

## Transfer Learning Technique with EfficientNet for Facial Expression Recognition System



Islam Nur Alam<sup>1\*</sup>, Iman H. Kartowisastro<sup>2,3</sup>, Pandu Wicaksono<sup>1</sup>

<sup>1</sup> Computer Science Departement, BINUS Graduate Program-Master of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia

<sup>2</sup> Computer Science Departement, BINUS Graduate Program-Doctor of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia

<sup>3</sup> Computer Engineering Departement, Faculty of Engineering, Bina Nusantara University, Jakarta 11480, Indonesia

Corresponding Author Email: [islam.alam@binus.ac.id](mailto:islam.alam@binus.ac.id)

<https://doi.org/10.18280/ria.360405>

### ABSTRACT

**Received:** 14 June 2022

**Accepted:** 12 August 2022

#### **Keywords:**

*convolutional neural networks, EfficientNet, facial expression recognition, transfer learning*

Facial Expression Recognition (FER) systems are helpful in a wide range of industries, including healthcare, social marketing, targeted advertising, the music industry, school counseling systems, and detection in the police sector. In this research, using Deep Convolutional Neural Networks (DCNN) architecture, specifically from the EfficientNet family (EfficientNet-B0, Efficient-Net-B01, EfficientNet-B02, EfficientNet-B03, EfficientNet-B04, EfficientNet-B05, EfficientNet-B06, and EfficientNet-B07) has previously gone through a combined scaling process of combined dept, width and resolution. First, the previously frozen sublayer EfficientNet model was used for feature extraction. Next, the layer closer to the output layer is melted by several layers to be retrained in order to recognize the pattern of the CK+ and JAFFE data sets. This process is called the transfer learning technique. This technique is very powerful for working on relatively small data sets, namely CK+ and JAFFE. The main of this research is to improve the accuracy and performance of facial expression recognition models with a transfer learning approach using EfficientNet pre-trained with fine-tuning strategy. Our proposed method, specifically using the EfficientNet-B0 architecture, achieves superior performance for each of the CK+ and JAFFE datasets, achieving 99.57% and 100% accuracy in the test set respectively.

## 1. INTRODUCTION

Facial expressions show a person's expressional state, which through this expression, can be decision support in action against someone [1]. Facial expression is a natural state humans feel because of an effort they experience. Furthermore, a person's expression is universal because of differences in the century. Ekman and Friesen confirmed Darwin's theory and classified six facial expressions in general, namely: happy, scared, surprised, disgusted, sad, and angry [2, 3].

Many challenges are faced in modeling a facial expression recognition system or FER. Facial expression recognition requires a fairly high-resolution image. The differences in the faces of each human and the expressions of a person are so difficult to distinguish that it complicates the task of classification [4]. In this case, training the CNN algorithm very deep by adding a lot of excess convolution layers will cause the model to not generalize well.

The vanishing gradient problem causes Extraction By adding more layers to the feature layer, and the accuracy cannot be raised continuously past a certain point. The most widely used pre-trained deep CNN models are VGG-16 [5], Resnet-50, Resnet-152 [6], Inception-v3 [7] and DenseNet 161 [8]. However, training deep CNN architectural models calls for a lot of power and powerful computing.

To solve the above-mentioned problems, the Google Brain team came up with a solution called EfficientNet which

improves model accuracy and computational requirements by scaling efficiently in all directions, such as not only depth but also width and resolution. Ideally, lead to optimal balance for each dimension relative to the others. In this way, EfficientNet does not require as much computational requirements as previous CNN models, resulting in better accuracy [9].

In this paper, the researcher proposes a facial expression recognition system using the EfficientNet model and then performs a transfer learning (TL) technique to reduce computational effort so that it is more efficient. The TL technique is a well-liked methodology for generating models quickly where learning begins from observed patterns [10]. The Facial Expression Recognition (FER) system proposed in this study is the EfficientNet pre-trained model [11], Originally designed for image classification, it was adopted by replacing its top layer with a solid layer to make it compatible with eight classes of CK+ data facial expressions and seven classes of JAFFE dataset face expressions.

The contribution of this research is to create a facial expression recognition model using a two-phase transfer learning approach. In this case, the researcher calls it representation transfer learning (RTF). Where the researcher uses the EfficientNet architectural model as feature extraction, then in this phase, the researcher will train the first model using 50 literacy and a learning rate of 0.001. In this phase, the EfficientNet architecture is in an unfrozen state. When the learning model is in the first training phase, the model will

store the seven or information that has been learned from ImageNet. After that, it was continued. Conducting the second training phase by unfreezing six layers of EfficientNet architecture, which were close to the output layer, then training it with a small learning rate value of 0.0001, then proceeding to add 50 iterations. In this case, the researcher will analyze eight different architectures in EfficientNet, namely (EfficientNet-B0, EfficientNet-B01, EfficientNet-B02, EfficientNet-B03, EfficientNet-B04, EfficientNet-B05, EfficientNet-B06, and EfficientNet-B07) in the collection CK+ and JAFFE public dataset.

## 2. RELATED WORK

Several techniques have recently been used to investigate the facial expression recognition system. Algorithm convolutional neural networks (CNN) as a feature extraction process from the image then enter the classifier process using a neural network. Therefore, recent research using the deep learning method for the task of recognizing facial expressions by using a combination of the two processes above. Several studies have been reviewed and compared with existing FER system methods [10-13], From those based on Deep learning methods among them [12, 14]. The following is a brief explanation of some of the well-known techniques used in the method of making the Facial Expression Recognition (FER) system.

### 2.1 Machine learning-based FER approach

Automatically making a Facial Expression Recognition (FER) system is a challenge in research, especially in machine learning. Several traditional machine learning algorithms, such as K-nearest neighbor and neural networks, have been used to test making the FER model

Xu and Wei [14] pioneered the FER method, which first added a wavelet energy feature (WEF) to the face image, then used the Fisher's linear discriminants (FLD) method for feature extraction, then continued with the expression classification process using the K-nearest neighbor (KNN) method. The KNN method was also used by Zhao et al. [15] in order to categorize facial expression recognition. However, they employed non-negative matrix factorization (NMF) and principal component analysis (PCA) for the feature extraction procedure.

Feng et al. [16] employed linear programming (LP) methods to categorize face expressions, local binary pattern (LBP) histogram extraction from various little portions of the image, and ultimately, feature histogram combination. Zhi and Ruan [17] facial feature vectors derived from 2D discriminant locality preservation projections. Lee et al. [18] employed a boosting method to classify the data and extend the wavelet transform to 2D, which it is known as the contourlet transform (CT), for the feature extraction procedure from the picture. Chang and Huang [19] integrated face recognition into the FER system to improve each person's skill at expression recognition. For classification, they used a neural network and a radial basis function (RBF).

One of the methods used for classifying facial expressions is the Support vector machine (SVM) from images that have undergone a feature extraction process using distinct techniques. In this category, Shih et al. [20] investigate several feature representations (such DWT and PCA), and it has been

found that DWT combined with 2D-linear discriminant analysis (LDA) performs better than other methods.

In this comprehensive investigation, they assessed various facial representations based on local statistical characteristics and LBP with various SVM variations [21]. Jabid et al. [22] investigated the local directional pattern (LDP), an appearance-based feature extraction method. Alshami et al. [23] SVM method was used to study two feature descriptors, facial landmarks descriptor and center of gravity descriptor. SVM and numerous other methods (e.g., KNN, LDA) were taken into account in the comparative analysis conducted by Liew and Yairi [11] for the classification of features extracted using various methods, including Gabor, Haar, and LBP. A recent investigation by Joseph and Geetha [24] based on their proposed facial geometry investigated various classification algorithms, including logistic regression LDA, KNN, classification and regression trees, naive Bayes, and SVM. The main limitation of the conventional methods mentioned above is that they only consider the frontal view for FER due to its features.

### 2.2 Deep learning-based FER approach

Creating facial expression recognition systems with deep learning approaches is relatively new in the machine learning [25] field. Until now, research on Convolutional Neural Networks (CNN) has been published in specialized journals. Zhao and Zhang used a deep belief network (DBN) algorithm and Neural Network for FER, where DBN was used for the unsupervised feature learning process. Then a neural network was used for facial expression classification. Pranav et al. [26] investigated FER on self-collected facial expressional images using a standard CNN architecture with two convolutional-pooling layers.

A larger architecture with four inception layers and two convolutional-pooling layers was investigated [27]. Pons and Masip [28] use a 72-CNN ensemble, with each CNN trained with a different filter size in the convolution layer or a different number of perceptions in each fully connected layer. Wen et al. [29] also use CNN ensembles, but they train 100 CNN and only use a subset of them in the final model. Ruiz-Garcia et al. [30] CNN sevens are trained with face images and initialized with encoder seven of the convoluted auto-encoder. This type of CNN initialization outperformed CNN with random initialization. FaceNet2ExpNet is a CNN architecture that extends the facial recognition architecture into FER [31]. Furthermore, Li et al. used transfer learning to create the FaceNet2ExpNet architecture. Jain et al. [32] consider the FER system's hybrid deep learning architecture, which includes the CNN algorithm and a recurrent neural network (RNN).

Shaees et al. used a hybrid architecture with a transfer learning approach [26], where SVM was used to identify the features of the AlexNet pre-trained architecture. Recently, CNN is used for the FER system, and DWT is used for feature extraction [33]. For the FER system, a relatively deep CNN architecture with 18 convolutional layers and four subsampling layers is used [34]. Recently, FER using a clustering approach with CNN [35]. Ngoc et al. [36] graph-based CNN was investigated for FER of natural facial features. Jin et al. [37] in their CNN-based method, they used both unlabeled and labeled data. In addition, Porcu et al. [38] used different data augmentation techniques, including synthetic images to train deep CNN architectures and combined

synthetic images with other methods to perform the FER system better. Existing deep learning-based methods have also taken frontal images into account, and most studies have even excluded profile view images from the data set in experiments to simplify the task [11, 23, 30, 39].

### 3. MATERIAL AND METHODOLOGY

In the implementation of this system, each image from the CK+ and JAFFE datasets, when entering each EfficientNet image, will be resized to a size of 224x224. Why should the size with 224x224? Because The images contained in the data frame will be uniformized spatial size to 224x224 by utilizing the OpenCV library. Image uniformization is done to adjust the input shape of the EfficientNet architecture and is expected to get good accuracy and performance [9, 40].

Figure 1 shows how the general architecture of the DCNN model works, especially EfficientNet based on transfer learning for the FER system, where the convolution base is part of the pre-trained DCNN that does not include its own classifier, and the classifier on the base is a newly added layer in the form of a neural network for FER.

Overall, the reuse of a pre-trained DCNN EfficientNet consists of two steps: replacing the original classifier with a new one and refining the model. The classifier part added is generally a solid layer combination from a fully connected layer. In practice, choosing a pre-trained model and determining a size-equality matrix for fine-tuning are critical steps in transfer learning [38, 41].

In the system implementation stage, researchers will design and manufacture the system illustrated in the flow diagram in Figure 2. In the experiment, the researcher conducted an experiment using the CNN EfficientNet [9] model, which was densely packed at the bottom layer, working as a feature extractor from the inserted image. This dense layer contains many layers of convolution processes in each layer that function as dimension reduction without reducing important information from the image. After the image has gone through the feature extractor process, fine-tuning is performed where the top layer of EfficientNet is melted so that it matches the characteristics of the labels for the seven facial expressions [3]. This process is called feature classification. So that this process strengthens and improves several CNN architectures in deep learning. In this case, transfer learning techniques are expected to reduce latency and computing resources [41].



Figure 1. General transfer learning-based efficientnet for facial expressions recognition

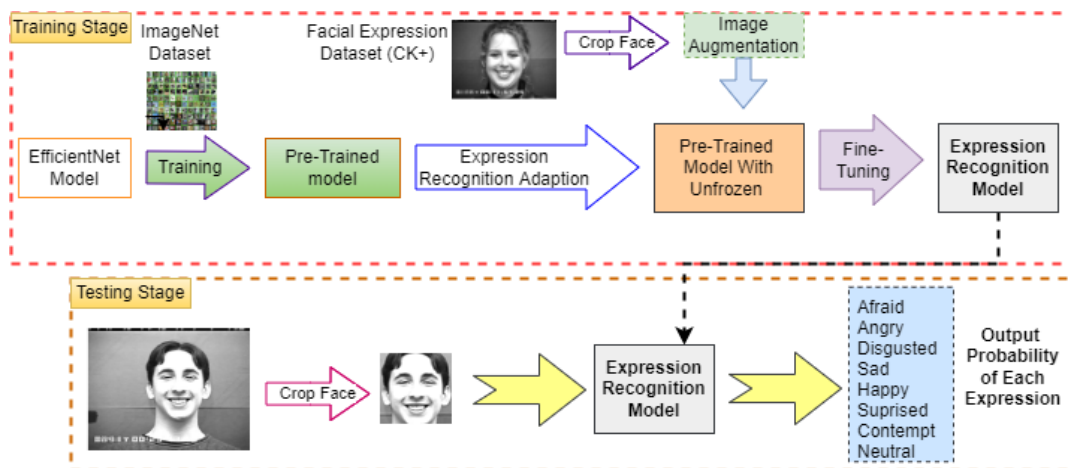


Figure 2. Explanation of the proposed FER system in DCNN with EfficientNet based on transfer learning

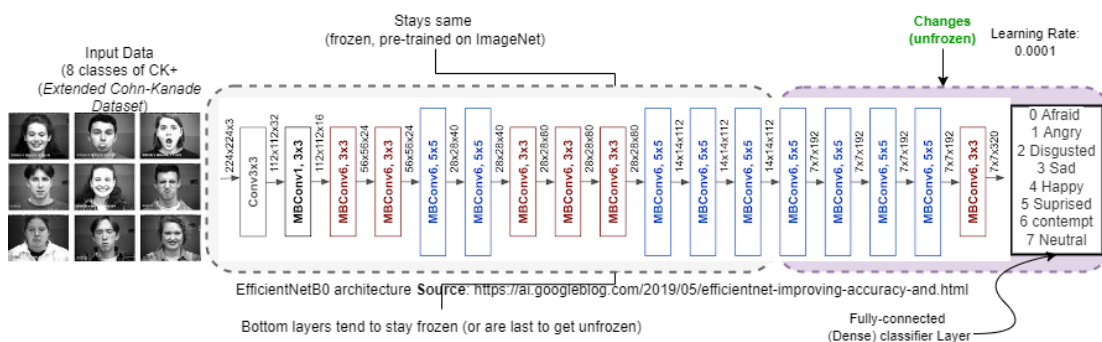


Figure 3. Demonstration of the proposed Expression Recognition model using the EfficientNet model and 8 dense neural networks (i.e., fully connected layers)

In Figure 3, implementing this system using the EfficientNet architecture, a good baseline model is needed to further build a model on it that has better performance. EfficientNet's base model of EfficientNet-B0 is built using reverse bottleneck convolution (MBConv) like MobileNetV2 and MnasNet. The MBConv block is nothing but an Inverted Residual block which was originally proposed in the CNN MobineNetV2 architecture. The reason behind using the Inverted Residual block is that, in the original residual block, the expansion layers in between are mere implementation details. The information can still be linked at low dimensions so that the computational requirements and running time are less [9]. EfficientNet opens a broad research arena to improve the outcomes of several computer vision tasks [9].

#### 4. EXPERIMENTAL STUDIES

This section investigates the efficiency of the proposed FER system using transfer learning on the CK+ and JAFFE datasets. First, a description of the CK+ datasets and then the experimental setup is presented. Finally, the results of the proposed model on the CK+ datasets were compared with several existing methods to verify the effectiveness of the proposed FER model.

JAFFE datasets are processed in the same manner. The section investigates the efficacy of the proposed FER system on CK+ datasets using transfer learning. Following a description of the CK+ datasets, the experimental settings, such as image preprocessing techniques with image augmentation, are presented. Finally, the proposed model's results on the CK+ datasets were compared to several existing

methods to validate the proposed FER model's effectiveness [40].

#### 4.1 Benchmark datasets

Several well-known public datasets are available for use in the FER system, such as (e.g., FER2013, JAFFE, CASIA-Web Face, and IMED). Specifically, in this paper, the researcher uses the CK+ (Extended Cohn-Kanade Dataset) datasets [42]. The CK+ datasets images are divided into eight expression classes: afraid, angry, disgusted, sad, happy, surprised, neutral, and contempt. The CK+ datasets are described briefly below, along with the reasons for their selection.

Figure 4 shows 327 image sequences with labeled facial expressions from the CK+ database. However, only the last frame of each image sequence is labeled with an expression. Consequently, to collect more images for training, in this study, we selected the last three frames of each sequence for training or validation. Next, the first frame of each of the 327 labeled sequences will be selected as the neutral expression. As a result, 1308 images with eight labeled facial expressions can be obtained from this dataset.

Figure 5 JAFFE (or JAFFE for short) dataset contains expression images of Japanese women from Kyushu University's Psychology Department. The JAFFE dataset was collected in a controlled environment to generate frontal facial expressions. Furthermore, JAFFE contains seven facial expression variants locally. The JAFFE dataset is small, with only 213 frontal images of 10 individuals; Figure 5 shows some examples. This dataset was selected to investigate how a small dataset responds to model training. Furthermore, many studies evaluate the FER model using the JAFFE dataset (e.g., [17, 21, 22]).



Figure 4. Sample images from CK+ dataset

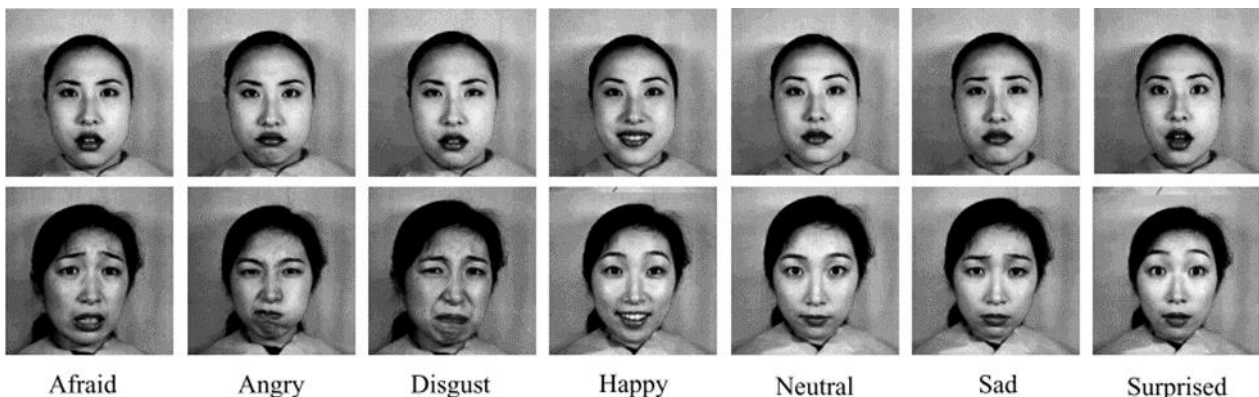


Figure 5. Sample images from JAFFE dataset

In this paper, the researcher divides the CK+ and JAFFE datasets into two parts, respectively: 90% for training and 10% for testing. Each section is placed into two different folders. Training data is used in the training process for building a model. Testing data is used for the testing process after building the FER model.

### 4.2 Preprocessing

In this case, preprocessing is done so that the image can be studied optimally by the model. In this case, cropping the face is done to take only part of the face image, areas that are not faced are removed so that they are not too burdensome for computation, and the model does not learn information that is not important in the image [39, 41].

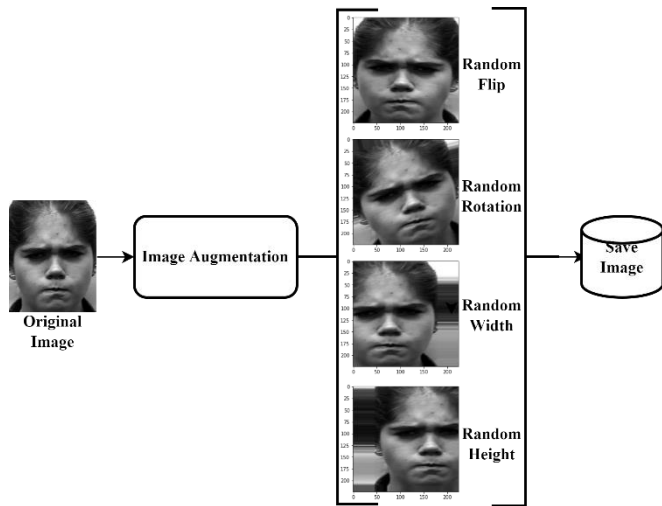


Figure 6. Image augmentation process

Deep learning requires many data variants to get optimal performance. Figure 6 data augmentation is a technique of manipulating an image without losing the essence or essence of the image. Augmentation performed Random Flip, Random Rotation, Random Height, Random Width.

### 4.3 Experimental setup

OpenCV library is used to do image cropping. When entering a model, the image will be resized to a size of 224 x 224 because, by default, the resolution size of the deep CNN architecture, especially EfficientNet, is 224 x 224. For the optimizer parameter, Adam is used by considering the learning rate value for the feature extraction process as 0.001 with 50 epochs. Then for the feature classification process, we unfreeze five to seven layers close to the output layer, then train them with a learning rate of 0.00001 and epochs from 51 to 100 epochs. The small learning rate value prevents us from destroying what EfficientNet has learned from ImageNet datasets. Meanwhile, only minor augmentation is applied to the data, with the augmentation settings Rotation: (-10° to 10°), Scaling factor: 1.1, and Horizontal Flip. Minor adjustments to the original image enhance its accuracy.

In this research, to train and experiment in terms of testing the performance of a state-of-the-art facial expression recognition model, the researcher used the TensorFlow framework, using the Python programming language with Keras. In this case, the researcher's training process used a GPU from google with specifications Tesla K80 Cuda Version:

11.2, NVIDIA-SMI 495.46.

How this add-in operates as follows First, when the model has gone through the fitting process. The next step of the model is saved into HDF5 format to ingest the weight from the results of the training that has been conducted by the model. Next, it uses the load\_model() function to be accommodated in the model variables. Then the model is tested with 10% of the testing data using the model.predict() function. Testing data is data that is separate from training data. After testing the model, the calculation evaluation was conducted using performance metrics, classification reports and confusion matrix using the scikit learn library. Researchers tested each model individually.

### 4.4 Experimental results and analysis

In this experiment, the efficiency of the proposed model for processing benchmark datasets is CK+ and JAFFE. Because CNN are architectures that consist of the proposed building blocks. Initially conducted a series of experiments using the standard CNN model to identify the initial performance. Then look for the difference in the impact of fine-tuning using EfficientNet-B0. Finally, create a model with fine-tuning techniques using a pre-trained architecture from the EfficientNet family.

Table 1. Testing accuracies of standard CNN CK+ and JAFFE datasets for various image input sizes are shown

| Input Image Size | CK+    | JAFFE  |
|------------------|--------|--------|
| 360 × 360        | 57.58% | 91.69% |
| 224 × 224        | 72.73% | 87.50% |
| 128 × 128        | 93.18% | 91.67% |
| 64 × 64          | 93.94% | 88.33% |
| 48 × 48          | 95.45% | 79.17% |

The test set accuracies for standard CNN with two layers with 5×5 size kernel, and 2×2 MaxPooling for various input sizes ranging from 360×360 to 48×48 are presented in Table 1 for both the CK+ and JAFFE datasets. The test size was chosen at random from 10% of the available data. The presented results are the best test set accuracies for a total of 50 iterations for a specific setting. For example, in Table 1, the results and analysis can be described using CK+ image input measuring 48×48, resulting in a testing accuracy of 95.45% because CK+ images have been preprocessed so that they have dimensions of 48×48 pixels before entering into a model and when inputting images measuring 360x360 produce accuracy 57.58% is because the image will be blurry because the sharpness is reduced due to the resizing process from 48×48 to 360×360. To use the JAFFE datasets in this observation, it was found that the best accuracy was obtained from the image sizes of 360×360 and 128×128, respectively 91.67% and 91.67%. This is because the sharper the image size, the information obtained from the model is smoother, and the sharper the model is smarter.

Because fine-tuning and its mode are critical in the proposed transfer learning-based system, experiments were conducted with various fine-tuning modes to gain a better understanding. On the CK+ and JAFFE datasets, Table 2 shows the accuracy of the testing data (10% of randomly selected data) of the proposed model with EfficientNet for different transfer learning modes.

In this experiment, the researcher compares two training modes and the features extraction model's histories with the

fine-tuning model. In this case, the CK+ and JAFFE datasets have undergone the data augmentation process before entering each model. First, the researcher did it in EfficientNetB0 as the base model, then set up the base model and froze its layers (this will extract features). Then set up model architecture with trainable top layers. The tuning parameters given in this training are using the loss categorical cross-entropy function and the optimizer using the adam function with a learning rate value of 0.0001 each so that the model in updating the weights is softer and the model gets more information. A model with a lower learning rate (it's typically best practice to lower the learning rate when fine-tuning). Second, to apply fine-tuning of the model, the researcher performs Unfreeze all the layers in the base model, then proceeds to Refreeze every layer except the last six because this is for which the top six layers of EfficientNetB0 can be trained according to custom datasets about this research. Each training mode for each model uses 50 epochs for the feature extraction model and 50 epochs for the fine-tuning model. So, the model has already done 50 epochs (feature extraction). This is the total number of epochs we're after (50 + 50 = 100).

**Table 2.** Testing data accuracy with EfficientNetB0 from different training modes

| Training Mode           | CK+    | JAFFE  |
|-------------------------|--------|--------|
| Feature Extraction Mode | 73.48% | 97.31% |
| Fine-Tuning             | 99.57% | 100.0% |

To find a suitable and best model, researchers experimented with using the architecture, specifically the EfficientNet model family (EfficientNet-B0, EfficientNet-B01, EfficientNet-B02, EfficientNet-B03, EfficientNet-B04, EfficientNet-B05, EfficientNet-B06, and EfficientNet-B07). The number with the model's name represents the EfficientNet B1 to B7 obtained using different scale coefficients in which different combined coefficient values produce EfficientNet B1-B7. Experiments were conducted on 10% of the data chosen at random as a test (i.e., 90% as a training set), and the accuracy of the test set after 50 iterations of fine-tuning is shown in Table 3. In the selected 10% test data case for the JAFFE data set, all models achieved 100% accuracy. The accuracy of the CK+ dataset, on the other hand, ranges from 95.45% to 99.57% at 10% of the selected test data. The CK+ data set is much larger than the JAFFE data set.

**Table 3.** Comparison of testing accuracies with various EfficientNet models on the CK+ and JAFFE

| Pre-Trained EfficientNet Model | CK+ in Selected 10% Test Sample | JAFFE in Selected 10% Test Sample |
|--------------------------------|---------------------------------|-----------------------------------|
| EfficientNet-B01               | 99.24%                          | 100.0%                            |
| EfficientNet-B02               | 99.24%                          | 100.0%                            |
| EfficientNet-B03               | 99.37%                          | 100.0%                            |
| EfficientNet-B04               | 97.73%                          | 100.0%                            |
| EfficientNet-B05               | 99.24%                          | 100.0%                            |
| EfficientNet-B06               | 95.45%                          | 100.0%                            |
| EfficientNet-B07               | 98.48%                          | 100.0%                            |

In Table 3, we can see that the model using EfficientNetB03 has an accuracy of 95.45%, especially studying the CK+ pattern. Exploring the JAFFE dataset of the EfficientNet architecture family starting from EfficientNetB1-B7 yielded 100.0% accuracy. Why this is different is because the CK+

datasets have more variety of images and a greater number of images which is 1308 out of eight classes of the sum of each expression. JAFFE only has 213 images and seven classes for each expression, so the EfficientNet model can be studied easily because it has little data variance. In studying patterns from the CK+ datasets, the EfficientNetB1-B7 model family has different accuracy because each EfficientNet category has several different parameter sizes ranging from the depth, width, and resolution scale of the network in making each CNN architecture to produce several models which are commonly called the EfficientNet model family.

To find a suitable and best model, researchers experimented using the architecture, specifically the EfficientNet model family (EfficientNet-B0, EfficientNet-B01, EfficientNet-B02, EfficientNet-B03, EfficientNet-B04, EfficientNet-B05, EfficientNet-B06, and EfficientNet-B07). The number next to the model's name represents the EfficientNet B1 to B7 obtained by varying the scale coefficients, with different combined coefficient values producing EfficientNet B1-B7.

Experiments were conducted on 10% of the data chosen at random as a test (i.e., 90% as a training set), and the accuracy of the test set after 50 iterations of fine-tuning is shown in Table 3. In the selected 10% test data case for the JAFFE data set, all models achieved 100 % accuracy. The accuracy of the CK+ dataset, on the other hand, ranges from 93.47% to 98.78% at 10% of the selected test data. This is due to the fact that the CK+ data set is much larger than the JAFFE data set.

In testing the performance of this testing, the researcher uses the F1-Score as Eq. (1). Performance metrics measurement because it is good for measuring imbalanced datasets in which the number of each Class is not balanced.

$$F1 - Score = \frac{2 * precision * recall}{precision + recall} \quad (1)$$

Table 4 shows the classification of the training results conducted by the EfficientNetB0 model on 132 test images from the CK+ data set. In which the model predicts the image of sadness is misclassified as disgust, and surprise is misclassified as sadness.



**Table 4.** The F1-score classification of each expression class in the CK+ dataset

| Expression    | True Label |    |    |    |    |    |    |    |
|---------------|------------|----|----|----|----|----|----|----|
|               | AN         | CO | DI | AF | HA | NE | SA | SU |
| Anger (AN)    | 13         | 0  | 0  | 0  | 0  | 0  | 0  | 0  |
| Contempt (CO) | 0          | 5  | 0  | 0  | 0  | 0  | 0  | 0  |
| Disgust (DI)  | 0          | 0  | 17 | 0  | 0  | 0  | 1  | 0  |
| Fear (AF)     | 0          | 0  | 0  | 8  | 0  | 0  | 0  | 0  |
| Happy (HA)    | 0          | 0  | 0  | 0  | 21 | 0  | 0  | 0  |
| Neutral (NE)  | 0          | 0  | 0  | 0  | 0  | 33 | 0  | 0  |
| Sadness (SA)  | 0          | 0  | 0  | 0  | 0  | 0  | 10 | 1  |
| Surprise (SU) | 0          | 0  | 0  | 0  | 0  | 0  | 0  | 24 |

Table 5 shows only two images that were misclassified by one of the EfficientNet-B0 models and reliability of the model in making predictions built with the transfer learning approach proposed by the researchers. The first image belongs to the label of the type of expression, but the model predicts as an expression of disgust. In this case, the model of difficulty in discriminating images of sadness is almost like the expression

of disgust specific to the facial expressions of the person presented in this Table 5, but visually the expressions of sadness and disgust have almost similarities such as the eyebrows wrinkle down or sometimes upwards and the lips tend to shrink, the teeth are closed, and the skin of the face looks wrinkled. The second image belongs to the label of the type of surprise expression, but the predictive model is included in the expression of sadness. Why this can happen is because specifically for a person's face in this second picture, when surprised the model predicts looking sad because his eyebrows seem to express sadness, but visually the expression of surprise and sadness has a slight difference that lies only in the state of the eyebrows tending upwards to widen and the skin wrinkles upwards.

**Table 5.** Misclassified images from the CK+ datasets

| Misclassified Image: True Class → Predicted Class |   |   |
|---|---|---|
| Samples From CK+                                  |  |  |
|   | Sadness → Disgust   | Surprise → Sadness  |

In the analysis of Table 5, it can be concluded that the state of expression of each person tends to be different and is influenced by certain skin color, age level, and ethnicity. In

other typical, when two people have the same expressions, it is not necessarily that the two people have expressions that look the same visually since everyone has a unique character in each of their expressions, although most of each expression can be classified. In certain cases, for example, visually expressing fear but the model predicts the person is being surprised, and the human being also has difficulty in predicting the expressions of everyone.

#### 4.5 Results comparison with existing method

Using the CK+ and JAFFE datasets, this section compares the proposed FER method's performance to that of existing leading expression recognition methods. Table 6 also includes the separation of training and test data, as well as the specific properties of the individual methods, for a better understanding of the accuracy of test set recognition. The analysis incorporates both traditional and deep learning-based methods. Most existing techniques make use of the JAFFE dataset, which has only 213 samples. The CK+ dataset, on the other hand, is significantly larger, with 1308 images containing frontal views. It should be noted that the front view image is easier to classify than the front view and profile image. In the existing studies, various strategies for separating training and test samples were used, as shown in Table 6. Furthermore, the comparison table's presentation of the significance of each method with the technique (used in feature selection and classification) is very useful for understanding transfer learning technique proficiency.

**Table 6.** Comparison of the accuracy of proposed method with existing work on CK+ and JAFFE datasets

| Work [Ref.], Year                       | Total Sample: Training and Test Division    | Test Set Accuracy |        | Method's Significant in Feature Extraction and Feature Classification                     |
|---|---|-------------------|--------|---|
|   |   | CK+               | JAFFE  |   |
| Jain et al. [32]                        | 213: 90%+10%                                |                   | 94.91% | CNN and RNN-based deep learning architecture  |
| Bendjillai et al. [43]                  | 213: 90%+10%                                |                   | 98.63% | CNN is used for image enhancement, feature extraction, and classification.                |
| Moravcik and Basterrech [3]             | 1308: 90%+10%                               | 95.00%            |        | VGG-Type for feature extraction and classification with Neural Network                    |
| Xu et al. [44]                          | 10-fold cross validation testing CK+# 1308: | 98.99%            |        | Convolutional Neural Networks and Edge Computing  |
| Proposed technique With EfficientNet-B0 | 90%+10% JAFFE# 213# 90%+10%                 | 99.57 %           | 100.0% | Transfer Learning using a pipeline strategy to fine-tune a pre-trained EfficientNet model |

## 5. DISCUSSION

Expression recognition from facial images in uncontrolled environments (e.g., public places), where obtaining a front view image is not always possible, is becoming increasingly important. To accomplish this goal, it is critical to employ a facial expression recognition model that allows for the recognition of expressions from a variety of facial expressions, particularly views from various angles. The profile view from multiple angles, however, does not show the front sight's landmark features, and traditional feature extraction methods cannot extract facial expression features from the profile view. As a result, facial expression recognition from high-resolution facial images using the EfficientNet model is regarded as the only viable option for overcoming such a difficult task. The proposed model was created using a transfer learning-based approach: ImageNet's pre-trained EfficientNet was made compatible with a facial expression recognition system that

replaced its overlying layer with a solid layer to enhance the model with facial expression data. A feature of the proposed method is the pipeline training strategy for fine-tuning: solid layers are tuned first, followed by tuning other EfficientNet layers sequentially.

The proposed technique performed admirably when evaluating datasets with frontal and profile views. The JAFFE dataset only contains the frontal view in grayscale, whereas the CK+ dataset contains the frontal view with more color variations, causing the recognition task more complex. We believe that testing on two different data sets is sufficient to demonstrate proficiency and that the proposed method will perform well on the other data set. However, datasets with low-resolution images or cases with a high degree of imbalance will necessitate additional preprocessing and appropriate modification in these methods, which will be the subject of future research. Working with images from uncontrolled environments or video sequences is also on the

agenda for future research.

Because generalizability (performance on invisible data) is an important attribute in machine learning, the concept of a test set (a sample that was not used in any training step) was used to validate the proposed model. Two popular techniques for keeping a test set are a fixed number of sample reservations from available data and cross-validation (backup of all samples in a round). This study considers both methods, while it is common to follow anyone. The test kit is only used to validate the proposed model at the end. Based on the achieved test set accuracy, the proposed method outperformed the existing one. It is worth noting that the proposed method only misclassifies a few images with perplexing perspectives, while the overall recognition accuracy remains very high. As a result, the method proposed in this paper holds promise for practical scenarios in which image classification is approached from the front.

A fundamental task for any machine learning system is the selection of parameter values. Only the top solid layer of the pre-trained inner EfficientNet is replaced by the corresponding multiple layers in the proposed facial expression recognition model. Several experiments focusing on pipe training problems led to the selection of hyperparameters. There is an opportunity to enhance the performance of the proposed method by further optimizing each specific EfficientNet model parameter for each data set.

## 6. CONCLUSIONS

This paper proposes an effective EfficientNet architecture for facial expression recognition which combines transfer learning with a pipeline tuning strategy. According to experimental results, the proposed method shows high recognition accuracy using eight different pre-trained EfficientNet architectures on the CK+ and JAFFE facial expression datasets with frontal profile display. In this study, experiments were conducted with general settings apart from the DCNN model, which was trained for simplicity, and some confusing facial images, primarily profile views, were misclassified. Further adjusting pre-trained individual models' hyperparameters and exceptional attention to profile views can improve classification accuracy. Current research, especially performance with profile views, will be compatible with a broader range of real-life industrial applications, such as monitoring patients in hospitals or surveillance security and monitoring students in learning. Furthermore, to cover emerging industrial applications, the concept of facial expression recognition can be extended to expression recognition from speech or gestures.

## REFERENCES

- [1] Izard, C.E., Woodburn, E.M., Finlon, K.J., Stephanie Krauthamer-Ewing, E., Grossman, S.R., Seidenfeld, A. (2011). Emotion knowledge, emotion utilization, and emotion regulation. *International Society for Research on Expression*, 3(1): 44-52. <https://doi.org/10.1177/1754073910380972>
- [2] Sayette, M.A., Cohn, J.F., Wertz, J.M., Perrott, M.A., Parrott, D.J. (2001). A psychometric evaluation of the facial action coding system for assessing spontaneous expression. *Journal of Nonverbal Behavior*, 25(3): 167-185. <https://doi.org/10.1023/A:1010671109788>
- [3] Moravčik, E., Basterrech, S. (2021). Image-based facial expression recognition using convolutional neural networks and transfer learning. *5th International Scientific Conference on Intelligent Information Technologies for Industry, IITI 2021*, 330: 3-14. [https://doi.org/10.1007/978-3-030-87178-9\\_1](https://doi.org/10.1007/978-3-030-87178-9_1)
- [4] Khan, A., Sohail, A., Zahoora, U., Saeed, A. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 53(8): 5455-5516. <https://doi.org/10.1007/s10462-020-09825-6>
- [5] Simonyan, K., Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. Published as a Conference Paper at ICLR 2015, pp. 1-14. <https://arxiv.org/abs/1409.1556>
- [6] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition Deep, Las Vegas, NV, USA*. <https://doi.org/10.1109/CVPR.2016.90>
- [7] Szegedy, C., Vanhoucke, V., Shlens, J. (2016). Rethinking the inception architecture for computer vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA. <https://doi.org/10.1109/CVPR.2016.308>
- [8] Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q. (2018). Densely connected convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-9. <https://doi.org/10.1109/CVPR.2017.243>
- [9] Tan, M., Le, Q.V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *36th International Conference on Machine Learning, ICML 2019*, 10691-10700. <https://arxiv.org/abs/1905.11946>
- [10] Ko, B.C. (2018). A brief review of facial expression recognition based on visual information. *Sensors (Switzerland)*, 18(2): 401. <https://doi.org/10.3390/s18020401>
- [11] Liew, C.F., Yairi, T. (2015). Facial expression recognition and analysis: A comparison study of feature descriptors. *IPSI Transactions on Computer Vision and Applications*, 7: 104-120. <https://doi.org/10.2197/ipsjtcv.7.104>
- [12] Huang, Y., Chen, F., Lv, S., Wang, X. (2019). Facial expression recognition: A survey. *Symmetry (Basel)*, 11(10): 1189. <https://doi.org/10.3390/sym11101189>
- [13] Li, S., Deng, W. (2020). Deep facial expression recognition: A survey. *IEEE Transactions on Affective Computing*, 3045: 1-20. <https://doi.org/10.1109/TAFFC.2020.2981446>
- [14] Qi, X.X., Jiang, W. (2007). Application of wavelet energy feature in facial expression recognition. *2007 IEEE International Workshop on Anti-counterfeiting, Security, Identification, ASID*, pp. 169-174. <https://doi.org/10.1109/IWASID.2007.373720>
- [15] Zhao, L.H., Zhuang, G.B., Xu, X.H. (2008). Facial expression recognition based on PCA and NMF. *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, pp. 6826-6829. <https://doi.org/10.1109/WCICA.2008.4593968>
- [16] Feng, X., Pietikäinen, M., Hadid, A. (2007). Facial expression recognition based on local binary patterns. *Pattern Recognition and Image Analysis*, 17(4): 592-598. <https://doi.org/10.1134/S1054661807040190>



- [17] Zhi, R.C., Ruan, Q.Q. (2008). Facial expression recognition based on two-dimensional discriminant locality preserving projections. *Neurocomputing*, 71(7-9): 1730-1734. <https://doi.org/10.1016/j.neucom.2007.12.002>
- [18] Lee, C.C., Shih, C.Y., Lai, W.P., Lin, P.C. (2002). An improved boosting algorithm and its application to facial expression recognition. *Journal of Ambient Intelligence and Humanized Computing*, 3(1): 11-17. <https://doi.org/10.1007/s12652-011-0085-8>
- [19] Chang, C.Y., Huang, Y.C. (2010). Personalized facial expression recognition in indoor environments. *Proceedings of the International Joint Conference on Neural Networks, Barcelona, Spain*. <https://doi.org/10.1109/IJCNN.2010.5596316>
- [20] Wang, P.S.P. (2008). Performance comparisons of facial expression recognition in jaffe database. *International Journal of Pattern Recognition and Artificial Intelligence*, 22(3): 445-459. <https://doi.org/10.1142/S0218001408006284>
- [21] Shan, C.F., Gong, S.G., Mcowan, P.W. (2009). Facial expression recognition based on Local Binary Patterns: A comprehensive study. *Image and Vision Computing*, 27(6): 803-816. <https://doi.org/10.1016/j.imavis.2008.08.005>
- [22] Jabid, T., Kabir, M.H., Chae, O. (2010). Robust facial expression recognition based on local directional pattern. *ETRI Journal*, 32(5): 784-794. <https://doi.org/10.4218/etrij.10.1510.0132>
- [23] Alshamsi, H., Kepuska, V. (2017). Real time automated facial expression recognition app development on smart phones. *2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 384-392. <https://doi.org/10.1109/IEMCON.2017.8117150>
- [24] Joseph, A., Geetha, P. (2019). Facial expression detection using modified eyemap – mouthmap algorithm on an enhanced image and classification with tensorflow. *The Visual Computer*, 36: 529-539. <https://doi.org/10.1007/s00371-019-01628-3>
- [25] Zhao, X.M., Shi, X.G., Zhao, S.Q. (2015). Facial expression recognition via deep learning. *IETE Technical Review*, 32(5): 37-41. <https://doi.org/10.1080/02564602.2015.1017542>
- [26] Li, M., Xu, H., Huang, X., Song, Z., Liu, X., Li, X. (2018). Facial expression recognition with identity and expression joint learning. *IEEE Transactions on Affective Computing*, 12(2): 544-550. <https://doi.org/10.1109/TAFFC.2018.2880201>
- [27] Mollahosseini, A., Chan, D., Mahoor, M.H. (2016). Going deeper in facial expression recognition using deep neural networks. *2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, USA*. <https://doi.org/10.1109/WACV.2016.7477450>
- [28] Pons, G., Masip, D. (2017). Supervised committee of convolutional neural networks in automated facial expression analysis. *IEEE Transactions on Affective Computing*, 9(3). <https://doi.org/10.1109/TAFFC.2017.2753235>
- [29] Wen, G.H., Hou, Z., Li, H.H., Li, D.Y., Jiang, L.J., Xun, E.Y. (2017). Ensemble of deep neural networks with probability-based fusion for facial expression recognition. *Cognitive Computation*, 9: 597-610. <https://doi.org/10.1007/s12559-017-9472-6>
- [30] Ruiz-Garcia, A., Elshaw, M., Altahhan, A., Palade, V. (2017). Stacked deep convolutional auto-encoders for expression recognition from facial expressions. *2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA*. <https://doi.org/10.1109/IJCNN.2017.7966040>
- [31] Ding, H., Zhou, S.K., Chellappa, R. (2017). FaceNet2ExpNet: Regularizing a deep face recognition net for expression recognition. *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pp. 118-126. <https://doi.org/10.1109/FG.2017.23>
- [32] Jain, N., Kumar, S., Kumar, A., Shamsolmoali, P., Zareapoor, M. (2018). Hybrid deep neural networks for face expression recognition. *Pattern Recognit. Pattern Recognition Letters*, 115: 101-106. <https://doi.org/10.1016/j.patrec.2018.04.010>
- [33] Bendjillali, R.I., Beladgham, M., Merit, K. (2019). Improved facial expression recognition based on dwt feature for deep CNN. *Electronics*, 8(3): 324. <https://doi.org/10.3390/electronics8030324>
- [34] Devi, M.S. (2019). Expression recognition from facial expression using deep convolutional neural network. *Journal of Physics: Conference Series, Volume 1193, 2018 International Conference of Computer and Informatics Engineering, Bogor, Indonesia*. <https://doi.org/10.1088/1742-6596/1193/1/012004>
- [35] Shi, M.I.N. (2020). A novel facial expression intelligent recognition method using improved convolutional neural network. *IEEE Access*, 8: 57606-57614. <https://doi.org/10.1109/ACCESS.2020.2982286>
- [36] Ngoc, Q.T., Lee, S. (2020). Facial landmark-based expression recognition via directed graph neural network. *Electronics*, 9(5): 764. <https://doi.org/10.3390/electronics9050764>
- [37] Jin, X., Sun, W., Jin, Z. (2019). A discriminative deep association learning for facial expression recognition. *International Journal of Machine Learning and Cybernetics*, 11: 779-793. <https://doi.org/10.1007/s13042-019-01024-2>
- [38] Rawat, W., Wang, Z.H. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural Computation*, 29(9): 1-98. [https://doi.org/10.1162/NECO\\_a\\_00990](https://doi.org/10.1162/NECO_a_00990)
- [39] Expression, F., Systems, R. (2020). Evaluation of data augmentation techniques for facial expression recognition systems. *Electronics*, 9(11): 1892. <https://doi.org/10.3390/electronics9111892>
- [40] Akhand, M.A.H., Roy, S., Siddique, N., Kamal, M.A.S., Shimamura, T. (2021). Facial expression recognition using transfer learning in the deep CNN. *Electronics (Switzerland)*, 10(9): 1036. <https://doi.org/10.3390/electronics10091036>
- [41] Hung, J.C., Lin, K.C., Lai, N.X. (2019). Recognizing learning expression based on convolutional neural networks and transfer learning. *Applied Soft Computing Journal*, 84: 105724. <https://doi.org/10.1016/j.asoc.2019.105724>
- [42] Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, L. (2010). The extended cohn-kanade dataset (CK+): A complete dataset for action unit and expression-specified expression. *2010 IEEE Computer Society Conference on Computer Vision and Pattern*

- Recognition - Workshops, San Francisco, CA, USA.  
<https://doi.org/10.1109/CVPRW.2010.5543262>
- [43] Bendjillali, R.I., Beladgham, M., Merit, K., Taleb-Ahmed, A. (2019). Improved facial expression recognition based on DWT feature for deep CNN. *Electronics* (Switzerland), 8(3): 324. <https://doi.org/10.3390/electronics8030324>
- [44] Xu, G.Z., Yin, H.R., Yang, J.H. (2020). Facial expression recognition based on convolutional neural networks and edge computing. In 2020 IEEE Conference on Telecommunications, Optics and Computer Science, TOCS 2020, pp. 226-232. <https://doi.org/10.1109/TOCS50858.2020.9339739>