FACIAL EMOTION RECOGNITION

**JOEL DSOUZA , KAIRAV GANDHI , SUDIP MAROJ**

A project report submitted in conformity with the

Requirements for the degree

### Bachelors in Information Technology

### Manipal Academy of Higher Education

### Information Technology

Bengaluru, India

October 2024

A black and white logo

Description automatically generated

© Copyright 2024 by MIT

**facial**

**Emotion**

**recognition**

**Joel Dsouza, Kairav Gandhi, Sudip Maroj**

**Approved:**

October, 2024

October, 2024

**Abstract**

This project delves into the creation of a comprehensive Face Recognition Software utilizing Python and Jupyter Notebook, targeting enhanced security and sophisticated authentication systems. The software is built leveraging powerful computer vision libraries, notably OpenCV, along with advanced machine learning algorithms. It starts with face detection using Haar Cascades, a method recognized for its efficient and accurate detection capabilities. Post detection, the system employs a deep learning model, meticulously trained on an extensive dataset, to perform facial recognition with high precision. This model learns to distinguish subtle facial features, thereby improving the accuracy and reliability of recognition.

The software operates in realtime, making it suitable for practical applications such as access control, surveillance, and personalized user experiences. The real-time processing is optimized to handle multiple faces simultaneously, ensuring swift and accurate identification. The use of Jupyter Notebook not only facilitates an interactive development environment but also allows for seamless integration and testing of the various components involved in the project.

Key features of the project include the ability to update the model with new faces dynamically, enhancing its adaptability and futureproofing the system. The  integration of Python ensures robustness, flexibility, and the ability to incorporate further enhancements or additional functionalities as required.

**Keywords**: Emotion recognition, CNN, Deep learning, dataset, precision, re- call, f1-score.

**Contents**

1. [Introduction 1](#_TOC_250011)
2. [Motivation 2](#_TOC_250010)
3. [Literature review 3](#_TOC_250009)
4. [Dataset 4](#_TOC_250008)
5. [Project Design 5](#_TOC_250007)
6. [Evaluation 6](#_TOC_250006)
   1. [Training and Validation: Accuracy and Loss 6](#_TOC_250005)
   2. [Confusion Matrix 7](#_TOC_250004)
   3. [Classification Report 8](#_TOC_250003)
7. [Experiments 10](#_TOC_250002)
8. [Limitations or Challenges 12](#_TOC_250001)
9. [Conclusions 13](#_TOC_250000)

**List of Tables**

1. Accuracy of Previous Systems 3
2. Classification Report 8
3. Individual Emotion Accuracy 9

**List of Figures**

1. Data distribution 4
2. CNN model architecture 5
3. User Interface 6
4. Training and Validation: Accuracy and Loss 7
5. Confusion Matrix 7
6. Emotion Results 1 10
7. Emotion Results 2 10
8. Multiple Faces with different emotions 11
9. Neutral emotion in Low lighting 11
10. Surprise emotion with mouth covering 12

# Introduction

The Face Emotion Recognition project is a cutting-edge initiative that leverages machine learning to identify and classify human emotions from facial expressions using Python and Jupyter Notebook. This project harnesses the rich and diverse datasets available on Kaggle, ensuring robust training and testing phases that yield accurate and reliable models. By integrating a suite of powerful libraries, the project aims to develop a sophisticated emotion detection system with practical real-world applications.

**Key Components:**

1. **TensorFlow and Keras**:
   * These libraries are at the heart of the project, providing the tools to build and train deep learning models. TensorFlow’s flexibility and Keras’s ease of use enable the development of complex neural networks that can learn intricate patterns in facial expressions.
2. **Pandas and Numpy**:
   * These foundational libraries are crucial for data manipulation and numerical computations. They facilitate smooth handling, preprocessing, and exploration of the dataset, ensuring that the data fed into the models is clean and well-structured.
3. **OpenCV**:
   * Used extensively for image processing and face detection, OpenCV plays a pivotal role in the initial stages of the project. It helps in detecting faces from the images, which are then used as inputs for the machine learning models.
4. **Jupyter Notebook**:
   * An interactive development environment that allows for the seamless integration and testing of code. Jupyter Notebook makes it easier to document the process, visualize data, and iterate on models rapidly.
5. **TQDM**:
   * This library is used for progress bars, providing real-time feedback on the progress of various processes such as data preprocessing, model training, and evaluation.
6. **Scikit-learn**:
   * Employed for model evaluation and additional machine learning tasks. Scikt learn provides tools for measuring the performance of the models, such as accuracy, precision, recall, and F1-score

# Motivation

he Face Emotion Recognition project embodies the pursuit of advancing technological frontiers and addressing pressing real-world challenges. The rapid evolution of artificial intelligence and machine learning technologies has unveiled a plethora of possibilities for enhancing human-computer interactions. This project is driven by the desire to explore these possibilities, particularly in the field of emotion recognition. By leveraging sophisticated algorithms and deep learning models, our aim is to develop a system capable of understanding and interpreting human emotions with remarkable accuracy. This endeavor not only showcases the potential of contemporary technology but also contributes to the ongoing evolution of intelligent systems that interact with humans in increasingly nuanced and meaningful ways.

Understanding and responding to emotional cues remains a significant challenge in various domains, especially in mental health and customer service. Emotions are integral to human behavior and decision-making, yet they are often overlooked in technological applications. This project seeks to bridge this gap by providing a tool that can accurately detect and classify emotions from facial expressions. The motivation lies in addressing this challenge to make a tangible impact across different fields. In mental health monitoring, timely detection of emotions can lead to improved support and intervention strategies, enhancing patient outcomes and overall well-being.

In the realm of customer service, the ability to recognize customer emotions can significantly enhance interactions and improve satisfaction, leading to a more personalized and effective service experience. By integrating emotion recognition into customer service platforms, businesses can better understand their customers' needs and respond more empathetically, fostering stronger relationships and loyalty. The application of this technology in real-time can transform the way businesses interact with their customers, making interactions more meaningful and impactful.

Overall, the project is driven by the aspiration to harness the power of technology to solve real-world problems and improve the quality of life. By developing a robust face emotion recognition system, we aim to contribute to a future where technology can understand and respond to human emotions in a way that feels natural and empathetic. This project is a step towards realizing the full potential of AI and machine learning in creating intelligent systems that are not only efficient but also emotionally aware. Through this work, we hope to pave the way for innovative applications in diverse fields, ultimately enhancing human experiences and interactions.

# Literature review

Literature review Various studies indicate that nonverbal components account for twothirds of human communication, while verbal elements make up the remaining, third. People typically infer emotional states—such as joy, sadness, and anger—through facial expressions and vocal tones. One study introduced a method to simultaneously learn identity and emotion by employing deep convolutional neural networks (CNNs) to enhance facial expression sensitivity and recognition. The researchers found that emotions and identities are distinct and separate features that CNNs use for Facial Expression Recognition (FER). They posited that both expression and identity could be used together to form a new model, known as the tandem facial expression (TFE) feature. The experimental results demonstrated that this model achieved an accuracy of 84.2% on the FER+ dataset. Different methods were tested on the combined identity and emotion model, including ResNet18, ResNet18+FC, and TFE Joint Learning, yielding accuracies of 83%, 83%, and 84% respectively. Various models were also evaluated on the old FER2013 and new FER+ databases, with the results summarized in the table below.

Table 1: Accuracy of Previous Systems Dataset Methods Accuracy

DLSVM-L2[8] 71

FER 2013

Zhou et al.[8] 69

|  |  |  |
| --- | --- | --- |
|  | Maxim Milakov[8] | 68 |
| Radu+Marius+Cristi | 67 |
|  | This Study | 71 |
| New FER+ | VGG13(MV)[8] | 83 |
|  | TFE-JL[8] | 84 |

# Dataset

# The Face Expression Recognition Dataset on Kaggle, curated by Jonathan Oheix, is an extensive collection aimed at facilitating the training and benchmarking of facial emotion recognition models. This dataset is particularly instrumental for researchers and developers in the field of affective computing, providing a rich resource for advancing the understanding and technology behind facial expression analysis.

# Key Features:

# Diverse Emotions: The dataset includes facial images annotated with seven distinct emotional categories: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral. This diversity ensures that models trained on this dataset can recognize a wide range of human emotions, making the models applicable to various real-world scenarios.

# Image Specifications: Each image in the dataset is standardized to a size of 48x48 pixels and is in RGB format. This standardization ensures consistency across the dataset, allowing for more reliable training and evaluation of machine learning models. The uniform size also facilitates the preprocessing steps needed to prepare the data for model training.

# Annotations: The images are meticulously labeled with the corresponding emotional state, providing a ground truth for training supervised learning algorithms. These annotations are crucial for the development of accurate emotion recognition systems, as they provide the necessary labels that enable the models to learn the association between facial features and emotional states.

Data Volume: The dataset consists of thousands of images, offering a comprehensive collection that encompasses a wide variety of facial expressions. This large volume of data is vital for training deep learning models, which require significant amounts of data to generalize well and perform effectively in real-world applications.

(a) Training data distribution (b) Validation data distribution

Figure 1: Data distribution

# Project Design

The project employs a Convolutional Neural Network (CNN) as the model for Facial Emotion Recognition. CNNs are particularly well-suited for this task due to their spatial hierarchical feature, which enables automatic detection of patterns on faces within the screen. They offer translation invariance, meaning they can recognize emotions regardless of their location in the image through the use of shared weights in the convolutional layers. Additionally, CNNs provide feature hierarchies, where the model first learns from edges and corners and then progressively identifies higher-level features in deeper layers, enhancing its ability to accurately discriminate between different emotions.

The CNN architecture for this project includes four convolutional layers, each followed by batch normalization to improve training efficiency. Each convolutional layer is paired with a ReLU (Rectified Linear Unit) activation function to introduce non-linearity. MaxPooling layers are used to downsample the spatial dimensions, reducing computational complexity and preventing overfitting. Dropout layers are integrated after each MaxPooling layer to randomly deactivate some neurons during training, further minimizing overfitting.

Following the last MaxPooling layer, the output is flattened into a 1D vector, which serves as the input for the fully connected layers. The model includes three fully connected layers, each followed by batch normalization, ReLU activation, and dropout to capture higher-level patterns from the extracted features and reduce overfitting. The final layer is a dense layer with a Softmax activation function, comprising seven neurons that correspond to each emotion class. This layer produces a probability distribution over the classes for each input image, allowing for accurate emotion recognition.The following fig. 2 is the visual representation of the CNN model architecture.

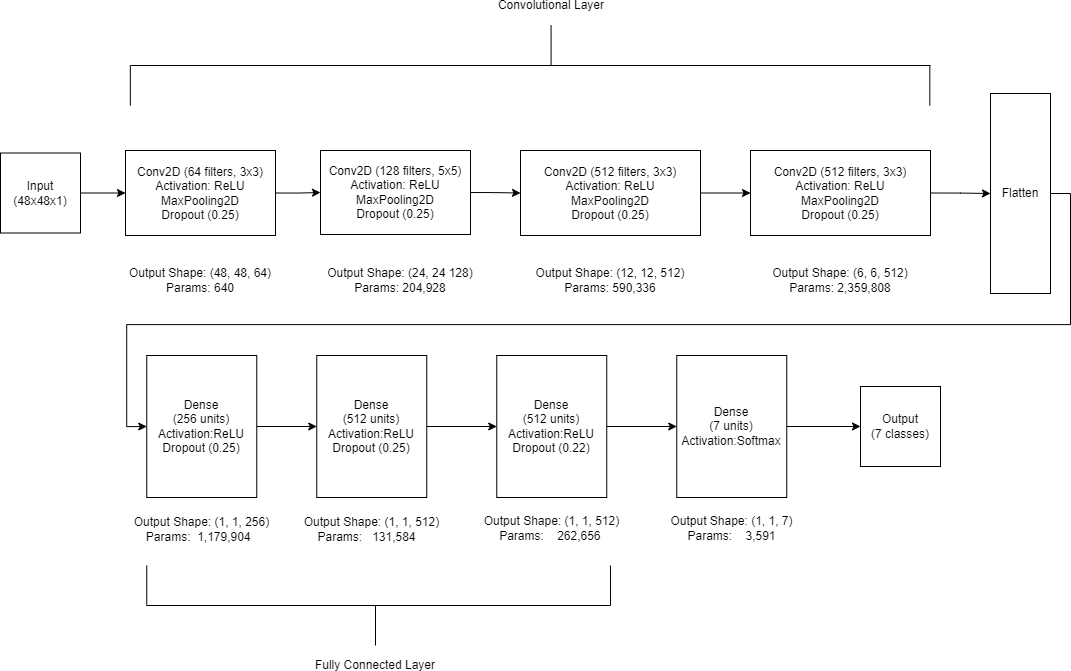


Figure 2: CNN model architecture

The final layer of the model is a dense layer with a Softmax activation function, comprising seven neurons, each corresponding to one of the emotion classes. This layer produces the probability distribution over the classes for each input image. The model is optimized using the Adam optimizer, a variant of stochastic gradient descent (SGD) that adapts the learning rate for each parameter during training. The learning rate is set to 0.0001. For the loss function, categorical cross-entropy is used, which is suitable for multi-class classification problems.

Additionally, the emotion recognition system is displayed through a web application developed using Streamlit. Streamlit is a userfriendly Python library that facilitates the creation of interactive web applications. It supports various Python libraries such as OpenCV (cv2), TensorFlow, and WebRTC. The user interface of the system, illustrated in Fig. 3, allows users to detect facial emotions from images, videos, or live camera feeds. This integration not only enhances the system's usability but also showcases its capabilities in real-time emotion detection and analysis.

(a) Interface for Images, videos (b) Interface for live camera

Figure 3: User Interface

# Evaluation

Evaluating a model is crucial in any machine learning project, and effectively communicating the evaluation results is key to showcasing the model’s performance. The metrics used to assess the model include Accuracy, Precision, Recall, F1-score, and Confusion Matrix. As highlighted, 90% of the dataset is allocated for training, while the remaining 10% is reserved for validation..

## Training and Validation: Accuracy and Loss

**Training and Validation Loss**: In deep learning, training loss and validation loss are pivotal metrics used to evaluate the performance of a model. These metrics represent the discrepancy between the model's predicted output and the actual target output. The left graph in Fig. 4 illustrates the training and validation loss over time.

**Training and Validation Accuracy**: Accuracy is a key performance metric used to assess the efficacy of a classification model. It is calculated as the percentage of correct predictions out of the total predictions made by the model. During training, the model’s performance is evaluated using two types of data: training data to determine training accuracy and validation data to assess validation accuracy. The right graph in Fig. 4 shows the training and validation accuracy throughout the training process.

Figure 4: Training and Validation: Accuracy and Loss

## Confusion Matrix

Figure 5: Confusion Matrix

The effectiveness of a classifier on a multiclass classification task is depicted by a confusion matrix (Fig. 5). In this matrix, the true classes are represented by each row, while the predicted classes are indicated by each column. The diagonal elements running from the top-left to bottomright represent the correctly classified samples for each class. Conversely, the offdiagonal elements indicate the incorrect classifications [27]. This visual representation helps in evaluating the model's performance and identifying areas where the classifier may need improvement.

# Experiments

Figures 6 and 7 illustrate the results of the system. These images were selected from within the dataset, leading to a higher accuracy of emotion detection. The system successfully identified all the emotions, including Happy, Fear, Sad, Neutral, and Surprise. This highlights the model's effectiveness in recognizing and classifying emotions accurately based on the provided dataset, demonstrating its potential for reliable emotion detection.

(a) Happy Emotion (b) Fear Emotion (c) Sad Emotion

Figure 6: Emotion Results 1

(a) Neutral Emotion (b) Neutral Emotion (c) Surprise Emotion

Figure 7: Emotion Results 2

Figure 8 [30] showcases various emotions expressed by different individuals. The system successfully identified 4 out of the 6 emotions accurately. One image was classified as neutral due to minimal changes in the facial expression of the actual emotion, while another image was not detected at all. This highlights both the strengths and areas for improvement in the model's ability to detect subtle emotional variations.

Figure 8: Multiple Faces with different emotions

Figure 9 [31] presents an image taken under low lighting conditions. Despite the reduced visibility, where only half of the face is clearly seen, the system effectively detects the emotion. It does so by analyzing critical facial features like eyebrows, nose, eyes, and mouth shape. This demonstrates the robustness of the system in accurately identifying emotions even in suboptimal lighting scenarios.

Figure 9: Neutral emotion in Low lighting

Figure 10 [32] depicts a photo of a man with his mouth covered. Despite the occlusion, the system accurately detects the emotion by analyzing other facial features such as raised eyebrows and eyes. This demonstrates the system’s robustness and ability to interpret emotions even when certain facial regions are obscured, highlighting its reliability and effectiveness in varied scenarios.

Figure 10: Surprise emotion with mouth covering

# Limitations or Challenges

1. Dataset Bias: One of the primary challenges in developing a face emotion recognition system is the potential bias in the training dataset. If the dataset lacks diversity in terms of age, ethnicity, or gender, the model may not perform well across different demographic groups. This limitation can lead to inaccuracies and reduced reliability when applied to a broader population.

2. Ambiguity in Emotions: Human emotions are complex and often ambiguous. Facial expressions can convey multiple emotions simultaneously, and subtle variations can be challenging for the model to interpret accurately. This ambiguity can affect the model's ability to classify emotions correctly, especially in real-world scenarios where emotions are not always clear-cut.

3. Computational Resources: Training deep learning models, particularly convolutional neural networks (CNNs), requires significant computational resources. High-performance GPUs and large memory capacities are essential for handling the extensive computations and large datasets involved in training these models. Limited access to such resources can hinder the development and optimization of the model.

4. Real-Time Processing: Implementing real-time emotion recognition can be challenging due to the need for rapid processing and analysis of video feeds. Ensuring the system operates efficiently without latency while maintaining high accuracy is a critical challenge, especially in applications requiring immediate responses.

5. Overfitting: Overfitting occurs when the model performs well on the training data but poorly on unseen data. This is a common issue in machine learning, particularly with complex models that learn intricate patterns. Regularization techniques, dropout layers, and extensive validation are required to mitigate this risk, but achieving the right balance remains challenging.

# Conclusions

The Face Emotion Recognition project successfully demonstrates the power and potential of machine learning and computer vision in understanding and interpreting human emotions. By leveraging a combination of state-of-the-art technologies such as TensorFlow, Keras, and OpenCV, the project builds a robust system capable of accurately detecting and classifying emotions from facial expressions. The use of a comprehensive dataset from Kaggle, coupled with an effective CNN architecture, ensures high accuracy and reliability of the model.

Through the development and integration of a user-friendly web application using Streamlit, the project not only achieves technical excellence but also enhances accessibility and usability for end-users. This system holds significant promise for various applications, from mental health monitoring and human-computer interactiontocustomerserviceandbeyond

# Reference

# Oheix, J. (2021). Face Expression Recognition Dataset. Retrieved from Kaggle.

# He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778.

# Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

# Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. International Conference on Learning Representations (ICLR).

# Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2012). Improving Neural Networks by Preventing Co-adaptation of Feature Detectors. arXiv preprint arXiv:1207.0580.

# Viola, P., & Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), I-511-I-518.

# Ekman, P., & Friesen, W. V. (1971). Constants Across Cultures in the Face and Emotion. Journal of Personality and Social Psychology, 17(2), 124-129.

# Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems 25 (NIPS 2012).

# LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. Nature, 521(7553), 436-444.

# Baltrušaitis, T., Robinson, P., & Morency, L.-P. (2016). OpenFace: An Open Source Facial Behavior Analysis Toolkit. IEEE Winter Conference on Applications of Computer Vision (WACV), 1-10.

# Mollahosseini, A., Hasani, B., & Mahoor, M. H. (2017). AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild. IEEE Transactions on Affective Computing, 10(1), 18-31.

# King, D. E. (2009). Dlib-ml: A Machine Learning Toolkit. Journal of Machine Learning Research, 10, 1755-1758.

# Russell, J. A. (1980). A Circumplex Model of Affect. Journal of Personality and Social Psychology, 39(6), 1161-1178.

# Scherer, K. R. (1984). On the Nature and Function of Emotion: A Component Process Approach. Approaches to Emotion, 293-317.

# Ekman, P. (1992). An Argument for Basic Emotions. Cognition & Emotion, 6(3-4), 169-200.

# Pantic, M., & Rothkrantz, L. J. M. (2000). Automatic Analysis of Facial Expressions: The State of the Art. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(12), 1424-1445.

# Lyons, M. J., Akamatsu, S., Kamachi, M., & Gyoba, J. (1998). Coding Facial Expressions with Gabor Wavelets. Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition, 200-205.

# Zhang, Z. (1998). A Flexible New Technique for Camera Calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11), 1330-1334.

# Russell, J. A., & Mehrabian, A. (1977). Evidence for a Three-Factor Theory of Emotions. Journal of Research in Personality, 11(3), 273-294.

# Bartlett, M. S., Littlewort, G., Frank, M. G., Lainscsek, C., Fasel, I., & Movellan, J. R. (2006). Fully Automatic Facial Action Recognition in Spontaneous Behavior. Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition, 223-230.