10**	**10**
	50	50	50
	100	100	100
Xception	**10**	**10**	**10**
	50	50	50
	100	100	100

Table 9. Optimal hyperparameter tuning scenarios for each model

Model	Input shape	Optimizer	Learning rate	Batch size	Epoch
ResNet-50	150×150	RMSprop	0.0001	128	100
Inception V3	224×224	Adam	0.0001	64	50
Xception	150×150	Adam	0.001	64	50

4.1.3. Augmentation

The results of data augmentation across all transfer learning models indicate that augmenting the disgust class in the FER-2013 dataset yields better accuracy compared to applying augmentation across all classes in the dataset. The augmentation applied to the disgust class, which is the minority class in the FER-2013 dataset, enables the transfer learning model to better recognize and deeply learn the image patterns within that class. However, the application of data augmentation to the FER-2013 dataset using the ResNet-50, Inception V3, and Xception transfer learning models did not significantly impact the overall model performance. In the ResNet-50 and Xception models, the implementation of augmentation led to accuracy decreases of 1.5185% and 1.3235%, respectively. Meanwhile, in the Inception V3 model, augmentation slightly increased accuracy by 0.1533%. The results of augmentation for the disgust class alone and for all classes in the FER-2013 dataset are presented in Table 10.

Table 10. Augmentation testing results only for the disgust class and entire classes

Model	Augmentation	Accuracy	Loss
ResNet-50	None	64.8927%	3.1570
	Disgust	63.3742%	3.3166
	All Class	50.2647%	4.2476
Inception V3	None	66.0351%	1.9707
	Disgust	66.1884%	2.0119
	All Class	57.8434%	2.5671
Xception	None	65.8261%	2.4100
	Disgust	64.5026%	2.1018
	All Class	52.8002%	2.4890

4.1.4. Addition of layers

Based on the experimental results of adding dropout and batch normalization layers to the disgust class augmentation, it can be seen that for the ResNet-50 model, the highest accuracy of 63.3742% was achieved without adding any layers. In contrast, the Inception V3 model achieved the highest accuracy of 66.3973% when the dropout layer was included. In the Xception model, the highest accuracy of 65.4778% was achieved with the addition of both the dropout and batch normalization layers. For the ResNet-50 and Xception models, applying augmentation to the disgust class along with adding dropout or batch normalization layers did not improve accuracy compared to the models without augmentation. However, in the Inception V3 model, the inclusion of the dropout layer enhanced accuracy, surpassing the accuracy achieved without augmentation on the disgust class. The results of incorporating these layers into each model are presented in Table 11. A comparison of accuracy values with and without augmentation for the disgust class is shown in Table 12.

Table 11. Experimental results of adding dropout layers and batch normalization to each model

Model	Layer	Accuracy	Loss
ResNet-50	None	**63.3742%**	**3.3166**
	Layer Dropout	61.2566%	4.9117
	Layer Dropout + Batch Normalization	57.8434%	3.4950
Inception V3	None	66.1884%	2.0119
	Layer Dropout	**66.3973%**	**2.1880**
	Layer Dropout + Batch Normalization	66.1884%	1.8499
Xception	None	64.5026%	2.1018
	Layer Dropout	24.7144%	1.9056
	Layer Dropout + Batch Normalization	**65.4778%**	**2.1353**

Table 12. Comparison of accuracy values without applying augmentation, applying augmentation, and applying additional layers to each model

Model	Augmentation	Layer	Accuracy	Loss
ResNet-50	None	None	**64.8927%**	**3.1570**
	Disgust	None	63.3742%	3.3166
	Disgust	Dropout	61.2566%	4.9117
Inception V3	None	None	66.0351%	1.9707
	Disgust	None	66.1884%	2.0119
	Disgust	Dropout	**66.3973%**	**2.1880**
Xception	None	None	**65.8261%**	**2.4100**
	Disgust	None	64.5026%	2.1018
	Disgust	Dropout + Batch Normalization	65.4778%	2.1353

The implementation of dropout and batch normalization layers resulted in a significant decrease in accuracy across all models. The application of dropout alone on the Xception model led to a substantial accuracy drop of 41.1117%, while ResNet-50 experienced a decrease of 3.6361%. In contrast, Inception V3 showed a slight accuracy increase of 0.3622%, although this change was not significant. This indicates that the effects of using dropout and batch normalization layers can vary greatly depending on the transfer learning model employed. Adjustments based on deeper observations are necessary if these techniques are to be applied to transfer learning models.

4.1.5. Training method

Based on the test results, it can be concluded that all three transfer learning models achieved the best accuracy by applying the fine-tuning training method. For the ResNet-50 and Xception models, the highest accuracy was obtained without applying any augmentation or additional layers, with accuracy values of 64.8927% and 65.8261%, respectively. In the case of the Inception V3 model, the best accuracy was achieved by applying augmentation and adding a dropout layer, resulting in an accuracy of 66.3973%. The training method test results are presented in Table 13. The combination of augmentation and layers that produced the highest accuracy for each model can be seen in Table 14, while the combinations of augmentation and additional layers for each transfer learning model during training are displayed in Table 15.

Table 13. Training method test results

Model	Layer	Augmentation	Training method	Acc (%)	Loss
ResNet-50	None	None	Fine Tuning	**64.8927**	**3.1570**
			Frezee Layer	39.9415	1.5731
			Freeze + Fine Tuning	60.5461	3.3006
	None	Disgust	Fine Tuning	**63.3742**	**3.3166**
			Frezee Layer	39.8300	1.6070
			Freeze + Fine Tuning	61.3123	3.7348
	Dropout	Disgust	Fine Tuning	61.2566	4.9117
			Frezee Layer	18.1666	1.9184
			Freeze + Fine Tuning	**63.2488**	**6.1145**
Inception V3	None	None	Fine Tuning	**66.0351**	**1.9707**
			Frezee Layer	56.7707	3.5197
			Freeze + Fine Tuning	63.7643	1.9639
	None	Disgust	Fine Tuning	**66.1884**	**2.0119**
			Frezee Layer	55.7119	3.6822
			Freeze + Fine Tuning	64.6420	1.8421
	Dropout	Disgust	Fine Tuning	**66.3973**	**2.1880**
			Frezee Layer	42.7835	1.5189
			Freeze + Fine Tuning	65.4221	1.7153
Xception	None	None	Fine Tuning	**65.8261**	**2.4100**
			Frezee Layer	54.7228	3.3743
			Freeze + Fine Tuning	64.2240	1.6895
	None	Disgust	Fine Tuning	64.5026	2.1018
			Frezee Layer	52.5355	3.5379
			Freeze + Fine Tuning	**64.8091**	**1.6027**
	Dropout + Batch Normalization	Disgust	Fine Tuning	**65.4778**	**2.1353**
			Frezee Layer	57.2026	1.8044
			Freeze + Fine Tuning	64.2658	1.8135

Table 14. A combination that produces the best accuracy value for each model

Model	Metode training	Layer	Augmentation	Acc (%)	Loss
ResNet-50	Fine tuning	None	None	64.8927	3.1570
Inception V3	Fine tuning	Dropout	Disgust	66.3973	2.1880
Xception	Fine tuning	None	None	65.8261	2.4100

Table 15. Combination of augmentation and additional layers for each model in the training method

Training method	Layer	Augmentation	Model	Acc (%)	Loss
Fine tuning	None	None	ResNet-50	64.8927	3.1570
	Dropout	Disgust	Inception V3	66.3973	2.1880
	None	None	Xception	65.8261	2.4100
Freeze layer	None	None	ResNet-50	39.9415	1.5731
	None	None	Inception V3	56.7707	3.5197
	Dropout + Batch Normalization	Disgust	Xception	57.2026	1.8044
Freeze + Fine tuning	Dropout	Disgust	ResNet-50	63.2488	6.1145
	Dropout	Disgust	Inception V3	65.4221	1.7153
	None	Disgust	Xception	64.8091	1.6027

4.2. Discussion

The results of this research were obtained from a series of tests such as testing input image size, hyperparameter tuning, data augmentation, adding dropout layers, and batch normalization to the transfer learning model architecture and determining the appropriate model training method. This test provides some information regarding the performance of the transfer learning model in various conditions so that it can help identify the best combination of methods to achieve optimal accuracy on the FER-2013 dataset.

The results of testing different input image sizes show differences in the accuracy values obtained in each transfer learning model. Choosing an appropriate input image size can help the transfer learning model effectively capture and analyze image information, thereby enhancing its ability to recognize and classify images accurately. It is important to note that the optimal input image size is not necessarily the one with the highest resolution. This research demonstrates that ResNet-50 and Xception achieve their best accuracy with the smallest input image size of 150×150 pixels, while Inception V3 performs best with the largest input image size of 224×224 pixels.

The hyperparameter tuning test results show that selecting the optimizer, learning rate, batch size, and epoch for each transfer learning model is very important to improve the learning ability and accuracy of the model. Selecting the right hyperparameter tuning can help increase the efficiency and accuracy of the

model in recognizing data that has never been encountered or seen before. This can be proven from the results of hyperparameter tuning testing on ResNet-50, Inception V3, and Xception which increased model accuracy by 6.3527%, 4.6949%, and 1.039% respectively.

The results of data augmentation in all transfer learning models show that data augmentation in the disgusting class in the FER-2013 dataset has better accuracy when compared to augmenting all classes in the FER-2013 dataset. The augmentation applied to the disgust class alone, which is a minority class in the FER-2013 dataset, allows the transfer learning model to better recognize and study in depth the image patterns in the disgust class. Applying data augmentation to the FER-2013 dataset using the ResNet-50, Inception V3, and Xception transfer learning models did not have a significant impact on overall model performance. In the ResNet-50 and Xception models, applying augmentation causes a decrease in accuracy of 1.5185% and 1.3235%. Meanwhile, in the Inception V3 model, applying augmentation can increase accuracy, although only by 0.1533%.

The addition of dropout and batch normalization layers in the transfer learning model architecture and the application of augmentation to the disgust class were carried out to determine the potential for increasing accuracy in the transfer learning model. The results show that applying layer dropout and batch normalization shows a very significant decrease in accuracy for all models. Applying the dropout layer alone results in a significant accuracy decrease in Xception, dropping to 41.1117%. In ResNet-50, it leads to a 3.6361% decrease in accuracy. However, in Inception V3, it causes a slight increase in accuracy of 0.3622%. This indicates that the impact of using dropout and batch normalization layers can vary greatly depending on the transfer learning model employed. Adjustments need to be made with more in-depth observations if you want to apply them to the transfer learning model.

Testing several training methods, including fine-tuning, freeze fine-tuning, and freeze layers, revealed that fine-tuning provided the best performance. It achieved the highest accuracy values across all transfer learning models used. The freeze-fine tuning method [1] turns out to have lower performance when compared with the fine-tuning method but turns out to be better when compared with the freeze-layer method. This is evident from the significant increase in accuracy values for all transfer learning models that apply the freeze fine-tuning method when compared with the application of the freeze layer method.

The results of this research reveal significant differences compared to Yen's research [1], which shows that the freeze-fine-tuning method performs better than the fine-tuning method. Yen's research [1] used the source dataset ImageNet (14,197,122 images) and AffectNet (0.4 million images) with the target dataset FER-2013 (35,887 images), whereas this study only used the FER-2013 dataset which was divided into train data (28,709 images) and test data (7,178 images). The differences in the results of this study can be caused by differences in the number of datasets used in the training process. Freeze-fine tuning does have better performance than fine-tuning on very large training datasets with millions of images. Meanwhile, on a small training dataset that only has thousands of images, fine-tuning performance is proven to be superior when compared to freeze fine-tuning performance on all transfer learning models used. A comparison of the accuracy values obtained in this research with previous research can be seen in Table 16.

Table 16. Comparison of results of this study with those of previous studies

Technique/Model	Source dataset	Target dataset	Accuracy
CNN [28]	FER-2013	FER-2013	65,97%
ResNet-50 (This Study)	**FER-2013**	**FER-2013**	**64.89 %**
ResNet-50 [1]	**ImageNet**	**FER-2013**	**70%**
Improved ResNet-50 [29]	FER-2013	FER-2013	57.31%
ResNet-50 [30]	FER-2013	FER-2013	62,55 %
Inception V3 (This Study)	**FER-2013**	**FER-2013**	**66.40 %**
Inception V3 [1]	**ImageNet**	**FER-2013**	**65 %**
Inception V3 [31]	FER-2013	FER-2013	63,9 %
Inception V3 [32]	FER-2013	FER-2013	63,21 %
Xception (This Study)	**FER-2013**	**FER-2013**	**65.83 %**
Xception [1]	**ImageNet**	**FER-2013**	**67%**
Xception [30]	FER-2013	FER-2013	60,35%

5. CONCLUSION

This study compares the performance of transfer learning models ResNet-50, Inception V3, and Xception in recognizing facial emotions using the FER-2013 dataset. Initially, Xception outperformed both ResNet-50 and Inception V3. However, after implementing various adjustments, Inception V3 exhibited superior performance compared to the other two models.

The techniques applied during data preprocessing, hyperparameter tuning, data augmentation, layer addition, and different training methods had varied impacts on model accuracy. Input image size significantly influenced transfer learning performance; larger images did not always lead to higher accuracy, with some models performing better with smaller images. Hyperparameter tuning was crucial for identifying optimal parameters that could enhance the accuracy of each model significantly. Data augmentation had a minimal effect, as indicated by slight changes in accuracy across models. The addition of layers, such as dropout and batch normalization, showed varying impacts, often resulting in decreased accuracy. Among the training methods tested, fine-tuning proved to be the most effective in improving accuracy compared to freeze-fine tuning and freezing layers.

The evaluation results indicate that the fine-tuning training method achieved the highest accuracy on the FER-2013 dataset. This finding contrasts with Yen's study, which claimed that freeze-fine tuning was superior for large datasets like ImageNet and AffectNet. In contrast, fine-tuning was more effective for the smaller FER-2013 dataset.

The differences in accuracy may stem from the dataset sizes used in the studies, with freeze-fine tuning being more effective for very large datasets, while fine-tuning yielded better results on smaller datasets like FER-2013. This highlights the importance of selecting training methods based on the size and characteristics of the dataset to achieve optimal performance.

FUNDING INFORMATION

Authors state no funding involved.

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] C. T. Yen and K. H. Li, "Discussions of different deep transfer learning models for emotion recognitions," *IEEE Access*, vol. 10, no. August, pp. 102860–102875, 2022, doi: 10.1109/ACCESS.2022.3209813.

[2] K. Kaulard, D. W. Cunningham, H. H. Bülthoff, and C. Wallraven, "The MPI facial expression database - a validated database of emotional and conversational facial expressions," *PLoS ONE*, vol. 7, no. 3, 2012, doi: 10.1371/journal.pone.0032321.

[3] P. Ekman, "Basic emotions," in *Handbook of Cognition and Emotion*, Chichester, UK: John Wiley & Sons, Ltd, 2005, pp. 45–60.

[4] E. Vidyaningrum and Prihandoko, "Human face detection by using eigenface method for various pose of human face," *Undergraduate program, Faculty of Industrial Technology*, pp. 1–16, 2009.

[5] Y. Du, F. Zhang, Y. Wang, T. Bi, and J. Qiu, "Perceptual learning of facial expressions," *Vision Research*, vol. 128, pp. 19–29, 2016, doi: 10.1016/j.visres.2016.08.005.

[6] M. Sajjad *et al.*, "A comprehensive survey on deep facial expression recognition: challenges, applications, and future guidelines," *Alexandria Engineering Journal*, vol. 68, pp. 817–840, 2023, doi: 10.1016/j.aej.2023.01.017.

[7] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN Computer Science*, vol. 2, no. 3, May 2021, doi: 10.1007/s42979-021-00592-x.

[8] M. Puttagunta and S. Ravi, "Medical image analysis based on deep learning approach," *Multimedia Tools and Applications*, vol. 80, no. 16, pp. 24365–24398, 2021, doi: 10.1007/s11042-021-10707-4.

[9] M. Iman, H. R. Arabnia, and K. Rasheed, "A review of deep transfer learning and recent advancements," *Technologies*, vol. 11, no. 2, pp. 1–14, 2023, doi: 10.3390/technologies11020040.

[10] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with Gabor wavelets," in *Proceeding third IEEE International Conference on Automatic Face and Gesture Recognition*, 1998, pp. 200–205, doi: 10.1109/AFGR.1998.670949.

[11] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, Jun. 2010, pp. 94–101, doi: 10.1109/CVPRW.2010.5543262.

[12] C. Viegas, "Two stage emotion recognition using frame-level and video-level features," in *Proceedings - 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2020*, 2020, pp. 912–915, doi: 10.1109/FG47880.2020.00143.

[13] Y. S. Kong, V. Suresh, J. Soh, and D. C. Ong, "A systematic evaluation of domain adaptation in facial expression recognition," *arXiv preprint arXiv:2106.15453*, 2021.

[14] J. C. Hung, K.-C. Lin, and N.-X. Lai, "Recognizing learning emotion based on convolutional neural networks and transfer learning," *Applied Soft Computing*, vol. 84, Nov. 2019, doi: 10.1016/j.asoc.2019.105724.

[15] I. Oztel, G. Yolcu, and C. Oz, "Performance comparison of transfer learning and training from scratch approaches for deep facial expression recognition," in *UBMK 2019 - Proceedings, 4th International Conference on Computer Science and Engineering*, 2019, pp. 290–295, doi: 10.1109/UBMK.2019.8907203.

[16] E. S. Agung, A. P. Rifai, and T. Wijayanto, "Image-based facial emotion recognition using convolutional neural network on

emognition dataset," *Scientific Reports*, vol. 14, no. 1, p. 14429, 2024, doi: 10.1038/s41598-024-65276-x.
[17] L. Liao, S. Wu, C. Song, and J. Fu, "RS-Xception: a lightweight network for facial expression recognition," *Electronics (Switzerland)*, vol. 13, no. 16, 2024, doi: 10.3390/electronics13163217.
[18] T. Shahzad, K. Iqbal, M. A. Khan, Imran, and N. Iqbal, "Role of zoning in facial expression using deep learning," *IEEE Access*, vol. 11, no. January, pp. 16493–16508, 2023, doi: 10.1109/ACCESS.2023.3243850.
[19] N. N. Urnisha, S. I. Bithi, M. M. S. Rafee, N. I. Remon, M. M. Hasan, and R. R. Chowdhury, "A transfer learning approach for facial emotion recognition using a deep learning model," *International Journal of Research and Scientific Innovation*, vol. XI, no. IV, pp. 274–284, 2024, doi: 10.51244/ijrsi.2024.1104022.
[20] M. R. González-Rodríguez, M. C. Díaz-Fernández, and C. Pacheco Gómez, "Facial-expression recognition: An emergent approach to the measurement of tourist satisfaction through emotions," *Telematics and Informatics*, vol. 51, p. 101404, 2020, doi: 10.1016/j.tele.2020.101404.
[21] C. Xu *et al.*, "A novel facial emotion recognition method for stress inference of facial nerve paralysis patients," *Expert Systems with Applications*, vol. 197, p. 116705, 2022, doi: 10.1016/j.eswa.2022.116705.
[22] Mangaras Yanu Florestiyanto, "Emotion recognition for improving online learning environments: a systematic review of the literature," *Journal of Electrical Systems*, vol. 20, no. 4s, pp. 1860–1873, 2024, doi: 10.52783/jes.2255.
[23] S. Sarraf and G. Tofighi, "Classification of Alzheimer's disease using fMRI data and deep learning convolutional neural networks," *CoRR*, vol. abs/1603.0, no. May, pp. 8–12, 2016.
[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.
[25] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *Prepr. arXiv.1610.02357*, Oct. 2016.
[26] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.
[27] I. J. Goodfellow *et al.*, "Challenges in representation learning: A report on three machine learning contests," *Neural Networks*, vol. 64, pp. 59–63, 2015, doi: 10.1016/j.neunet.2014.09.005.
[28] L. Zahara, P. Musa, E. Prasetyo Wibowo, I. Karim, and S. Bahri Musa, "The facial emotion recognition (FER-2013) dataset for prediction system of micro-expressions face using the convolutional neural network (CNN) algorithm based Raspberry Pi," in *2020 5th International Conference on Informatics and Computing, ICIC 2020*, 2020, no. March 2021, doi: 10.1109/ICIC50835.2020.9288560.
[29] W. Du, "Facial emotion recognition based on improved ResNet," *Applied and Computational Engineering*, vol. 21, no. 1, pp. 242–248, 2023, doi: 10.54254/2755-2721/21/20231152.
[30] M. Rao, R. Bao, and L. Dong, *Face emotion recognization using dataset augmentation based on neural network*, vol. 1, no. 1. Association for Computing Machinery, 2022.
[31] E. G. Moung, C. C. Wooi, M. M. Sufian, C. Kim On, and J. A. Dargham, "Ensemble-based face expression recognition approach for image sentiment analysis," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 3, pp. 2588–2600, Jun. 2022, doi: 10.11591/ijece.v12i3.pp2588-2600.
[32] E. G. Dada, D. O. Oyewola, S. B. Joseph, O. Emebo, and O. O. Oluwagbemi, "Facial emotion recognition and classification using the convolutional neural network-10 (CNN-10)," *Applied Computational Intelligence and Soft Computing*, vol. 2023, 2023, doi: 10.1155/2023/2457898.

BIOGRAPHIES OF AUTHORS

Nida Muhliya Barkah received the bachelor's degree in informatics from Mulawarman University, Samarinda, Indonesia, in 2022 and the master's degree in informatics from Universitas Islam Negeri Sunan Kalijaga, Yogyakarta, Indonesia, in 2025. Her research interests include the areas of image recognition, machine learning, and deep learning. She can be contacted at email: nidamuhliya35@gmail.com.

Shofwatul 'Uyun is a full-time lecturer in the Department of Informatics, Universitas Islam Negeri (UIN) Sunan Kalijaga Yogyakarta, Indonesia. She obtained her bachelor's degree in informatics from Universitas Islam Indonesia. She received her M.Kom. and Dr. in computer science from Universitas Gadjah Mada. She was honored with a professorship in intelligent systems in 2023. Her research interests are pattern recognition, computer vision, artificial intelligence, and medical image processing. She can be contacted at email: shofwatul.uyun@uin-suka.ac.id.