Implementation of CNN in Facial Expression Analysis for Web-Based Emotion Detection

[1]Steven Ongki, [2]Jowie Ardinata, [3]Fanny, [4]Kelvin Chen
[1,2,3]Informatics Engineering, Mikroskil University
Email: [1]211110595@students.mikroskil.ac.id, [2]211110457@students.mikroskil.ac.id,
[3]211110163@students.mikroskil.ac.id, [4]kelvin.chen@mikroskil.ac.id

Article Info
Article history:

Keyword:
FER (Facial Expression Recognition)
Emotion Detection
Machine Learning
Convolutional Neural Network

ABSTRACT
A real-time facial expression and emotion detection system is presented, utilizing Convolutional Neural Networks (CNN) implemented directly in the user's web browser. The system, built using the Face Expression Recognition Dataset from Kaggle, automatically detects the user's face after granting camera access and analyzes facial features to classify emotions such as happiness, sadness, anger, and more. The entire detection process is performed within the browser, ensuring rapid results without the need for external applications or installations. Additionally, the system supports image and URL uploads for emotion analysis, offering versatility in detection. The analyzed emotional expressions are displayed instantly on the user's screen, providing an interactive and seamless user experience.
This study demonstrates a significant improvement in system accuracy compared to previous research, which utilized the FER2013 dataset and ran on Google Colab. While the prior study achieved 64% accuracy on test data, the proposed model in this research achieved 72% accuracy, providing higher accuracy and a more interactive and user-friendly experience.

Corresponding Author:
Fanny,
Informatics Engineering,
Mikroskil University,
Email: 211110163@students.mikroskil.ac.id

DOI: http://dx.doi.org/10.24014/ijaidm.v2i2.xxxx (10 pt)

1. INTRODUCTION

Facial Expression Recognition (FER) is a primary branch of artificial intelligence that seeks to understand human emotions by analyzing facial expressions [1]. The human face functions as a universal communication tool, conveying feelings and social messages through expressive movements without the need for verbal language [2]. Research has shown that human communication is predominantly non-verbal, with approximately 55% of information conveyed through facial expressions, 38% through voice intonation, and only 7% through spoken words [3]. Consequently, the ability to identify and interpret facial expressions is crucial in various aspects of life, ranging from social interactions to the development of advanced technologies, such as artificial intelligence and user experience-based applications [4].

Several previous studies have attempted to develop Facial Expression Recognition (FER) systems using various approaches and technologies, one of which employs the Convolutional Neural Network (CNN) method. For instance, research [5] demonstrated the performance of an Autoencoder + CNN + Attention model compared to CNN + Attention under various testing scenarios. The results revealed that the Autoencoder + CNN + Attention model achieved an accuracy of 64% on the Test Dataset Evaluation and an average accuracy of 65% on 100 random image tests, outperforming the CNN + Attention model, which only reached accuracies of 55% and 56%, respectively. However, when tested with external datasets, the accuracy of both models drastically dropped to 43%, highlighting the need for further adjustments in the model and dataset.

Although previous studies have significantly contributed to FER technology development, most systems developed so far are limited to executing code via platforms such as Google Colab, utilizing methods inaccessible to all users. Thus, while previous research provides a clear overview of CNN's potential in detecting facial expressions, this study employs a slightly different approach to achieve broader objectives.

Unlike prior research, this study offers a novel approach by developing a system accessible through a website and operable in any browser. The system employs a custom Convolutional Neural Network (CNN) with 25 layers, enabling deep data processing to produce more accurate facial expression classifications. This approach allows users to easily access the system without requiring specialized hardware or connections to external servers, making this technology more practical, faster, and flexible to use.

2. RESEARCH METHOD

This study aims to develop a web-based Facial Expression Recognition (FER) system utilizing Convolutional Neural Network (CNN) to recognize and classify facial expressions in real time. The following are the steps and stages implemented in this research:

2.1 Model Development Process

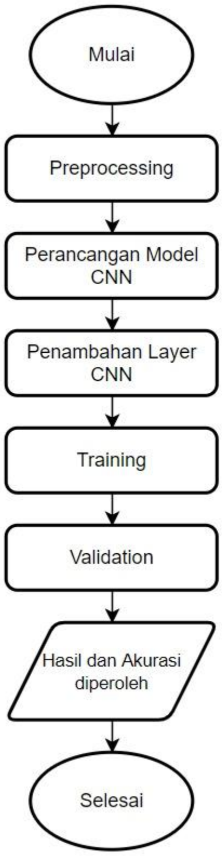


Figure 1. Model Development Process

The development of the CNN model was carried out through the following steps:
Explanation of Stages:

1. Start
   The initial phase of model development begins with the collection of relevant facial image data for training the model.
2. Preprocessing
   The collected facial images undergo preprocessing, including data cleaning, normalization, and image transformations such as resizing to enhance data quality.

3. CNN Model Design
   The CNN model is designed with an architecture consisting of several layers, such as convolutional, flatten, and fully connected layers, to extract deep features from facial images.
4. Addition of CNN Layers
   Each CNN layer aims to capture more complex features by adding deeper convolutional layers.
5. Training
   The model is trained on a large dataset using optimization algorithms such as backpropagation to adjust weights and achieve optimal performance.
6. Validation
   The model is tested on a validation dataset to evaluate its performance based on accuracy and loss metrics.
7. Results and Achieved Accuracy
   After training and validation, the model produces results with high accuracy in classifying facial expressions.
8. Completion
   The model is finalized and ready to be implemented locally on user devices via Flask without requiring an external server.

CNN Architecture Diagram :

Figure 2. Convolutional Neural Network Architecture

1. Input Layer
   In this stage, facial images are fed into the model as input data. These images typically undergo preprocessing, such as resizing to a fixed dimension (e.g., 48x48 pixels) and pixel value normalization. The input data is represented as a numerical matrix array, ready for processing by the CNN model.
2. Convolutional Layer
   The convolutional layer is responsible for extracting key features from the image, such as edge patterns, corners, and textures. This is achieved by applying filters (kernels) to the input image. The output of this stage is a feature map, which represents the important features found in the image.
3. Pooling Layer (Opsional)
   If pooling is used, this layer helps reduce the dimensions of the feature map while preserving critical information and decreasing computational complexity. Pooling can also prevent overfitting. Common techniques include max pooling and average pooling.
4. Flatten Layer
   Once features are extracted from the convolutional (and pooling) layers, the result is in the form of a two-dimensional matrix. The flatten layer converts this matrix into a one-dimensional vector, making it suitable for input into the fully connected layers.
5. Fully Connected Layer
   This layer connects all neurons from the flatten layer to the subsequent layers. It combines all extracted feature information to produce the final classification. The weights of these neurons are adjusted during training to improve prediction accuracy.
6. Output Layer
   The final layer provides the ultimate prediction as a classification of facial expressions. For instance, if the model is designed to detect seven expressions (happy, sad, angry, fearful, surprised, disgusted, and neutral), the output layer will have seven neurons, each representing one expression. The neuron with the highest activation indicates the detected expression.

2.2 Technology and Implementation

Once the model is developed, the technologies used to implement it include:

- Flask: A framework used to manage the CNN model and provide a web-based user interface.
- TensorFlow and Keras: Utilized to build and train the Convolutional Neural Network (CNN) model for facial expression detection.

- HTML, CSS, and JavaScript: Employed to create an interactive and responsive user interface, enabling image processing and prediction results to be displayed directly on the web page.

2.3 Model Testing Process

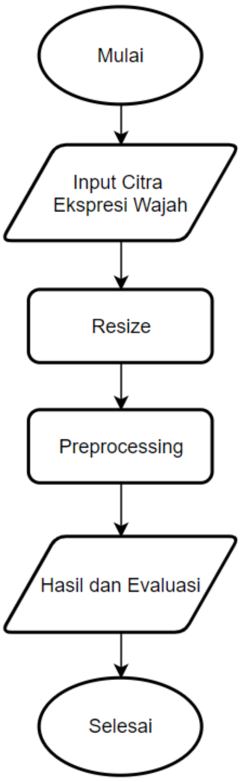


Figure 3. Model Testing Process

The testing process of the Facial Expression Recognition (FER) system follows the sequence below:
Explanation of Stages:

1. Start
   The process begins with accessing the camera or inputting a facial image for analysis.
2. Input Facial Expression Image
   The input facial image undergoes preprocessing, such as resizing, to ensure the image meets the required standard.
3. Resize
   The input facial image is resized to a specific standard, such as 48x48 or 224x224 pixels.
4. Processing
   The processed image then goes through feature extraction using the CNN model to recognize and classify the facial expression.
5. Results and Evaluation
   The results of the classification are evaluated to measure accuracy and loss values, ensuring optimal performance.
6. Completion
   The testing process concludes once the evaluation results show good performance, and the system is ready for real-time use in a browser without relying on external servers.

2.4 Dataset used

This study utilizes the dataset from Kaggle titled Face Expression Recognition Dataset [6]. This dataset can be accessed through the following link: Face Expression Recognition Dataset. The dataset includes thousands of facial images categorized by various expressions, such as happy, sad, angry, surprised, neutral, disgusted, and fearful. Some examples from the dataset used include:

Figure 4. Example from the Face Expression Recognition Dataset

Table 1. Face Expression Recognition Total Dataset

Emotion	**Total Data**	
	Train Data	**Test Data**
Angry	3993	960
Disgust	436	111
Fear	4103	1018
Happy	7164	1825
Neutral	4982	1216
Sad	4938	1139
Surprise	3205	797

2.5 System Implementation

The system is implemented locally on the user's device using Flask technology. The image data processing is carried out without relying on external servers, providing flexibility in usage while maintaining user data privacy.

3. RESULTS AND ANALYSIS

3.1 Model Training

In previous research, the dataset used was FER2013. In contrast, our study uses the Face Expression Recognition Dataset taken from Kaggle. The analysis results of this dataset are presented as follows:

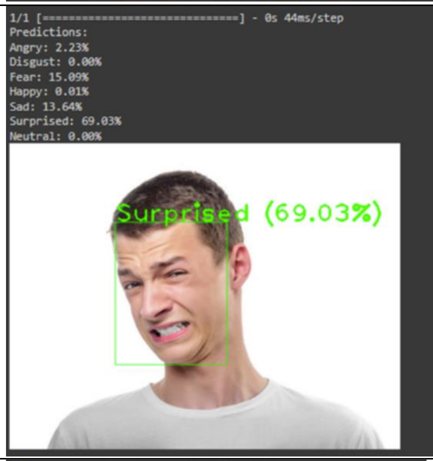
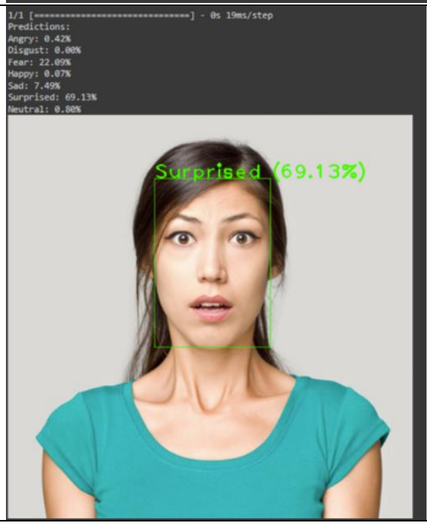
Training Results		
Model	**Accuracy**	**Loss**
Custom CNN with 25 layers	72%	0.7399
Autoencoder + CNN + Attention	64%	1.24

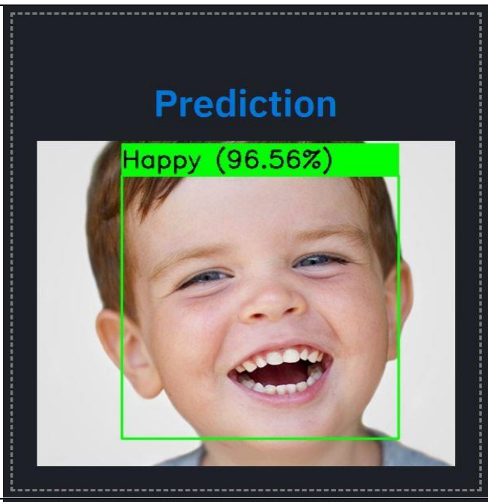
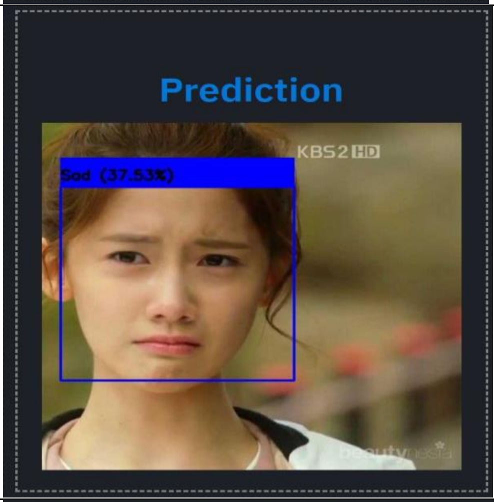
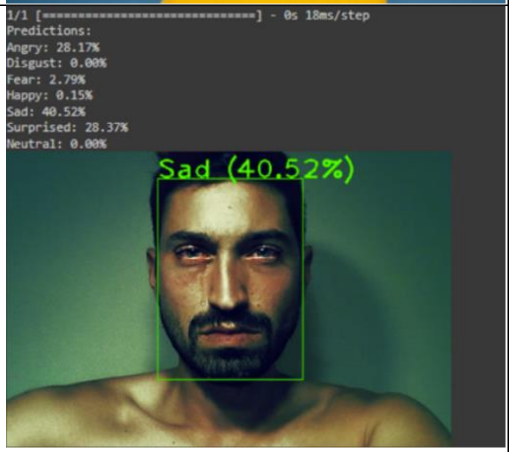
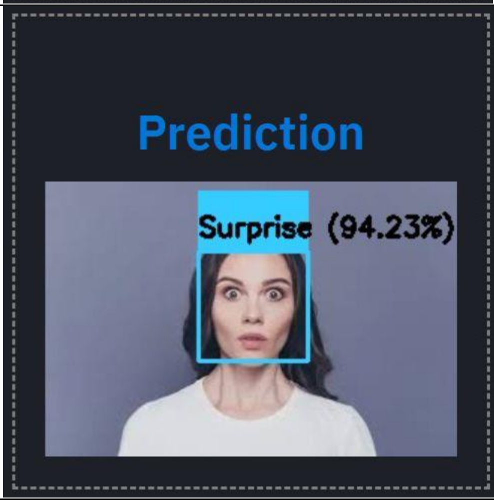
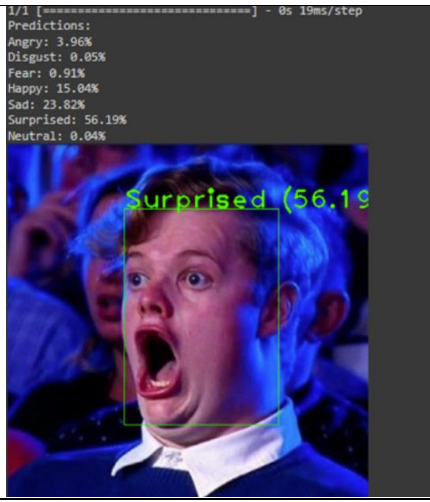
3.2 Results

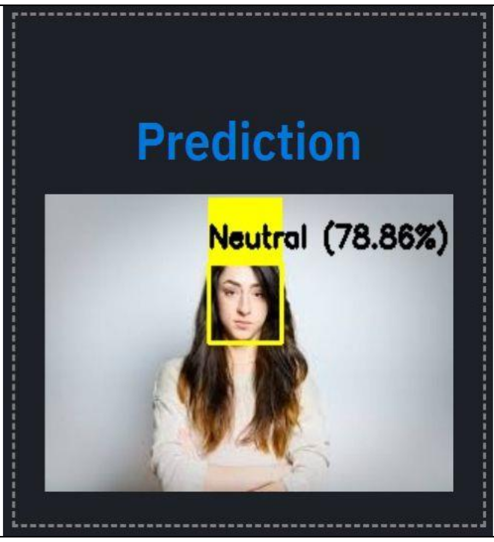
This study compares the performance of the developed Facial Expression Recognition (FER) model with previous research using the FER2013 dataset. The previous study employed an Autoencoder + CNN + Attention approach, achieving an accuracy of 64%. Meanwhile, this study uses the Face Expression Recognition Dataset from Kaggle and applies a custom CNN model with 25 layers.

Table 2. Comparison of Research Results Table

Expression	**Custom CNN with 25 layers**	**Autoencoder + CNN + Attention**

<table>
<tr>
<td>Angry</td>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>Disgust</td>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>Fear</td>
<td>
</td>
<td>
</td>
</tr>
</table>

<table>
<tr><td>Happy</td><td>
</td><td></td></tr>
<tr><td>Sad</td><td>
</td><td>
</td></tr>
<tr><td>Surprised</td><td>
</td><td>
</td></tr>
</table>

<table>
<tr>
<td>Neutral</td>
<td>
</td>
<td>
</td>
</tr>
</table>

4. CONCLUSION

This study successfully developed a Facial Expression Recognition (FER) system based on a custom Convolutional Neural Network (CNN) with 25 layers, accessible through a website and functional in any browser. The system uses the Face Expression Recognition Dataset from Kaggle and demonstrates better performance compared to previous research that utilized the FER2013 dataset.

The results of this study indicate that the developed model achieves an accuracy of 72% on the test dataset, an 8% improvement over the previous study's accuracy of 64%.

Another advantage of this system is its flexibility and ease of access. Unlike previous studies that required specialized hardware and were run through Google Colab, this system is designed to be implemented directly on the user's device via a browser, without reliance on external servers. This makes it more practical, faster, and secure for use by a wide range of users.

With these results, this study contributes significantly to the development of FER technology, both in terms of accuracy and implementation. However, further testing on more diverse datasets and improvements in model generalization are still needed to ensure optimal performance in real-world conditions.

REFERENCES

[1] H. V. Manalu e A. P. Rifai, “Detection of human emotions through facial expressions using hybrid convolutional neural network-recurrent neural network algorithm,” *Intelligent Systems with Applications,* vol. 21, nº 200339, pp. 1-18, 2024.

[2] M. Ihsan, R. K. Niswatin e D. Swanjaya, “DETEKSI EKSPRESI WAJAH MENGGUNAKAN TENSORFLOW,” *JOUTICA,* vol. 6, nº 1, pp. 428-433, 2021.

[3] D. Lapakko e A. College, “Communication is 93% Nonverbal: An Urban Legend Proliferates,” *Communication and Theater Association of Minnesota Journal,* vol. 34, pp. 7-19, 2015.

[4] N. Yudistira, “Peran Big Data dan Deep Learning untuk Menyelesaikan Permasalahan Secara Komprehensif,” *JURNAL MANAJEMEN SISTEM INFORMASI DAN TEKNOLOGI,* vol. 11, nº 2, pp. 78-89, 2021.

[5] M. A. Latumahina, H. Santoso, I. Wasito e H. , “Emotion Recognition System Using Autoencoder + CNN + Attention,” *Jurnal Teknik Informatika dan Sistem Informasi,* vol. 10, nº 4, pp. 150-161, 2023.

[6] J. Oheix, “Face expression recognition dataset,” Kaggle, 3 January 2019. [Online]. Available: https://www.kaggle.com/datasets/jonathanoheix/face-expression-recognition-dataset. [Accessed 28 December 2024].

BIBLIOGRAPHY OF AUTHORS

First author's photo(3x4cm)	First Author, make in paragraph (9 pt)
Second author's photo(3x4cm)	Second Author, make in paragraph (9 pt)
Thirth author's photo(3x4cm)	Thirth Author, make in paragraph (9 pt)