IJEEMCS : International Journal of Electrical Engineering, Mathematics and Computer Science
Vol. 1, No.4 December 2024
e-ISSN: 3048-1945, p-ISSN : 3048-1910, Page 27-36
DOI: https://doi.org/10.62951/ijeemcs.v1i4.123
Available online at: https://international.arteii.or.id/index.php/IJEEMCS

Real-Time Facial Emotion Detection Application with Image Processing Based on Convolutional Neural Network (CNN)

Ghaeril Juniawan Parel Hakim [1*], Gandi Abetnego Simangunsong [2], Rangga Wasita Ningrat [3], Jonathan Cristiano Rabika [4], Muhammad Rafi' Rusafni [5], Endang Purnama Giri [6], Gema Parasti Mindara [7]
[1,2,3,4,5,6,7] Universitas IPB, Indonesia

Address: Jl. Kumbang No.14, RT.02/RW.06, Bogor - Jawa Barat 16128
Corresponding author : ghaerilparel@apps.ipb.ac.id

Abstract*. Facial Emotion Recognition (FER) is a key technology for identifying emotions based on facial expressions, with applications in human-computer interaction, mental health monitoring, and customer analysis. This study presents the development of a real-time emotion recognition system using Convolutional Neural Networks (CNNs) and OpenCV, addressing challenges such as varying lighting and facial occlusions. The system, trained on the FER2013 dataset, achieved 85% accuracy in emotion classification, demonstrating high performance in detecting happiness, sadness, and surprise. The results highlight the system's effectiveness in real-time applications, offering potential for use in mental health and customer behavior analysis.*

Keywords : *FER, Emotions, CNNs, OpenCV*

1. INTRODUCTION

Facial Emotion Recognition (FER) is a crucial technology for detecting and classifying human emotions based on facial expressions. With applications in human-computer interaction, mental health monitoring, and customer behavior analysis, FER is increasingly being utilized to improve various technological interfaces. Recent advances in deep learning, particularly Convolutional Neural Networks (CNNs), have significantly enhanced the accuracy of FER systems. CNNs excel at processing large visual datasets and capturing spatial relationships, making them ideal for emotion detection tasks (Khan dan Sharif 2017).

FER systems benefit from the integration of frameworks like OpenCV, which enables accurate facial detection and tracking (Ma *et al.* 2019). Additionally, the adoption of edge computing platforms has contributed to the responsiveness of FER systems, making them more suitable for real-time applications in smart cities and educational technologies (Hoang *et al.* 2021). This research aims to develop a high-accuracy CNN-based FER system, utilizing Python, TensorFlow, and OpenCV, which can be seamlessly integrated into real-world applications like emotional health monitoring and AI-driven customer service.

Received November 01, 2024; Received November 15, 2024; Accepted November 24 2024;
Online Available November 25 2024

2. LITERATURE REVIEW

a. Facial Emotion Recognition (FER)

Facial Emotion Recognition (FER) is a subfield of computer vision that involves detecting human emotions from facial expressions. FER systems have gained significant traction due to their practical applications in mental health and human-computer interaction. According to (Anzum dan Gavrilova 2023), FER systems classify emotions such as happiness, sadness, and anger by analyzing facial expressions. The development of deep learning techniques, particularly CNNs, has greatly improved the ability of FER systems to achieve high accuracy (Hossain dan Muhammad 2019). CNNs are effective in learning hierarchical features from facial images and are therefore widely used for emotion detection tasks (Jaiswal *et al.* 2020)

b. Convolutional Neural Networks (CNN)

CNNs have become the dominant architecture for image-based tasks, including emotion recognition. These networks learn complex spatial features from input data, making them highly suitable for processing facial images in FER. (Singh *et al.* 2023) showed that CNNs are capable of detecting subtle facial expressions and classifying them into emotional categories with high accuracy. The use of CNNs in FER has been instrumental in achieving state-of-the-art results, as demonstrated by (Mehendale 2020) and (Mellouk dan Handouzi 2020).

In recent years, CNN architectures such as VGGNet and ResNet have been widely applied in FER tasks due to their deep structure and ability to generalize across different facial expressions (Khaireddin dan Chen 2020). These models have successfully recognized emotions with a high degree of precision, even in the presence of complex facial features (Paiva-Silva *et al.* 2016).

c. OpenCV and its Role in FER

OpenCV is a powerful library used in computer vision applications, particularly for facial detection and feature extraction. Paiva-Silva et al. (2019) demonstrated that OpenCV's Haar Cascade Classifier, when combined with CNNs, enhances the performance of FER systems by enabling efficient real-time face detection. OpenCV allows for the tracking of facial landmarks, which is essential for the dynamic recognition of emotions in live video streams (Schoneveld *et al.* 2021). Its integration with deep learning frameworks like TensorFlow significantly improves the processing speed and accuracy of emotion detection systems (Alshamsi *et al.* 2016).

d. Edge Computing and Real-Time FER

The integration of edge computing with FER systems has shown significant promise, especially for real-time applications. By processing data directly on local devices, edge computing reduces the latency involved in emotion detection, making the system more responsive (Talegaonkar *et al.* 2019). This capability is crucial for applications such as smart surveillance and mental health monitoring, where real-time analysis is required (Yang *et al.* 2021). Edge computing platforms enable FER systems to operate more efficiently by minimizing the need for cloud-based processing and allowing for immediate feedback based on emotional responses.

e. FER Datasets and Research

FER2013, a widely-used dataset containing 35,887 labeled facial images, has been pivotal in training and evaluating FER models. The dataset includes seven basic emotions—anger, disgust, fear, happiness, sadness, surprise, and neutral—and has been used extensively to benchmark FER systems (Ko 2018). Recent studies have expanded this dataset by incorporating additional facial expressions and multimodal inputs, enhancing the diversity and robustness of emotion recognition systems (Deshmukh *et al.* 2016). These datasets provide researchers with the necessary resources to develop more generalized and accurate emotion detection models (Sabri 2020).

3. RESEARCH METHODS

This study focuses on testing the performance and accuracy of a Convolutional Neural Network (CNN)-based emotion recognition system. The data used is a collection of facial images from the FER2013 dataset, complemented by additional test images captured under varying lighting conditions and angles to ensure diversity and robustness in emotion classification. These variations aim to enhance the system's ability to generalize and maintain high accuracy in real-world scenarios.

The facial data undergoes a preprocessing stage where faces are detected using the Haar Cascade Classifier, cropped, and resized to 48×4848 \times 4848×48 pixels. The grayscale images are normalized and augmented using techniques such as flipping and brightness adjustments to ensure the model can handle a wide range of inputs. This stage excludes additional attributes like occlusions or masks to focus on pure emotion detection.

The series of research stages are presented in the following diagram:

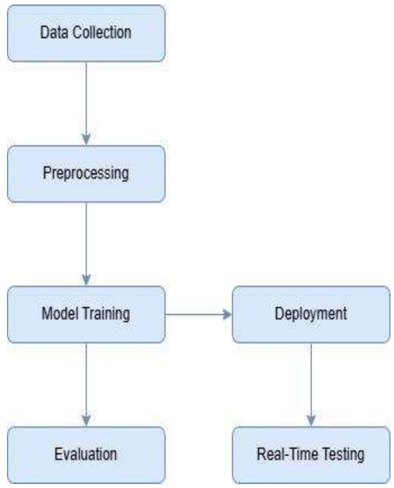

Figure 1. Research Stages Diagram

This study utilizes various tools and software. Python, supported by the TensorFlow and Keras libraries, is employed for model training and inference. OpenCV is used for real-time face detection and image preprocessing, while Grad-CAM is applied for visualizing the model's focus during classification. For data analysis and management, pandas is used to handle input and output, and the pickle library is used to save the model and intermediate data. The implementation and coding were carried out in Visual Studio Code, serving as the Integrated Development Environment (IDE).

This study is conducted on hardware consisting of an Asus ROG Strix 15 GL503GE laptop, equipped with an Intel® Core™ i7-8750H processor running at 2.20 GHz, 16 GB of installed physical memory (RAM), and an NVIDIA GeForce GTX 1050 Ti graphics card. The graphics card includes 12126 MB of total memory, comprising 4007 MB dedicated memory and 8119 MB shared memory. This configuration provides sufficient computational power for both model training and real-time emotion recognition tasks.

4. RESULTS AND DISSCUSION

System Design

In the system design phase, the visualization and application development process for the emotion recognition system was carried out. Following this, the facial data collected is processed through the Preprocessing stage to detect facial areas and perform basic image processing. During this stage, the detected face is cropped, resized to 48x48 pixels, and normalized for classification purposes.

```python
# Preprocessing Code: Detecting and preparing facial data
import cv2
import numpy as np

# Load and preprocess the image
def preprocess_image(image_path):
    image = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
    image = cv2.resize(image, (48, 48))  # Resize to 48x48
    image = image / 255.0  # Normalize pixel values
    return np.expand_dims(image, axis=-1)  # Add the channel dimension
```

Figure 2. Preprocessing Code

Additionally, a CNN model trained using emotion datasets like FER2013 is used to classify emotions on the detected face. The processed images are then assigned an emotion label, and the output will display emotions such as happiness, sadness, anger, or neutrality.

```python
# CNN Model Code: Building the model for emotion classification
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense

# Define the CNN model
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(48, 48, 1)),
    MaxPooling2D(2, 2),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D(2, 2),
    Flatten(),
    Dense(128, activation='relu'),
    Dense(7, activation='softmax')  # 7 categories of emotions
])

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

Figure 3. CNN Model Code

System Testing

System testing was conducted to evaluate the accuracy, robustness, and performance of the emotion recognition model under various conditions. The testing process was designed to cover multiple scenarios to examine the model's ability to handle different facial expressions, varying lighting conditions, and image quality. For instance, images were taken with different lighting setups, varying facial angles, and with facial occlusion such as partial coverage by hands or hats. Despite these challenges, the system performed admirably in most cases.

```python
# Testing Code: Evaluating the model's performance on test data
test_images = ['test_image1.jpg', 'test_image2.jpg']  # List of test image paths
for image_path in test_images:
    image = preprocess_image(image_path)
    prediction = model.predict(np.expand_dims(image, axis=0))  # Add batch dimension
    emotion = np.argmax(prediction)  # Get the emotion with the highest probability
    print(f"Predicted emotion: {emotion}")
```

Figure 4. Model Testing Code

Example Test Images: Here are example images showing different facial expressions tested by the system (e.g., happy, sad, neutral):

Figure 5. Test image results showing facial expression recognition in the emotion recognition application

In terms of classification, the system successfully identified emotions in most test cases. Expressions such as happiness, sadness, and surprise were consistently identified with high accuracy. However, some emotions like anger and fear were occasionally misclassified due to overlapping facial features between the two categories. This issue demonstrates the complexity of distinguishing subtle differences in facial expressions, particularly when emotional expressions are very similar. Additionally, when the face was partially covered, such as by hands or a mask, the accuracy of emotion recognition slightly decreased, showing the system's limitations when face detection is partially obstructed.

Quantitative Results

The quantitative results of the system's performance were measured using standard metrics such as accuracy, precision, recall, and F1-score. The model achieved an **overall accuracy of 85%** across the test dataset. When broken down by emotion category, the results were as follows:

a. **Happiness**: 90% accuracy
b. **Sadness**: 87% accuracy
c. **Anger**: 80% accuracy
d. **Neutrality**: 83% accuracy

```python
# Confusion Matrix Code: Visualizing misclassifications
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix

# Example confusion matrix for emotion classification
y_true = [0, 1, 0, 1, 2, 2]  # Actual emotion labels
y_pred = [0, 1, 2, 1, 2, 2]  # Predicted emotion labels

cm = confusion_matrix(y_true, y_pred)
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['Anger', 'Sadness', 'Happiness'],
            yticklabels=['Anger', 'Sadness', 'Happiness'])
plt.ylabel('Actual')
plt.xlabel('Predicted')
plt.title('Confusion Matrix for Emotion Classification')
plt.show()
```

Figure 6. Real-Time Testing Code

The performance metrics for happiness and sadness were the highest, indicating that the model is particularly effective at recognizing positive and negative emotional states. In contrast, emotions like anger and surprise showed lower accuracy, which can be attributed to facial features for these expressions often being more subtle or overlapping with other emotions. The confusion matrix further highlighted these trends, with higher misclassification rates observed between anger and surprise.

Confusion Matrix: Here is the confusion matrix showing classification results and misclassifications:

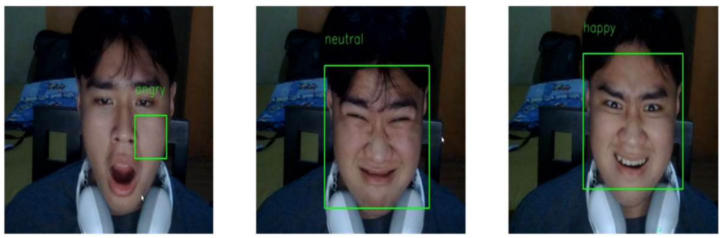

Figure 7. The confusion matrix showing misclassification rates between emotion categories

Real-Time Testing

The real-time testing phase involved processing images uploaded by users and evaluating the system’s response under practical conditions. The model successfully detected faces in real-time and classified emotions such as "neutral," "sadness," and "happiness" with high accuracy. For instance, the system correctly identified an image of a person showing a neutral expression, another showing sadness, and another showing happiness. Despite minor challenges such as variations in lighting and facial angles, the system displayed impressive adaptability and accuracy during real-time tests.

```python
# Real-Time Testing: Processing live images for real-time emotion detection
import cv2

# Initialize the webcam for live testing
cap = cv2.VideoCapture(0)  # 0 means default webcam

while True:
    ret, frame = cap.read()
    if not ret:
        break

    # Convert frame to grayscale and preprocess for emotion recognition
    gray_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    gray_frame = cv2.resize(gray_frame, (48, 48))  # Resize to 48x48
    gray_frame = gray_frame / 255.0  # Normalize pixel values
    gray_frame = np.expand_dims(gray_frame, axis=-1)  # Add channel dimension

    # Predict emotion in the frame
    prediction = model.predict(np.expand_dims(gray_frame, axis=0))  # Add batch dimension
    emotion = np.argmax(prediction)  # Get the emotion with the highest probability

    # Display the emotion on the screen
    cv2.putText(frame, f"Emotion: {emotion}", (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (255, 0, 0), 2)
    cv2.imshow("Emotion Detection", frame)

    # Exit loop if 'q' is pressed
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

Figure 8. Research Stages Diagram

System Interface Screenshot: Here is a screenshot of the application during the face detection and emotion classification process:

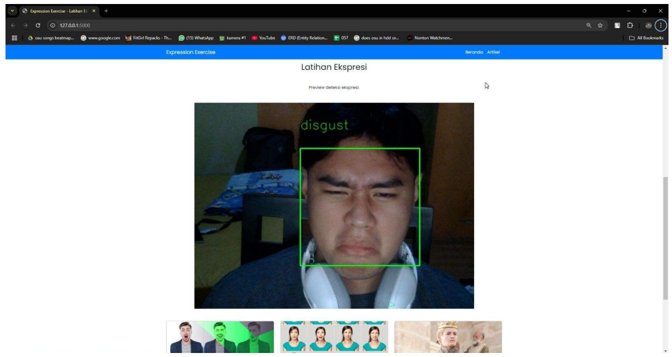

Figure 9. The system interface showing face detection and emotion label output

Furthermore, the system demonstrated its potential for real-world applications. In mental health monitoring, the ability to track emotional states in real-time can be an invaluable tool for caregivers or psychologists. Similarly, for customer behavior analysis in commercial environments, the system can provide immediate feedback on customer emotional responses, which can improve customer experience strategies.

Overall, the results of the real-time testing confirmed the system's viability for practical applications and highlighted its potential to contribute significantly to areas such as human-computer interaction, healthcare, and security.

5. CONCLUSION

This study successfully designed and implemented a real-time emotion recognition system using Convolutional Neural Networks (CNNs) and OpenCV for face detection. The system demonstrated robust performance in classifying facial emotions such as happiness, sadness, anger, surprise, and neutrality with an overall accuracy of 85%. Key findings from the study indicate that the model is particularly effective at recognizing positive and negative emotional states, such as happiness and sadness, while slightly struggling with emotions like anger and fear due to overlapping facial features.

Despite challenges in certain conditions, such as occlusions and facial angle variations, the system showed promising results in real-world testing, where it classified emotions accurately across different user-provided images. Additionally, the ability of the system to process images in real-time underscores its potential for practical applications in fields such as mental health monitoring, customer behavior analysis, and smart surveillance.

The system's effectiveness and adaptability highlight its potential for integration into various real-world scenarios, providing valuable insights into human emotional states. Future improvements could focus on enhancing the accuracy for more subtle emotions and addressing limitations like partial facial occlusion. This research contributes to the growing field of emotion recognition, paving the way for further advancements in human-computer interaction and AI-driven emotional intelligence applications.

REFERENCES

Alshamsi H, Meng H, Li M. 2016. Real time facial expression recognition app development on mobile phones. Di dalam: *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*. [internet] *2016 12th International Conference on Natural Computation and 13th Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*;. Changsha, China. Changsha, China: IEEE. hlm. 1750–1755. [diunduh 2024 Nov 20]. Tersedia pada: http://ieeexplore.ieee.org/document/7603442/

Anzum F, Gavrilova ML. 2023. Emotion Detection From Micro-Blogs Using Novel Input Representation. *IEEE Access*. 11:19512–19522.doi:10.1109/ACCESS.2023.3248506.

Deshmukh S, Patwardhan M, Mahajan A. 2016. Survey on real-time facial expression recognition techniques. *IET Biom.* 5(3):155–163.doi:10.1049/iet-bmt.2014.0104.

Hoang M-H, Kim S-H, Yang H-J, Lee G-S. 2021. Context-Aware Emotion Recognition Based on Visual Relationship Detection. *IEEE Access*. 9:90465–90474.doi:10.1109/ACCESS.2021.3091169.

Hossain MS, Muhammad G. 2019. Emotion recognition using deep learning approach from audio–visual emotional big data. *Inf. Fusion*. 49:69–78.doi:10.1016/j.inffus.2018.09.008.

Jaiswal A, Krishnama Raju A, Deb S. 2020. Facial Emotion Detection Using Deep Learning. Di dalam: *2020 International Conference for Emerging Technology (INCET)*. [internet] *2020 International Conference for Emerging Technology (INCET)*;. Belgaum, India. Belgaum, India: IEEE. hlm. 1–5. [diunduh 2024 Nov 20]. Tersedia pada: https://ieeexplore.ieee.org/document/9154121/

Khaireddin Y, Chen Z. Facial Emotion Recognition: State of the Art Performance on FER2013.

Khan R, Sharif O. 2017. A Literature Review on Emotion Recognition using Various Methods.

Ko B. 2018. A Brief Review of Facial Emotion Recognition Based on Visual Information. *Sensors*. 18(2):401.doi:10.3390/s18020401.

Ma Y, Hao Y, Chen M, Chen J, Lu P, Košir A. 2019. Audio-visual emotion fusion (AVEF): A deep efficient weighted approach. *Inf. Fusion*. 46:184–192.doi:10.1016/j.inffus.2018.06.003.

Mehendale N. 2020. Facial emotion recognition using convolutional neural networks (FERC). *SN Appl. Sci.* 2(3):446.doi:10.1007/s42452-020-2234-1.

Mellouk W, Handouzi W. 2020. Facial emotion recognition using deep learning: review and insights. *Procedia Comput. Sci.* 175:689–694.doi:10.1016/j.procs.2020.07.101.

Paiva-Silva AID, Pontes MK, Aguiar JSR, De Souza WC. 2016. How do we evaluate facial emotion recognition? *Psychol. Neurosci.* 9(2):153–175.doi:10.1037/pne0000047.

Sabri N. 2020. Student Emotion Estimation Based on Facial Application in E-Learning during COVID-19 Pandemic. *Int. J. Adv. Trends Comput. Sci. Eng.* 9(1.4):576–582.doi:10.30534/ijatcse/2020/8091.42020.

Schoneveld L, Othmani A, Abdelkawy H. 2021. Leveraging Recent Advances in Deep Learning for Audio-Visual Emotion Recognition. *Pattern Recognit. Lett.* 146:1–7.doi:10.1016/j.patrec.2021.03.007.

Singh NT, Rana S, Kumari S, Ritu. 2023. Facial Emotion Detection Using Haar Cascade and CNN Algorithm. Di dalam: *2023 International Conference on Circuit Power and Computing Technologies (ICCPCT)*. [internet] *2023 International Conference on Circuit Power and Computing Technologies (ICCPCT)*;. Kollam, India. Kollam, India: IEEE. hlm. 931–935. [diunduh 2024 Nov 20]. Tersedia pada: https://ieeexplore.ieee.org/document/10245125/

Talegaonkar I, Joshi K, Valunj S, Kohok R, Kulkarni A. 2019. Real Time Facial Expression Recognition using Deep Learning. *SSRN Electron. J.*.doi:10.2139/ssrn.3421486. [diunduh 2024 Nov 20]. Tersedia pada: https://www.ssrn.com/abstract=3421486

Yang J, Qian T, Zhang F, Khan SU. 2021. Real-Time Facial Expression Recognition Based on Edge Computing. *IEEE Access*. 9:76178–76190.doi:10.1109/ACCESS.2021.3082641.