

Enhancing Facial Expression Recognition and Analysis with EfficientNetB7

Chenxin Huang^a

School of Information Science and Engineering, Dalian Polytechnic University, Dalian, China

Keywords: Facial Expression Recognition, EfficientNetB7, Data Preprocessing, Capabilities.

Abstract: This paper introduces an advanced Facial Expression Recognition model utilizing EfficientNetB7 architecture. Through meticulous stages of data preprocessing, model training, testing, and evaluation, significant strides are made in accurately classifying various facial expressions. The model exhibits commendable performance, particularly excelling in identifying positive expressions. However, challenges persist in effectively distinguishing between neutral and sad expressions, warranting further investigation. Future enhancements could involve refining data preprocessing techniques, such as adversarial training and data synthesis, to bolster dataset diversity and robustness. Additionally, exploring more potent feature extraction methods, including ensemble learning and transfer learning, holds promise for augmenting recognition capabilities. To address nuances in neutral and sad expressions, integrating contextual cues or dynamic features into the model architecture is proposed. Moreover, enriching the model's understanding by incorporating auxiliary information like emotion dictionaries or sentiment labels offers a viable avenue for improvement. Overall, this study contributes insights and pathways for advancing Facial Expression Recognition systems towards greater accuracy and applicability in real-world scenarios.


1 INTRODUCTION

Facial emotions play a pivotal role in human communication, enabling us to decipher the intentions of others. Typically, individuals decipher emotional states like happiness, frustration, and anger by analyzing expressions on the face and vocal inflections. Surveys indicate that verbal communication accounts for only one-third of human interactions, whereas nonverbal cues constitute the remaining two-thirds (Ko, 2018). Among the various nonverbal cues, facial expressions are a prime source of information in interpersonal communication due to their emotional significance (Jain, 2018). Consequently, facial emotion research has garnered significant attention in recent decades, finding applications not just in the fields of perception and cognition, but also in the realm of emotion-aware computing and digitally animated sequences (Kaulard, 2012).

While humans find that replicating this task with a computer algorithm poses significant challenges (Mehendale, 2020). The traditional Facial Expression

Recognition involves two crucial steps: extracting features and recognizing emotions. Nowadays, Deep Neural Networks, particularly Convolutional Neural Networks, are extensively employed in Facial Expression Recognition due to their inherent ability to automatically extract features from images (Akhand, 2021). Researchers have enhanced model performance by increasing network depth (Mollahosseini, 2016), introducing residual connections (Bah, 2022), and adopting attention mechanisms (Daihong, 2021). However, despite the powerful abilities of Convolutional Neural Network models in attribute extraction and grouping, the accuracy of facial expression recognition is constrained by multiple factors, including variations in lighting, intensity of expressions, and occlusions (Borgalli, 2022). Hence, optimizing the Convolutional Neural Network model holds paramount importance in enhancing the accuracy of facial expression recognition.

This study assesses the effectiveness of the EfficientNetB7 model in tasks related to facial expression recognition. With a batch size of 20 and

^a <https://orcid.org/0009-0002-8510-1478>

just one epoch, the model's proficiency in extracting pivotal features from facial images and effectively categorizing diverse expressions is meticulously analyzed. Notably, the model showcases a remarkable stability in accuracy and training loss as the training progresses, indicating its robustness in handling the given configuration. Despite the relatively limited number of training iterations, the inherent superior performance of the EfficientNetB7 model shines through, enabling it to exhibit commendable recognition capabilities. However, amid these successes, certain areas for enhancement emerge. Particularly, the model's accuracy in discerning between neutral and sad expressions requires refinement, as it tends to misclassify these expressions. This underscores the necessity for further optimization to unlock the full potential of the model in accurately capturing subtle nuances in facial expressions.

2 METHODOLOGIES

2.1 Dataset Description and Preprocessing

The dataset was obtained from the Kaggle platform (Kero, 2024). The given dataset comprises multiple categories of facial expressions, each containing a specific number of files. Among them, the "angry" category holds 958 files, depicting the facial expressions of characters when they are feeling angry. The "disgusted" category contains 111 files, showing the disgusted facial expressions of the characters. The "fearful" category features 1024 files, capturing the fearful expressions of characters. The "happy" category boasts 1774 files, exhibiting the happy facial expressions of people. Additionally, the "neutral" category holds 1233 files, displaying characters with neutral or non-expressive facial expressions. The "sad" category contains 1247 files, showing the sad facial expressions of people. Finally, the "surprised" category has 831 files, depicting the surprised facial expressions of characters. This comprehensive dataset offers a diverse range of facial expressions, allowing for detailed analysis and understanding of human emotions through facial cues.

The process involves systematically iterating through each file and folder within the directory, printing their paths as progress. Inside the dataset are crucial features and target variables utilized for machine learning task, along with facial images and their corresponding expression labels. These images

and labels are pivotal for training the model to recognize various facial expressions. Data enhancement techniques are employed during processing to expand the dataset and enhance model generalization, focusing on extracting essential information and constructing appropriate models for training. Preparation of test data for facial expression recognition begins by specifying the directory housing the test dataset, organized into subfolders dedicated to specific expression categories. Each subfolder is systematically traversed, gathering the list of files it contains. For each file, its complete path is generated by combining the subfolder's path with the filename, and these paths are collected in a list. Concurrently, the name of each subfolder, representing the associated expression label, is captured and stored in a separate list to ensure proper alignment. After iterating through all subfolders and files, two lists are obtained: one containing paths to all test files and the other containing corresponding expression labels. These lists are structured to facilitate their use in subsequent steps, allowing for efficient loading of test data and evaluation of the facial expression recognition model's performance on unseen data. In terms of hyperparameter tuning, adjustments are made according to project requirements and data characteristics, carefully selecting and tuning hyperparameters such as learning rate and batch size to optimize the model for higher recognition accuracy.

2.2 Proposed Approach

The expression recognition system based on Convolutional Neural Networks aims to automatically recognize facial expressions through deep learning techniques. The research goal is to construct an efficient and accurate model that can process image data and classify different facial expressions effectively. To achieve this goal, the following research methodology process is used: first, import key libraries to support data processing and model construction; then, pre-process image data, including normalization, scaling, and other steps, to adapt to the input requirements of the model; subsequently, divide the dataset into training, validation, and test sets to evaluate the performance of the model; and then, use an image data generator to enhance the training data's diversity; display the training data samples to verify the data quality; construct the Convolutional Neural Network model structure, including convolutional layers, pooling layers, fully connected layers, etc.; fit the model and optimize the parameters by iterating to minimize the

loss function; evaluate the performance of the model on the validation set, including the accuracy and the loss values; perform prediction and classify the expressions on the test set; and, finally, save the trained model for future loading of the model for real-time prediction. The whole process covers the complete technical path from data preprocessing to model application, which provides effective support for the development of expression recognition technology. The pipeline is shown in the Figure 1.

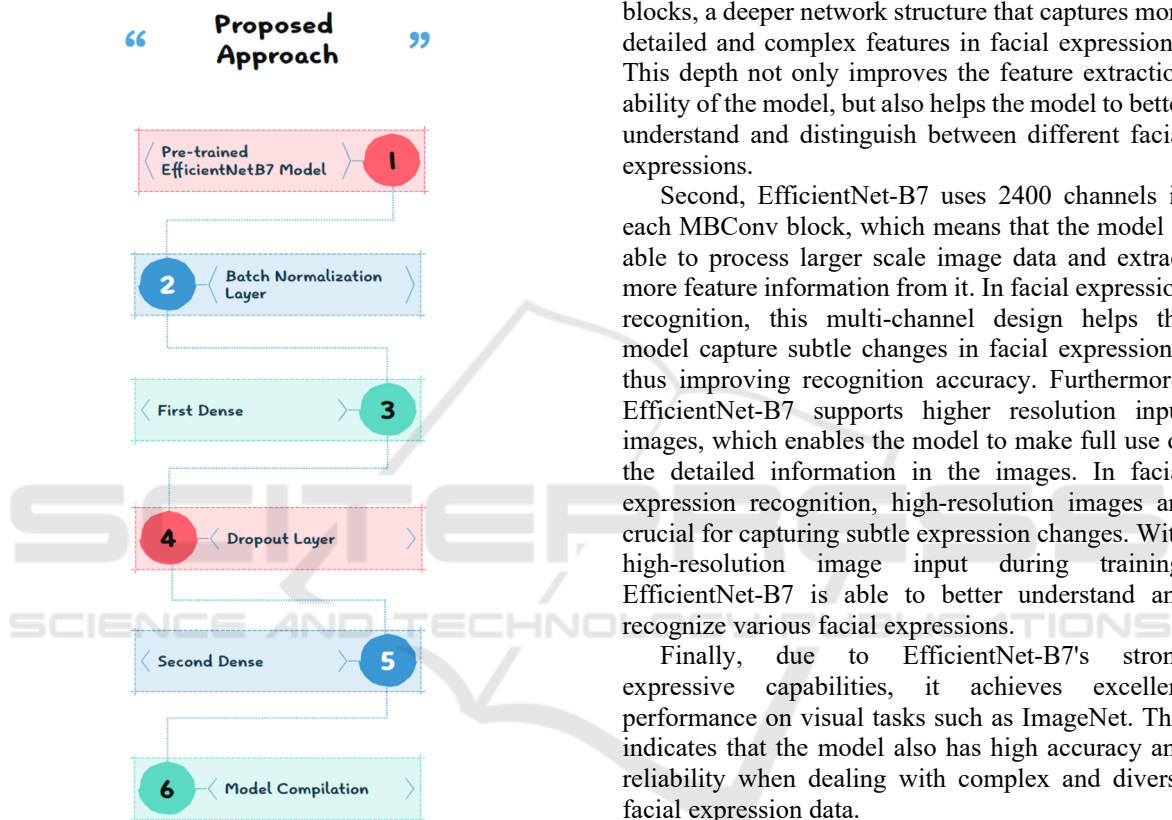


Figure 1: The pipeline of the study (Photo/Picture credit: Original).

2.2.1 EfficientNetB7

The version of the model loaded that does not include the top fully-connected layer, as one typically wants to define this part of the network structure according to one's task. The weights of the model are pre-trained from the ImageNet dataset, which helped the model converge faster on the new task because the pre-trained weights already contained some basic visual feature information.

Additionally, the shape of the model's input images are specified to ensure that the model would receive image data of the correct size. Finally, before the output layer of the model, a maximum pooling

operation is used, which transforms the feature map into a compact vector representation, which is useful for subsequently adding a classification layer and performing facial expression recognition. In this way, a custom model based on EfficientNetB7 is obtained, which can be used as a starting point for the facial expression recognition task and on which further optimization and training can be performed. EfficientNet-B7 was chosen because, firstly, it has 19 Mobile Inverted Bottleneck Convolution (MBConv) blocks, a deeper network structure that captures more detailed and complex features in facial expressions. This depth not only improves the feature extraction ability of the model, but also helps the model to better understand and distinguish between different facial expressions.

Second, EfficientNet-B7 uses 2400 channels in each MBConv block, which means that the model is able to process larger scale image data and extract more feature information from it. In facial expression recognition, this multi-channel design helps the model capture subtle changes in facial expressions, thus improving recognition accuracy. Furthermore, EfficientNet-B7 supports higher resolution input images, which enables the model to make full use of the detailed information in the images. In facial expression recognition, high-resolution images are crucial for capturing subtle expression changes. With high-resolution image input during training, EfficientNet-B7 is able to better understand and recognize various facial expressions.

Finally, due to EfficientNet-B7's strong expressive capabilities, it achieves excellent performance on visual tasks such as ImageNet. This indicates that the model also has high accuracy and reliability when dealing with complex and diverse facial expression data.

2.2.2 Dense/Dropout

The Dense and Dropout layers play a crucial role in deep learning, especially when building facial expression recognition models based on Convolutional Neural Networks. The Dense layer, commonly referred to as the fully connected layer, serves as a crucial element in neural networks. In Convolutional Neural Networks, the Dense layer is usually located after the convolutional and pooling layers, and is used to integrate the features extracted from the previous layers and map them to the final output space. In the Dense layer, each neuron establishes connections with all the neurons present in the preceding layer, thus being able to capture the global information of the input data. In the facial

expression recognition task, the Dense layer helps the model to learn and integrate features related to expressions, thus improving the accuracy of recognition.

However, excessively deep neural networks often tend to suffer from overfitting, a phenomenon where the model exhibits excellent performance on training data but performs poorly on unseen test data. To alleviate this problem, a Dropout layer is introduced into neural networks. The Dropout layer randomly sets the output of a portion of neurons to zero during training, thus preventing the model from relying too heavily on particular neurons or combinations of features. In this way, the Dropout layer helps to enhance the generalization ability of the model so that it can maintain good performance on unseen data.

In facial expression recognition models, the Dropout layer is usually placed after the Dense layer to reduce the complex co-adaptation between neurons in the Dense layer. This helps the model to learn a more robust and generalized feature representation, which improves the accuracy of facial expression recognition. Dense layer and a Dropout layer are defined to be used in the neural network model. The Dense layer has 256 neurons and reduces over-fitting by using L2 and L1 regularization techniques, where L2 applies regularization to the weights and L1 regularizes both the weights and the bias terms. Rectified Linear Unit (ReLU) is used as the activation function. Immediately after, the Dropout layer is used to prevent the model from over-dependence on specific neurons to improve generalization where the dropout probability is 0.45 and a fixed random seed 123 is used to ensure reproducible results.

2.2.3 Loss Function

The loss function plays a critical role in quantifying the disparity between the model's predictions and the true labels, guiding the model optimization process. It acts as a function that evaluates the discrepancy between the predicted outputs and the actual labels, yielding a numerical representation of this difference. A lower value signifies a higher alignment between the model's predictions and the true labels, indicating superior model performance.

The choice of loss function is particularly important in the task of facial expression recognition. Different loss functions may have a significant impact on the performance of the model. The Mean Square Error (MSE) loss function, which concentrates on

calculating the average squared difference between predicted and actual values, is well-suited for addressing regression problems, while the cross-entropy loss function is more suitable for classification problems, which takes into account the probability distribution of each category, so chose the latter. Here is the function of The Mean Square Error (MSE) loss, as:

$$MSE = \frac{1}{m} \sum_{i=1}^m (y - \hat{y}_i)^2 \quad (1)$$

2.3 Implementation Details

The project referenced the code on Kaggle (Kero, 2024) and took advantage of the functionality and performance of Python version 3.6 to complete the data processing and model construction for the facial expression recognition task. Through careful design and tuning, a data table containing test data paths and labels was successfully generated.

3 RESULTS AND DISCUSSION

In the task of facial expression recognition, the training parameters such as batch size and epochs have a significant impact on the final recognition performance of the EfficientNetB7 model. This study employs this model and focuses on exploring the specific effects of EfficientNetB7 on recognition performance. Next, the results obtained with a batch size of 20 and epochs count of 1 will be analyzed.

3.1 Training Loss and Validation Loss, Training Accuracy and Validation Accuracy

With the increase of epochs, the training loss becomes smaller, and the validation loss curve tends to be stable (see Figure 2). The optimal epoch is 6. As the epochs increase, the training accuracy becomes higher, and the validation accuracy also gradually increases and tends to be stable. The optimal epoch is 9, indicating that the EfficientNetB7 model performs well in facial expression recognition.

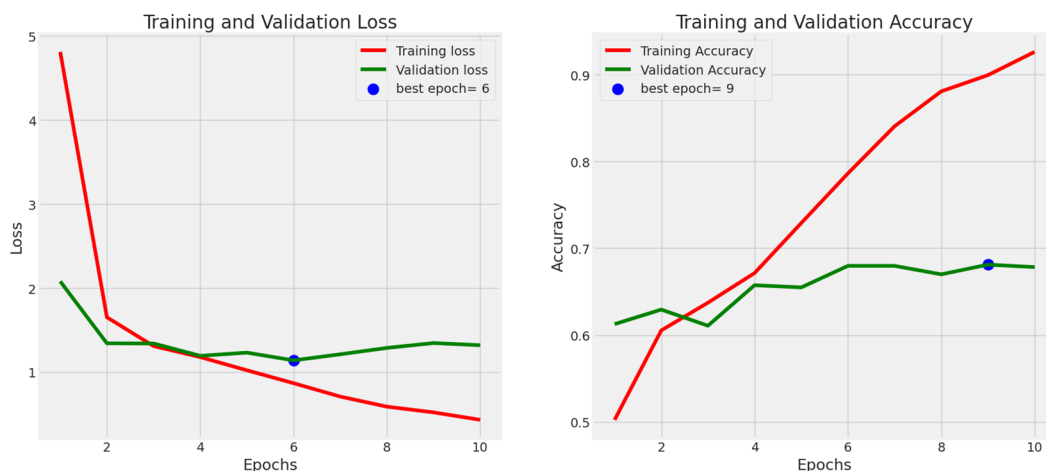


Figure 2: Training and validation loss and accuracy (Photo/Picture credit: Original).

3.2 The Accuracy of the Model Under Different Categories

Table 1 is the accuracy of the model under different categories. The model has the highest accuracy in recognizing happy expressions, reaching up to 90%, while the lowest accuracy is in recognizing sad expressions, which is 53%. Generally, the accuracy is higher when recognizing positive expressions and lower when recognizing negative expressions.

Table 1: Comparison with Faster R-CNN on PASCAL VOC 2017 dataset.

	Precision
Angry	0.63
Disgusted	0.68
Fearful	0.54
Happy	0.90
Neutral	0.59
Sad	0.53
Surprised	0.81

3.3 Confusion Matrix

By observing the confusion matrix (see Figure 3), some patterns of misclassification can be identified. The neutral expression is often misclassified as sad, and the sad expression is frequently misclassified as neutral. These two expressions share similar features, making it difficult for the EfficientNetB7 model to distinguish between them.

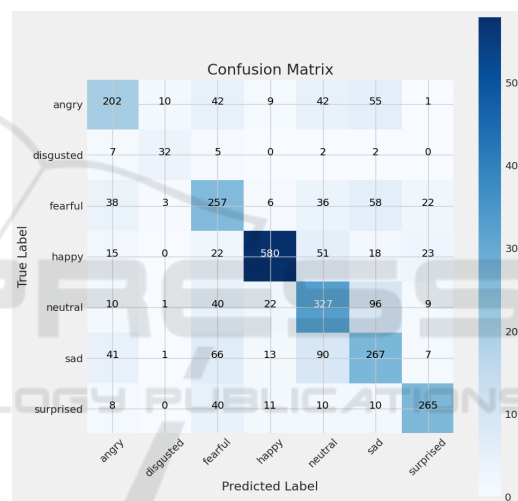


Figure 3: The confusion matrix (Photo/Picture credit: Original).

4 CONCLUSIONS

This paper develops a Facial Expression Recognition model using EfficientNetB7. Through data processing, model training, testing, and evaluation steps, this paper generates training loss and accuracy graphs, along with accuracy rates for expression recognition across different categories and their corresponding confusion matrices. Analysis of the results reveals EfficientNetB7's strong performance in facial expression recognition, particularly in identifying positive expressions. However, challenges remain in distinguishing neutral and sad expressions. To enhance the model's performance, future work can focus on several key areas. Firstly, further optimization of data preprocessing and

enhancement steps is warranted. Techniques like adversarial training and data synthesis can increase dataset diversity and robustness. Additionally, more detailed annotation and normalization of face images can mitigate the impact of individual differences and poses on recognition results.

Secondly, exploring more effective feature extraction and model training methods is essential. Techniques such as ensemble learning and transfer learning can bolster the model's recognition capabilities. To address issues with recognizing neutral and sad expressions, integrating more contextual information or dynamic features may capture subtle changes accurately. Additionally, enhancing the model's understanding of these expressions through auxiliary information like emotion dictionaries or sentiment labels holds promise.

REFERENCES

- Akhand, M. A. H., Roy, S., Siddique, N., Kamal, M. A. S., & Shimamura, T. (2021). Facial emotion recognition using transfer learning in the deep CNN. *Electronics*, vol. 10(9), p: 1036.
- Bah, I., & Xue, Y. (2022). Facial expression recognition using adapted residual based deep neural network. *Intelligence & Robotics*, vol. 2(1), pp: 78-88.
- Borgalli, M. R. A., & Surve, S. (2022, March). Deep learning for facial emotion recognition using custom CNN architecture. In *Journal of Physics: Conference Series*, vol. 2236(1), p: 012004).
- Daihong, J., Lei, D., (2021). Facial expression recognition based on attention mechanism. *Scientific Programming*, pp: 1-10.
- Jain, C., Sawant, K., Rehman, M., & Kumar, R. (2018, November). Emotion detection and characterization using facial features. In *2018 3rd International Conference and Workshops on Recent Advances and Innovations in Engineering (ICRAIE)* (pp. 1-6). IEEE.
- Kaulard, K., Cunningham, D. W., Bülthoff, H. H., & Wallraven, C. (2012). The MPI facial expression database—a validated database of emotional and conversational facial expressions. Vol. 7(3), p: e32321.
- Kero, A., (2024). Kaggle dataset. <https://www.kaggle.com/code/kirollosashraf/emotion-detection-cnn/input>
- Ko, B. C. (2018). A brief review of facial emotion recognition based on visual information. *sensors*, vol. 18(2), p: 401.
- Mehendale, N. (2020). Facial emotion recognition using convolutional neural networks (FERC). *SN Applied Sciences*, vol. 2(3), p: 446.
- Mollahosseini, A., Chan, D., & Mahoor, M. H. (2016, March). Going deeper in facial expression recognition using deep neural networks. In *2016 IEEE Winter conf. on applications of computer vision*, pp: 1-10).