**Antarang**

A PROJECT REPORT

BY

**Shlok Bhardwaj (E22CSE0041)**

**Vrinda Parmar (E22CSE0043)**

**Saksham Dhiman (E22CSE0048)**

**Mehak (E22CSEU1682)**

A close up of a sign

Description automatically generated

**SUBMITTED TO**

**SCHOOL OF COMPUTER SCIENCE ENGINEERING AND TECHNOLOGY, BENNETT UNIVERSITY**

GREATER NOIDA, 201310, UTTAR PRADESH, INDIA

**November 2024**

**Title:** Antarang (Facial Emotion Detection using CNN)

**Abstract**

Mental health plays a big role in overall well-being, but a lot of people struggle to get support they need due to the lack of resources and awareness. This project aims to create an easy-to-use website that provides emotional support by recognizing users' emotions and offering personalized advice/responses based on how they are feeling and their age. The ultimate goal is to help users manage their emotions better and feel supported.

Our system uses a custom-trained Convolutional Neural Network (CNN) model built with TensorFlow, which is trained on the FER 2013 dataset, to identify emotions from facial images captured through the user’s webcam. Users also input their age, which, along with their detected emotion, is used to generate advice according to the situation. For example, a happy teenager may be encouraged to share their happiness with others, while a sad adult might be advised to connect with friends or focus on positive memories.

The platform is built on Flask for the backend and HTML, CSS, and JavaScript for the frontend, for a smooth and interactive user experience. The design is flexible, that is it allows for future improvements like support for multiple languages or better emotion detection.

This project closes the gap in accessible mental health applications by combining technology with empathy. It gives users simple, and meaningful advice to help them manage their emotions and promotes overall mental health in the society.

1. **Introduction**

This section provides an overview of the project, including its background, the challenges to be addressed, and the objectives to be achieved. It also outlines the scope of the project.

* 1. Background and Motivation: Mental health is an important part of a person’s overall health. However, many people face challenges in knowing their emotional needs due to a lack of awareness, or limited access to professional help. In today’s digital age, technology offers multiple ways to support mental health. The idea for this project comes from the need to create an application that can make emotional support more accessible and personalized. By combining emotion recognition with age-specific advice, this project aims to provide a simple yet an effective solution for people to manage their emotions and so they don’t feel alone in their mental health journey.
  2. Problem Statement: Many individuals struggle with dealing and understanding their emotional needs but lack accessible and immediate support systems. Existing solutions fail to provide personalized help, ignoring the user’s emotional state or age. This creates a gap in mental health support, where people feel isolated or uncertain about how to deal with their emotions. There is a need for an easy-to-use platform that can offer tailored advice based on both the user’s emotions and age, helping them feel understood and guide them better.

1.3 Objectives: The main objectives of this project are:

* To develop a web application that recognizes emotions using facial images using a custom-trained CNN model.
* To provide personalized advice/responses to users based on their detected emotions and age.
* To create a user-friendly interface that lures individuals towards using the application for emotional support.
* To design a scalable system for future improvements, such as new emotions, multiple languages, or advanced advice systems.

1.4 Scope of the Project: This project focuses on creating an emotional support system that is both functional and accessible. The key features include:

* Real-time emotion detection from facial images using a CNN model.
* Age-based advice generation tailored to specific emotions.
* A simple and engaging web interface developed using Flask, HTML, CSS, and JavaScript.
* Predefined responses/advice for emotions such as happiness, sadness, and anger, customized for various age groups.

**2. Literature Review**

2.1 Existing Solutions/Studies

* There are a handful of tools available that can detect emotions by analyzing facial expressions. For example, some applications use pre-trained models based on the FER 2013 dataset. These tools are mainly used for mood tracking or emotion recognition but they often stop at detection.
* Popular mental health applications like Calm and BetterHelp provide support through guided meditations or therapist connections. However, they don’t include features like real-time emotion recognition and advice generation.
* Some websites do offer generalized advice or links to mental health resources, but these aren’t according to the user's specific emotions or age.

2.2 Gaps in Existing Research

* Many existing applications fail to give advice that is personalized to both the user’s emotion and their age, making them less effective.
* Accessibility remains a problem since many applications require subscriptions or are only available in certain locations, limiting their reach.
* The stigma associated with seeking mental health help in our society often discourages people from using such applications that fully focus on mental health.
* Most systems offer general solutions and don’t provide advice that feels specific to the user’s emotional or life context.

2.3 Relevance of the Proposed Work

* The application we are developing fills this gap by combining real-time emotion detection with advice that is tailored to the user’s age and emotional state. This makes the advice more meaningful and helpful.
* By designing a web application, we make sure it’s easy to access and doesn’t require paid subscriptions or specialized devices.
* Our approach promotes mental wellness in a simple and friendly way, motivating the users to seek help without feeling judged or stigmatized.
  1. **System Design and Architecture**

3.1 System Overview

The system is designed as a web application that provides the users with personalized mental health support by detecting their emotions from a captured facial image and considering their age input. The application combines image processing, machine learning, and pre-created responses/advices to give tailored guidance. The overall architecture promotes real-time interaction, and scalability to ensure a solid and user-friendly experience.

The system works as follows:

1. The user opens the web application and captures an image using their webcam, that is after providing the necessary permissions and inputs their age.
2. The image is analyzed using a custom-trained CNN model to identify the user's emotional state.
3. The detected emotion and age are mapped to a database of predefined advice.
4. The advice is displayed to the user via the web interface in an intuitive format.

3.2 Functional Requirements

* Emotion Detection: The system detects emotions from images captured through the webcam with decent accuracy.
* Age Input Handling: System processes the age input to give tailored responses.
* Response/Advice Generation: Application provides empathetic and accurate responses based on the emotion detected and the age group of the user.
* Real-Time Processing: The system processes and provides responses in real-time for a seamless experience.
* Web Interface: The web interface should be easy to use and interactive for the users to engage with.

3.3 Technology Stack Used

* Front-End: HTML, CSS, JavaScript for the user interface.
* Back-End: Flask (Python) for handling detection and integration.
* Machine Learning: TensorFlow and Keras for building and training the model.
* Database: SQLite (SQL-alchemy) for storing the credentials.
* Additional Tools: OpenCV for image preprocessing, Kaggle for model training and fine-tuning.

**4. Methodology**

The methodology to develop the facial emotion detection application involves systematic design, training, and implementation of the custom CNN model, as well as making the web application. This section portrays the dataset used, model, and the algorithm at play along with the entire development process, tools and platforms used.

4.1 Dataset Description

The dataset used for the system is the FER2013 (FER stands for Facial Emotion Recognition), which is a very popular dataset for the task at hand. It contains approximately 35887 grayscale images of size 48x48 categorized into seven emotion classes: Anger, Sad, Happy, Neutral, Surprised, Disgusted and Fear.

Challenges: The dataset contains super noisy images, as well as misclassified images along with several categories being underrepresented which doesn’t allow for a perfect predicting model.

4.2 Model/Algorithm Used

The system makes use of a custom trained Convolutional Neural Network (CNN) model for emotion detection.

Architecture:

* Input Layer: Accepts grayscale images of 48x48 size.
* Convolutional Layers: Extracts features using kernels with ReLU activation function for non-linearity.
* Pooling Layers: To reduce dimensions and complexity.
* Fully Connected Layers: Maps the learnt features to the emotions.
* Output Layer: Uses SoftMax activation to output probabilities of each class.

Training:

* Optimizer: Adam with a learning rate of 0.0001.
* Loss Function: Categorical Crossentropy for multi-class classification.
* Regularization: Dropout layers to prevent overfitting.
* Epochs: Model was trained for 50 epochs to achieve decent accuracy and optimal convergence.

4.3 Development Process

The project followed an iterative development path for continuous additions and consistent workflow:

1. Dataset Selection: The FER2013 dataset was chosen out of multiple options available due to the amount of diversity in images as compared to datasets like the CK+ or the Facial Emotion Recognition dataset on Kaggle.
2. Model Training: Developed and trained the model on dataset while improving it with time based on metrics such as accuracy and loss.
3. System Integration: Integrated the trained model with the web application by Flask while also developing the frontend subsequently.
4. Testing and Deployment: Functional testing to check how well the application predicts the users’ emotion.

4.4 Tools and Platforms Used

Development Tools:

* Python: Model training, backend development, system integration.
* Kaggle: Dataset selection, model training, fine tuning.
* Flask: Server-side connection between model and frontend.

Machine Learning Frameworks:

* TensorFlow: Foundation for building and training the model.
* Keras: Simple implementation of the architecture.

Web Development:

* HTML, CSS, JavaScript for designing the interface along with the image capture mechanism.

The above-mentioned methodology assured a well-rounded approach towards achieving the objectives while providing scope for future improvements.

**5. Implementation**

5.1 Module Description

The project consists of distinct modules where each one of them is intended to handle a specific functionality of the system. The modules operate together tenaciously to maintain an effective and smooth experience for the user. Description of the project modules is given below:

Web interface module

Our website is designed using HTML, CSS, and JavaScript. It serves as a primary means of user interaction with our service. Along with being a user-friendly the interface is also minimal and provides a clean look, allowing the users to:

* Register or log into their account.
* Allow webcam access for capturing their face.
* Input their age for tailored response.
* Receive personalized advice based on their mood and age.

Emotion Detection Module

After capturing user’s image, process of evaluating their emotional state takes place which is carried out by this module. This is done using a custom-trained Convolutional Neural Network (CNN) model. About the model:

* The mode is trained on FER 2013 dataset, Includes 7 labels for high accuracy in real world situations.
* The image is classified as one of the predefined motions such as happy, sad, angry, disgust, etc.
* Other than ReLU, SoftMax activation function is used at the output layer to calculate probability for each emotion.
* The model consists total of twenty-three layers with six convolutional layers and 2 dense layers.
* There are total of 5869767 trainable parameters.

Advice Generation Module

This module is responsible for determining the personalized output for the user. It combines the age and predicted emotion to generate recommendations. This module:

* Selects advice that is empathetic and appropriate for user’s age, also ensuring respect towards their emotional state.
* Provides the advice to the user through the web interface in an easy-to-understand format.
* Uses a pre-defined database of advice which is categorized by emotions and age groups.

Back-End Processing Module

Flask framework is used for back-end module. It is a python-based framework which is easy to implement. It acts as the bridge between front-end and the model. It:

* Simulates the flow of data from user interface to the backend i.e. CNN model and vice versa.
* Retrieves the appropriate advice from the database based on the user input.
* Ensures smooth user experience with real-tie processing.

Database Module

This module is responsible for storing user’s login credentials and advice library which is categorized by age group and emotions.

* Ensures privacy and stability for any future updates.
* Provides an efficient and organized way to map inputs to their respective outputs.

Together, these modules form an integrated system that captures the user’s emotional state and age, processes user’s image through the model, predicts emotional state and delivers meaningful advice tailored to individual needs.

5.2 Code Structure/Workflow

The code component of emotional health support system is organized in such a way that it ensures clarity and maintainability of the project. Workflow of the project includes an end-to-end process that captures the image from user interface, processes it through CNN model and generates tailored responses.

Code Structure

Main Application (main.py):

* It serves as the entry point for the connection with website.
* It handles routes to connect front-end with back-end logic in Flask.
* Integrates emotion detection module with advice generation module.

Model Training (Model Training.ipynb):

* Kaggle notebook is used to train and evaluate the CNN model.
* Preprocessing of input image, model architecture definition, and performance evaluation are included under this section.
* Generates the trained model file (model.h5) used in prediction.

Pre-Trained Model (model.h5):

* Includes the CNN model trained on FER 2013 dataset for emotion classification.
* Positioneed in the application for real-time emotion detection.

Front-End (website/):

* Consists of HTML, CSS, JavaScript files which are responsible for user interface.
* Includes code to capture images using webcam and to send it for processing at the backend.

Database (database.db):

* Stores login credentials of the users.
* Contains structured library containing responses based on age and detected emotions.

This structured approach ensures precise integration of all the components together for a better understanding of workflow as well as smooth function of the application.

5.3 Features Implemented

The system makes use of multiple features to provide an efficient application for users seeking mental health support. It detects emotions from facial images and takes age as an input to give personalized responses to the user based on the previously stated factors. This ensures the system is functional and user centric, thus providing real-time assistance.

The primary objective of the system is to recognize emotions through a custom trained CNN model, developed using TensorFlow and trained on the FER 2013 dataset, can classify multiple emotions such as happiness, sadness, anger, with decent accuracy. The facial image is captured using the webcam, age is taken as input through a text box in the interface, the model determines the user’s emotional state, which are both crucial factors in generating relevant advice.

The responses/advice are pre-created by the team, after sufficient research on mental health and psychological topics. This feature is carefully categorized by emotions and age groups, to provide empathetic and appropriate responses to the users. The addition of user age input further refines the responses, thus making them more impactful and relatable.

Our application provides the users with real-time interaction capabilities, by capturing their image and giving them advice immediately. This enhances the user experience, since the system is quick and efficient. The frontend which is developed using HTML, CSS and JavaScript gives a clean and engaging interface, thus making sure that users can easily navigate through the site. The backend developed using the framework Flask, connects the frontend to the CNN model flawlessly ensuring smooth workflow.

Security and privacy are maintained throughout the system, as the captured images and age inputs are all processed in real time and are not stored, complying with ethical standards. This builds trust with users and maintains the integrity of the application.

In summary, the need of providing mental health support to the user has been tackled by cohesive approach of implemented features in the project. The platform provides user-friendly interface along with personalized experience, this is achieved by combining artificial intelligence for emotion detection, age-appropriate personalized data and real time interaction.

**6. Results and Analysis**

6.1 Experimental Setup

The experimental setup involves model training, experimenting with layers of convolutional neural network, adjusting the hyperparameters, evaluation of the model, and testing the end-to-end functionality of the system including the connectivity of frontend with model. Key components are:

Training Environment: Model was trained on Kaggle Notebook using libraries like TensorFlow, Python, and Keras. The model was trained on FER 2013 dataset, containing over 22000 images labelled as happy, angry, disgust etc. Learning rate was set to the rate of 0.001, batch size was around 32, and 20 epochs for optimal training of the model.

Testing Environment: The system was tested on both local and hosted servers; Real time webcam image capture was tested across different devices to ensure proper working. Model was also tested on unseen data to ensure correct prediction.

Hardware and Software: Intel i7 processor, 24GB RAM, and GPU support for model training. Kaggle Notebook for training and flask for integration.

Dataset: FER 2013 dataset was divides into 80% for training, 20% for validation.

6.2 Test Cases and Outcomes

Different test cases were carried out to evaluate system’s functionality like accuracy of emotion prediction by the model, advice generation, user interaction, and integration. Observations are listed below:

The system predicted correct emotion from real-time webcam images in 68% cases during cases. Performance was consistent across each emotion label such as happiness, sadness, and anger. The advice generated matched the detected emotion and user provided age every time with empathetic responses. After capturing the image system processed and responded in about 1.2 seconds on average hence ensuring smooth user experience.

Edge Case Handling: Images with unclear facial expressions or with some kind of distortions resulted in a fallback in turn returning a default response based on the advice generation mechanism. Overall, the system demonstrated robustness but also required further tuning for real world scenarios.

6.5 Evaluation

Accuracy and Loss Graphs

A screenshot of a computer

Description automatically generated

(evaluation of model on unseen data)

Frontend

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

(Testing of integrated system)

**7. Discussion**

7.1 Insights and Interpretation of Results

The results achieved in the project depict the viability of using a machine learning-based system to provide accurate emotion predictions. The custom-trained CNN model showed accuracy of about 68% while testing process which is the crucial component of the system.

Emotion detection mechanism when combined with personalized advice generation system gives effective and tailored outputs enhancing the user experience by delivering meaningful and empathetic guidance. User feedback while testing ensured that advice was relevant, appropriate to their age and tailored according to user needs which in turn created a safe environment for user to be expressive. System’s real-time processing and minimal interface contributed to a positive user experience.

7.2 Limitations of the Current Work

Despite a remarkable performance of system, it has some limitation that must be addressed and improved for broader applicability and improved performance:

1. Age Input Handling: The system uses manual input for age which is a factor for personalizing the user experience but, it adds an additional step for the user. Automating this process could improve the user experience.
2. Dataset Dependencies: The model is trained using FER 2013 dataset, which is large but lacks the representation of diversity in real-world facial expressions. Along with that it also lacks representation of certain ethnicity hence creating bias in the model prediction. This can lead to reduced accuracy when dealing with expressions that differ from examples in the dataset.
3. Limited emotion Categories: The system only detects limited emotions which do not cover the diversity in human emotions. For example, Mixed emotions remain a challenge for the system.
4. Edge Case Performance: Circumstances like blurry images, low lighting, obscured faces, mixed emotions can result in low accuracy and wrong prediction.
5. Advice System: While the advice system is accurate and to the point, but it is using a static database of predefined responses. Advice that is tuned to the context of user’s situation and is dynamic could enhance the experience further.

7.3 Future Scope

While the model has a strong foundation and an effective system to provide mental health support, but there is significant potential for future improvements and expansions.

Additional dataset that captures wide range of ethnicity, demographics and facial expressions can be incorporated for an improvement in accuracy. An expansion in emotion categories to include complex stated and mixed emotions would result in more detailed and enhanced performance of the model. Fine tuning can further enhance the model’s accuracy. Integrating an automated age detection feature possibly through image, can make the experience better and process simpler for the user.

Transitioning from a static database to a Dynamic database that uses natural language processing (NLP) to generate personalized advice in real time could increase the relevance of the outputs provided. Expanding the platform to other devices like mobile phones and incorporating support for multiple languages could make the application more accessible. Linking the system with online services like counselling or hotlines will provide customers with additional support along with the generated advice.

**8. Conclusion**

8.1 Summary of Work

The project uses a machine learning model for facial emotion recognition along with manually entered age to provides personalized support that is appropriate and empathetic for their needs and life stage. The CNN-based emotion detection model, trained on the FER 2013 dataset, demonstrated accuracy of about 68%. Integrated system of user-friendly web interface with back-end model ensured a smooth and real-time interaction experience. Overall, the system was able to achieve its primary goal of delivering meaningful support in an effective manner.

8.2 Key Contributions

This project has made a significant contribution in the fields of mental health:

* **Emotion Recognition model**: Developed a custom-trained CNN model from square one which is optimized for real time emotion detection.
* **Ensuring data privacy**: By processing the data in real time without storing any sensitive information, the system establishes ethical data handling.
* **Real-time Interaction**: Built a platform that provides users with instant results hence enhancing engagement and user experience.
* **Scalability**: Implemented architecture is flexible hence making it possible to add further enhancements like additional emotions, automated age detection, and multilingual support. Which makes it easier to work on future enhancements.
* **Personalized Advice System**: Created a unique advice generating system that combines and emotional state to provide guidance.

8.3 Concluding Remarks

In conclusion this platform acts as a meaningful step towards contribution in helping and providing guidance to people who are struggling with their mental health challenges. It leverages technology by combining machine learning model, minimal web interface, and user-focused design. While the system achieves its primary goal effectively, there is still a significant amount of scope in betterment of the project which can increase the satisfaction for user by making advice more dynamic by integrating advance techniques like NLP, automating age input, using a diverse dataset for model training etc.

This project shows how crucial it is to incorporate empathy into the development of technological systems, particularly within the field of mental health, this would help in creating a safe environment for the users. Expanding on this system and enhancing its features could have a lasting effect on people looking for emotional support and engaging in mental health initiatives.

**9. References**

1. Barsoum, E., Zhang, C., Ferrer, C. C., & Zhang, Z. (2016). *Training deep networks for facial expression recognition with crowd-sourced label distribution*. Proceedings of the 18th ACM International Conference on Multimodal Interaction.

1. Goodfellow, I., Erhan, D., Carrier, P. L., et al. (2013). *Challenges in representation learning: A report on three machine learning contests*. Neural Networks, 64, 59–63.
2. Lundqvist, D., Flykt, A., & Öhman, A. (1998). *The Karolinska Directed Emotional Faces (KDEF)* [Database of standardized emotional images].
3. Ekman, P., & Friesen, W. V. (1971). *Constants across cultures in the face and emotion*. Journal of Personality and Social Psychology, 17(2), 124–129.
4. Kaggle. (n.d.). *FER2013 Dataset*. Retrieved from <https://www.kaggle.com>.
5. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). *Gradient-based learning applied to document recognition*. Proceedings of the IEEE, 86(11), 2278–2324.
6. Viola, P., & Jones, M. (2001). *Rapid object detection using a boosted cascade of simple features*. Proceedings of IEEE CVPR.
7. Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.
8. TensorFlow. (n.d.). *Documentation and Resources for TensorFlow Framework*. Retrieved from <https://www.tensorflow.org>.
9. Keras. (n.d.). *API Reference for Keras Framework*. Retrieved from <https://keras.io>.
10. Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980.
11. OpenCV. (n.d.). *Open Source Computer Vision Library*. Retrieved from <https://opencv.org>.
12. Calm. (n.d.). *Mental health and mindfulness app*. Retrieved from <https://www.calm.com>.
13. BetterHelp. (n.d.). *Online Therapy Platform*. Retrieved from <https://www.betterhelp.com>.
14. Scherer, K. R., & Ekman, P. (Eds.). (1982). *Handbook of methods in nonverbal behavior research*. Cambridge University Press.
15. Zeng, Z., Pantic, M., Roisman, G. I., & Huang, T. S. (2009). *A survey of affect recognition methods: Audio, visual, and spontaneous expressions*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(1), 39–58.
16. Jain, A., Ambekar, A., & Athavale, V. (2020). *Emotion recognition from facial expressions using CNN*. International Journal of Engineering and Advanced Technology (IJEAT), 9(4), 427–431.
17. Russell, J. A. (1980). *A circumplex model of affect*. Journal of Personality and Social Psychology, 39(6), 1161–1178.
18. He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep residual learning for image recognition*. Proceedings of the IEEE CVPR, 770–778.
19. National Institute of Mental Health. (n.d.). *Mental health statistics and resources*. Retrieved from <https://www.nimh.nih.gov>.