FACIAL EXPRESSION RECOGNITION FOR HUMAN - COMPUTER INTERACTION

PHASE I REPORT

Submitted by

MANNURU SHREEYA - 2116210701148

PRATHEEPA R - 2116210701192

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

NOV 2024

ANNA UNIVERSITY, CHENNAI

BONAFIDE CERTIFICATE

Certified that this Report titled **"FACIAL EXPRESSION RECOGNITION FOR HUMAN COMPUTER INTERACTION"** is the bonafide work of **MANNURU SHREEYA (210701148)** and **PRATHEEPA R (210701192)** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Dr. P. Kumar, M.E., Ph.D.,
Professor and Head
Computer Science and Engineering
Rajalakshmi Engineering College
Chennai – 602 105

Dr. P. Kumar, M.E., Ph.D.,
Professor and Supervisor
Computer Science and Engineering
Rajalakshmi Engineering College
Chennai – 602 105

Submitted to Project Viva Voce Examination held on _____________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ABSTRACT

Facial expressions are very important in understanding one's state of mind. We aim to incorporate this notion in our learning assistant application in order to provide a rich user experience. The proposed system provides information about any topic the user is seeking for, and it also simultaneously captures the facial expressions of the user while the retrieved information is being displayed. The query of the user is matched against all the titles of Wikipedia and the information under the title that achieves the maximum score of similarity with the query is provided. A deep learning model trained using the FER-2013 dataset on a convolutional neural network consisting of residual blocks is used to predict the facial expressions of the user in the frames captured by the camera. The model classifies each frame into any of the seven classes: angry, disgust, sad, surprise, happy, neutral, and fear. The first half of the predictions are given a weightage of 0.5 and the second half of the predictions are given a weightage of 1.0. A score for each class is calculated by performing a cumulative addition of weights and the class with the maximum score is determined to be the predominant expression of the user while reading through the provided information. Based on the predominant expression, the needs of the user are inferred and suitable responses are provided which improves the user experience.

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and shower his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman **Mr. S. MEGANATHAN, B.E., F.I.E.**, our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN, Ph.D.,** for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to **Dr. S. N. MURUGESAN, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. P. KUMAR, M.E., Ph.D.,** Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guide, **Dr. P. KUMAR, M.E., Ph.D.,** Department of Computer Science and Engineering, Rajalakshmi Engineering College for his valuable guidance throughout the course of the project. We are very glad to thank our Project Coordinator, **Dr. T. KUMARAGURUBARAN, Ph.D.,** Department of Computer Science and Engineering for his useful tips during our review to build our project.

MANNURU SHREEYA : 2116210701148

PRATHEEPA R : 2116210701192

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

S No.	Abbreviations	Expansion
1.	CBAM	Convolutional Block Attention Mechanism
2.	VGG	Visual Geometry Group
3.	MTCNN	Multi-task cascaded convolutional networks
4.	FER	Facial Expression Recognition
5.	GAN	Generative Adversarial Network
6.	MLP	Multilayer Perceptron
7.	SVM	Support Vector Machine
8.	RNN	Recurrent Neural Network
9.	LSTM	Long Short-Term Memory
10.	GCNN	Graph Convolutional Neural Network

CHAPTER 1
INTRODUCTION

1.1 GENERAL

In the world of growing technology, online learning is increasing everyday and new technologies are being adopted to improve the learning outcomes for the users. Human-computer interaction has also evolved in various forms in order to provide effective communication between the user and the system that enhances the results needed by the user. This project aims to incorporate both online learning and human-computer interaction so as to improve the learning efficiency and also the user experience. The system provides information about any topic that the user is looking for. It inputs a query from the user in the textual format and retrieves the information that the user is seeking by comparing the query against all the titles of Wikipedia. The title that achieves the maximum score of similarity with the query is chosen and the information under that title is retrieved and provided for the user. While displaying the information, the webcam is switched on and the frames are captured every 2 seconds. A deep learning model trained on a convolutional neural network with residual blocks is used to classify each frame into one of the seven classes: angry, happy, sad, fear, surprise, neutral, and disgust. The processes of providing information and expression recognition are performed simultaneously in order to analyse the level of understanding of the provided content for the user. The predominant expression throughout the reading time is identified and based on that, suitable responses are provided for the user.

1.2 OBJECTIVE

The objective of this project is used to develop a learning assistant that uses facial expression recognition. The system should be able to accurately identify the facial expressions in the captured frames. The required information by the user needs to be correctly identified and retrieved within a short period of time. The system aims to provide the required information for the user and also accurately identify the facial expressions. The right information needs to be provided with low latency. The project aims to improve the user experience and the learning efficiency of the user.

1.3 EXISTING SYSTEM

Existing systems with regard to human-computer interaction involve chatbots that communicate with users in an empathetic manner and identify depression in users to provide timely intervention. Online learning systems include interactive dashboards that shows the learning history of the user along with an analysis of their level of understanding. Based on this, future learning content is tailored according to the needs of the learners. Convolutional neural networks are used to train a deep learning model to identify facial expressions and autoencoders are being used for identifying the most important features in order to reduce the number of feature and the complexity of the architecture of the model. Generative adversarial networks are used to train models to identify facial expressions even in occluded images that contain objects that cover the face like sunglasses, hand etc.

1.4 PROPOSED SYSTEM

The proposed system aims to identify facial expressions in real-time throughout the learning process of a user and dynamically resorts to adapt the learning content in order to improve learning efficiency and user experience. The system inputs a query from the user to match with that title of Wikipedia that has the maximum score of similarity with the query. The information under the matched title is retrieved using the Wikipedia API and provided for the user.

The introduction part of the complete information is extracted and displayed letter by letter with latency so as to provide a sensation of reading for the user. As the information starts getting printed, the webcam is switched on and the frames are captured every 2 seconds. The FER-2013 dataset had only around 400 images for the 'disgust' class against 7200 images for the 'happy' class. This imbalance in the classes would make the model be more biased towards one particular class and might not help in capturing the features of the class that has lesser images. The images in real-time might contain background parts apart from the face, different lighting conditions, and varying contrast. The model must be trained so as to handle all these situations by using a training data that contains a variety of images. Considering all the factors, data augmentation is performed in the form of background addition, adjusting brightness, and adjusting contrast. A deep learning model is trained on this increased dataset using a convolutional neural network with residual blocks. The trained model is used to arrive at the predictions for all the frames captured by the webcam. The first half of the predictions are given a weightage of 0.5 and the second half of the predictions are given a weightage of 1.0. Based on these weights, a score is computed for each class using the cumulative addition of weights and the predominant expressions is identified. Suitable responses are provided for the user after the displayed information using this identified predominant expression.

CHAPTER 2

LITERATURE SURVEY

M. Aly et al. [1] proposed a Convolutional Block Attention Mechanism (CBAM) which was included in the enhanced version of ResNet-50 model. This technique enhances the precision of expression recognition by prioritizing significant face characteristics while excluding irrelevant background data. By analysing the identified facial expressions, the system assesses the learning states of pupils, such as their level of concentration, difficulty, and contribution which enables the system to autonomously adapt teaching methods, such as modifying the level of difficulty of the material or offering further assistance. This system reaches a higher accuracy of 87.62% on RAF-DB dataset and an accuracy of 88.13% for FER2013 dataset in facial expression recognition compared to traditional methods. Challenges in the quality of captured images. Real life scenarios may involve a more complex range of emotions but focuses on predefined set of emotional states. The 10-second interval for monitoring students' facial presence may be too long, as emotions can change much more rapidly. This delay could lead to inaccuracies in assessing the students' real-time emotional states and engagement.

W. E. Villegas-Ch et al. [2] used a pre-trained machine learning model for recognizing facial expressions, trained on labelled datasets to recognize six emotions. A laboratory was set up to capture facial expressions during practical sessions, with images captured in batches of 350 per student. 7000 images were collected, corresponding to six categories: commitment, interested, frustrated, neutral, boring and focused. First, a face is detected in the image and then crucial parts of the image such as eyes, mouth, eyebrows and nose are detected. A convolution neural network is used with 3 layers each of a convolution layer and a layer to capture features that are important. The neural network was trained with an accuracy of 70%. Challenge is that AI systems struggle to understand the context in which an emotion occurs. Without context, these systems might misread the emotions and provide inaccurate results. People can experience multiple emotions at a time which makes it challenging to detect accurate emotions.

G. Zhao et al. [3] proposed a lightweight emotion recognition model based on DenseNet architecture, which uses highly linked convolution layers and model compression methods. Facial recognition is done with HOG features and Support Vector Machine classifier. This approach for face alignment is an ensemble of regression trees. The training process uses the Nesterov momentum selection approach. It reaches a high accuracy of 85.89% using FERFIN dataset with a significantly fewer parameters. Effectiveness of FERFIN dataset is improved by removing noise classes from FERPLUS. Challenge is that there are lower accuracy on some emotion classes due to class imbalance so it is limited to basic category of emotions and is evaluated only on lab collected datasets and not on real world dataset.

J. Liu et al. [4] put forth a system that consists of three stages. They are: the pre-processing of data, the extraction of features, and classification. In data preprocessing stage the data enhancement method is used which improves the contrast of the input data. To obtain more discriminative features from the enhanced data, a hybrid feature representation approach is used. ResNet is used to replace the first part of the VGG network and as a classification layer, SoftMax is used. The proposed model can improve the contrast of the data, obtain more discriminative features and attains a 94.5% recognition rate. Whereas it does not focus on the occlusion in large area of the face.

N. Zhou et al. [5] presented this model for recognizing emotions using Multi-task cascaded convolutional networks detection (MTCNN), trained using the ADAM optimizer. The overall model consists of a neural network comprising of four residual depth wise separation convolutions, which is augmented with the MTCNN detection system to accomplish facial emotion identification. The final layer utilizes the Global Average Pooling layer and softmax activation function for the purpose of classification. This model is trained using the WIDER FACE dataset, which is a well-recognized dataset for face identification. Additionally, the Karolinska Directed Emotional Faces (KDEF) dataset is used due to its low image noise, making it excellent for training the model in expression recognition. By using these datasets, the accuracy achieved on the training set could be around 71%, while the accuracy rate on the validation set can reach around 67%. When it comes to identifying certain types of emotions, the accuracy is still not perfect and a limited dataset is available for a group of emotions.

J. Pu and X. Nie [6] designed a FER algorithm for enabling human-computer interaction. It was designed to give robots the ability to recognize facial expressions like a human. The proposed XRS module is suggested to reduce the number of parameters used in the model and the deterioration of the network. A channel attention mechanism based module SEnet can be used to highlight the important features in an image while concealing the unimportant features. This is done by adjusting the feature map's channel weights. For deploying this model to portable devices, a pruning algorithm is used along with a depth separable convolution filter and Principal component analysis. This approach is said to compress the network to easily deploy it to portable devices. FER2013, CK+ and RAF-BD are the datasets used in this study. The pruning strategy suggested has a good acceleration effect and reduces the amount of memory used by about 41% of the parameters, thus improving classification accuracy. This algorithm has shown about 99% of recognition accuracy along with an average accuracy of 80.39%. It's ability to focus on parallel pruning is a challenge.

P. Ganesan et al. [7] used regenerative Generative Adversarial Network (GAN) to recreate obscured areas of students' faces, essential for precise emotional interpretation in CK+ dataset. The dashboard offers a user-friendly interface for educators, with visualizations associated with different emotions. Teachers can access real-time feedback and historical data monitoring to adjust their teaching methods. The use of GAN ensures that the emotion detection is accurate even when students' faces are partially covered. But the system's accuracy depends heavily on the quality and diversity of the training dataset. In real-world applications, varying lighting conditions and camera angles may affect performance and this system is computationally intensive.

M. D'incà et al. [8] proposed a Convolutional Autoencoder (AE) for recognizing partial occlusions in images caused by objects like sunglasses and hands. It consists of an encoder which has three residual blocks with each block having three convolutions along with a max-pooling operation. Decoder is the transposed version of the encoder, which reconstructs the original image using the latent space. The AE is then used to extract features for training a linear classifier, a Multilayer Perceptron (MLP) consisting of two layers which are trained to perform the classification of the positive and negative emotions. Its focus is only on classifying into positive and negative classes and not discrete emotion classes.

M. Shi et al. [9] used Fuzzy C-Means (FCM) clustering technique on Convolutional Neural Network. The Haar-Like algorithm is used for facial detection. Feature extraction is performed using a trained CNN model, and the retrieved characteristics are inputted into a Support Vector Machine for emotion classification. The use of FCM clustering with CNN demonstrates a recognition accuracy of 83.86%, surpassing that of conventional CNN models. In order to get more accurate classification, the Support Vector Machine algorithm is used instead of the Softmax function. But, FCM clustering is computationally intensive.

E. Uzun [10] introduced a web scraping approach called "UzunExt" that improves the efficiency of extracting content from web pages. UzunExt collects three types of additional information to optimize future extractions: starting position, inner tag count and tag repetition. UzunExt uses string searching techniques to locate the start and end tags of the desired content, achieving faster content extraction than traditional methods. Experiments showed that UzunExt could be about 60 times faster than the best DOM-based parser. It can be adapted to existing DOM-based algorithms to improve their time efficiency without major modifications. But it cannot deal with webpages that change content dynamically.

G. Suddul et al. [11] used a Recurrent Neural Network (RNN) in the form of Long Short-Term Memory Algorithm (LSTM) to predict the most plausible outcome for a query. The query is loaded into a JSON document, tokenized, and converted into numeric configuration. Keras is used for creating a sequential model for a deep Convolutional Neural Network (CNN) trained on the FER 2013 dataset. Haar Cascade Classifier is used for feature extraction. Overall Accuracy is around 85.71%. Facial expression accuracy is around 65%. The model is neither overfitting nor underfitting. The model had been trained for lesser epochs which considerably reduces the training time. Relatively low accuracy for expression detection due to environmental factors. Loss is higher when validation test is done. Only 50% of the expression prediction are accurate.

B. Fang et al. [12] discussed facial expression recognition methods in educational research. Three steps are used in this machine learning-based facial expression recognition framework. They are: face acquisition, the extraction of the features of facial expression, and the classification of facial expressions. The Viola-Jones detector and the histogram of oriented gradient (HOG) detector are few of the conventional face detectors. This system uses 4 machine learning algorithms. They are: Support Vector Machines, K-nearest neighbors, random forest and classification and regression trees, out of which the best accuracy rates were procured by using the KNN and SVM algorithms. The transformer has an encoder-decoder architecture where each component of the data is weighted differently based on its significance. The transformer also has a self-attention mechanism.

A. V. Savchenko et al. [13] proposed a lightweight CNN model for face recognition trained using video stills of faces. Several light weight architectures were trained for facial processing, such as MobileNet v1, EfficientNet-B0, EfficientNet-B2. The model's training process includes pre-training for face recognition and fine-tuning for emotion categorization. The FFmpeg utility was implemented to extract the frame pictures from the video. The models are trained to categorize emotions using the AffectNet dataset and VGAF dataset. When it comes to recognizing emotions in static photos, the suggested CNN model performs better than the findings of the individual models. Students' emotion recognition cannot be supported by any publicly accessible datasets. Over a certain threshold, low-resolution facial expressions are disregarded. Improving the quality of the interaction engine, which is constrained owing to the small training set, requires more video data.

R. Miao et al. [14] proposed a study involving nine video lectures that revealed the students engagement levels are crucial for improving learning outcomes. Participants were asked to press a key when an auditory target was noticed by them, with an average loudness of 70 db and white noise of 0.66 db. Facial features were recorded during the experiment, and it was found that when a person is deviated from the lecture they wrinkled their nose and when the person is attentive frequent blinking, depression in the corners of the lips and the tightening of eyelids was observed. Facial expressions can predict learners' attention levels, with video camera-captured facial features predicting reaction times (RTs), which indicate the attention states of the students.

R. Bhargava et al. [15] proposed a model using Naive Bayes classification that processes the question it receives by using segment embedding and token embedding, while BERT, a pre-trained model, is fine-tuned to respond to user requests using incoming information. Moreover, a text summarizer model is used to augment the user's understanding of the terms. Scraping requires the use of BeautifulSoup (BS4) and Selenium Web Driver. The intent classification model attains an accuracy of around 97%. The latency in responding to a question is minimal. This is only relevant to certain sectors of websites that focus on selling products. The BERT and Summarizer models have a limit of 512 tokens. Websites that frequently change their structure may still pose challenges, requiring regular updates and maintenance of the scraping scripts.

Y. He et al. [16] introduced a multi-layer feature recognition algorithm using a three-channel convolutional neural network (HFT-CNN). The first channel trains on the entire facial image to capture overall muscle movements. The second channel focuses on the eyebrows and eyes region to extract features related to these areas. The third channel concentrates on the mouth region, extracting features from this part of the face. Random horizontal flipping and brightness adjustment were data augmentation techniques applied. CK+ and JAFFE were the datasets used. The softmax layer finalizes the classification by assigning the facial expression to one of the seven predefined categories (fear, disgust, anger, happiness, neutral, surprise and sadness) based on the highest probability. The model has reached an average recognition rate of 94.1% on the JAFFE dataset and 98% on CK+ dataset. But it is less efficient due to the complex structure.

D. -H. Lee and J. -H. Yoo [17] introduced a divide and conquer convolutional neural network (CNN) learning approach to increase the accuracy of facial expression recognition (FER). Face regions are identified, focussed, and normalized to the necessary dimensions. Pre-trained on ImageNet, a ResNet-18 model is adjusted to better accommodate modifications, such as lowering filter sizes and extracting more relevant information from face photos. In order to identify similar facial expressions with high confusion rates, the confusion matrix of the first predictions of the CNN models is analyzed. Facial expressions are classified as subproblems until the model can precisely distinguish between them. This model improved classification accuracy on various datasets by addressing the issue of variability in facial expressions. Increase in computational complexity is observed as it involves retraining the CNN model. Poor grouping can lead to suboptimal performance. May require manual intervention.

Y. Tang et al. [18] introduced FreNet model, a lightweight model based on deep learning that makes use of Discrete Cosine Transform (DCT) to convert images into the frequency domain. This process focuses on capturing low-frequency components by concentrating energy in the upper-left corner of the image, which contain critical information about the image. Learnable Multiplication Kernel (LMK) is trainable and works by multiplying input feature maps element-wise to produce new feature maps. A summarization layer is present that reduces the dimensionality of the data while preserving essential information. The model has reached an accuracy of 98.97% on the CK+ dataset. They are prone to overfitting due to the smaller size of CK+ dataset.

S. Li et al. [19] proposed a methodology that focuses on multi-hop open-domain question answering (QA) by using both structured and unstructured information from Wikipedia. The study presents a new retrieval technique named HopRetriever, intended to collect scattered reasoning evidence from several sources. It involves defining a "hop," which is the combination of a hyperlink and its corresponding outbound content. The system systematically acquires additional documents by traversing a sequence of "hops" between them, enhancing the evidence compilation at each stage. This enables HopRetriever to identify relevant documents that aid in addressing intricate inquiries.

X. Yang et al. [20] proposed a Graph Convolutional Neural Network (GCN) designed for micro-expression recognition. It uses Facial Action Units (FAUs) to capture detailed changes in facial expressions, and the Optical Flow Method (OFM) to estimate pixel motion in facial images. The GCN uses an adjacency matrix based on the co-occurrence of FAUs to model relationships between different facial regions. Facial features are divided into seven key regions: left eye, right eye, nose, lips, left cheek, right cheek, and chin. This division allows for targeted feature extraction from areas most affected by micro-expressions. The model incorporates a self-attention mechanism to focus on the most relevant facial features and reduce interference from irrelevant data. The model was evaluated on two well-known datasets for micro-expression recognition: CASME II and SAMM. On the CASME II dataset, the model reached an accuracy of 79.5% and on the SAMM dataset, the proposed model reached an accuracy of 73.8%. The load of computation for executing this model is high.

REF	METHODOLOGY	DATASETS	CONTRIBUTIONS	CHALLENGES
[1]	CBAM in enhanced version of ResNet	RAF-DB, FER2013	Enhances expression recognition by prioritizing significant features	10 second interval for monitoring expressions is long
[2]	Pre-trained model on labelled dataset	Custom dataset	Collected facial expression images related to learning	AI systems might not understand the context of emotions
[3]	Lightweight model based on DenseNet	FERFIN	Uses highly linked convolution layers and model compression methods	Lower accuracy on some emotion classes
[4]	ResNet and VGGNet	FER2013	Obtained more discriminative features using hybrid representation	Does not focus on large area occlusion
[5]	MTCNN	WIDER FACE	Trained on images with low noise to give higher accuracy	Trained on a limited dataset
[6]	XRS module	FER2013	Reduced the number of parameters and the complexity of the network	Focusing on parallel pruning
[7]	GAN	CK+	Ensures expression recognition even	Does not handle varying lighting

			when faces are partially covered	conditions and camera angles
[8]	Convolutional Autoencoder	FER2013	Recognizing expressions with partial occlusions	Classifies only into positive and negative classes
[9]	Fuzzy C-Means	FER2013	Support Vector Machine is used to get more accurate classification	Computationally intensive
[10]	UzunExt	-	Retrieves content from webpages 60 times faster	Cannot deal with dynamic webpages
[11]	RNN, CNN	FER2013	Answers queries and also recognizes facial expressions	Low accuracy for expression detection
[12]	Study	-	Analysed the various methods used in facial expression recognition	-
[13]	Lightweight CNN	AffectNet	Used a hybrid architecture to recognize emotions well	Small training data
[14]	Predict reaction times of students	-	Inferred that the facial expression of students can be used to understand attention levels	-
[15]	Naïve Bayes	-	Performs effective text summarization	Difficult to consider

				websites that change structure frequently
[16]	HFT-CNN	CK+	Focuses on important regions of the face to recognize expressions	Complex structure
[17]	CNN	FER2013	Adjusted to modifications while retrieving more relevant information	Increased computational complexity
[18]	FreNet	CK+	Reduces the dimensionality of data while preserving important information	Prone to overfitting due to small dataset size
[19]	HopRetriever	-	Answers queries by hopping between pages of Wikipedia	Identifying relevant documents
[20]	GCN	CASME	Recognizes expressions by focusing on important regions of the face	Load of computation is high

Table 2.1. Literature Survey Summary

SUMMARY

The existing literature includes analysis of facial expressions of students while they are solving a practical exercise and while they are listening to lectures. The analysis indicates whether they are comfortable with the learning or not. These results are later used to adapt the difficulty and the content of the learning material. Recognizing facial expressions include the steps of recognizing the face, extracting the important features and classification. Viola-Jones algorithm and the Histogram of Oriented Gradients are popularly used for recognizing the face. Convolutional neural networks are used in terms of convolution layers, filters, pooling layers and fully connected layers along with attention mechanisms to learn the important features. Softmax function is used at the classification layer as it gives a probability distribution suitable for multi-class classification. Some literature has attempted to solve the problem of occlusion in facial expression recognition using autoencoders and generative adversarial networks.

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

3.1.1. SYSTEM FLOW DIAGRAM

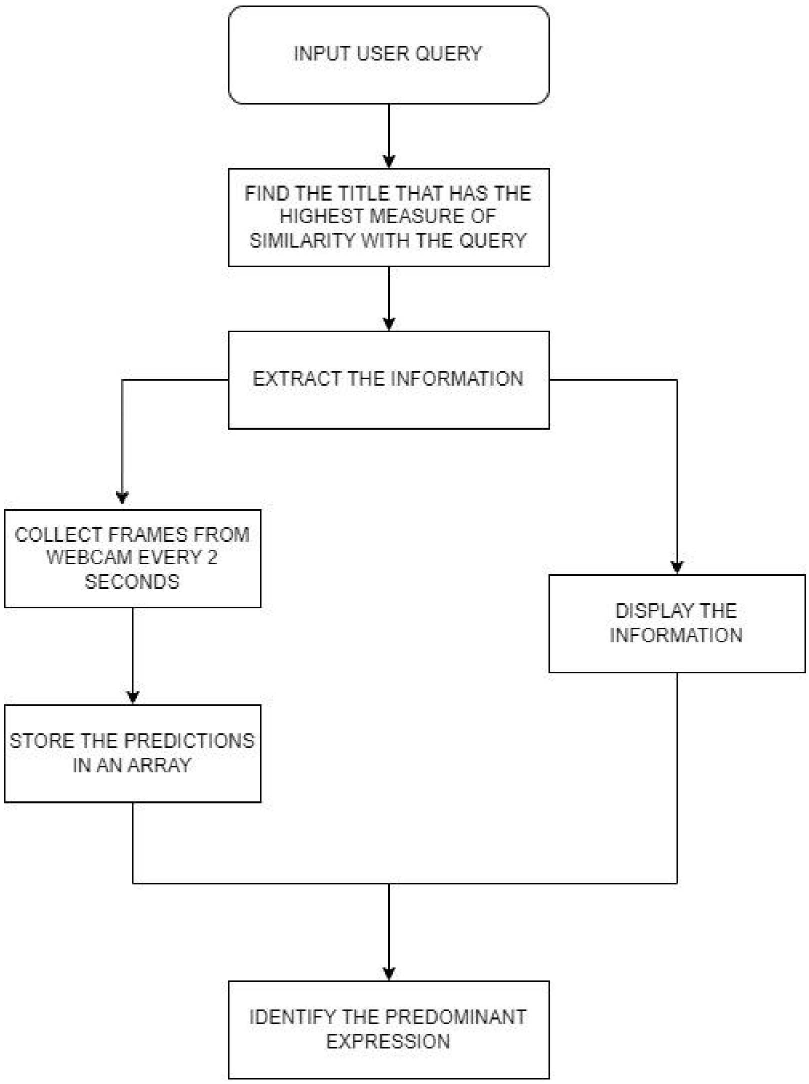

Fig. 3.1 SYSTEM FLOW DIAGRAM

The figure 3.1 denotes the system flow diagram of the proposed system highlighting the flow of processes. Initially, the user query is inputted and the query is compared against all the titles of Wikipedia to identify the title that has the maximum similarity score with the query. The score is computed using a measure of the percentage of words that match in the title from that of the query and vice versa. The information under this title is extracted using the Wikipedia API and displayed for the user. The process of capturing frames and storing the predictions is performed in parallel with the process of displaying information. Finally, the predominant expression is identified to analyze the requirements of the user.

3.1.2. ARCHITECTURE DIAGRAM

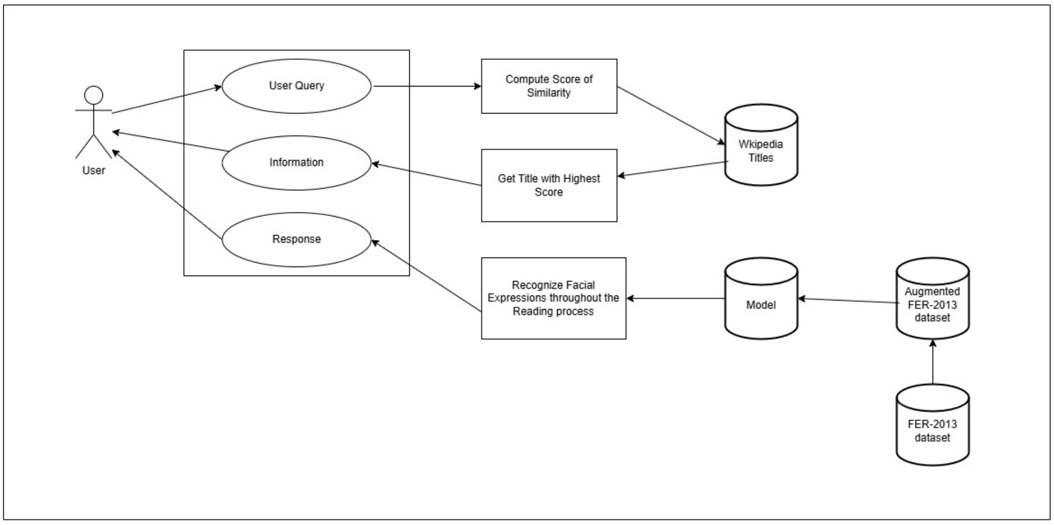

Fig. 3.2 ARCHITECTURE DIAGRAM

The figure 3.2 denotes the architecture diagram of the system. The number of images in the FER-2013 dataset has been increased by performing variations for the existing images in terms of background addition, adjusting brightness, and adjusting contrast. The number of images in the augmented dataset is 50613. A model is trained on this increased dataset with the help of a convolutional neural network with residual blocks. The user interacts with the system in the form of the query, the retrieved information, and the response that is provided based on the predominant expression. The query is compared against all the titles of Wikipedia that are regularly released in the form of dumps. The information under the matched title is retrieved using the Wikipedia API and is presented for the user.

The trained model identifies the facial expressions from the captured frames and the overall predominant expression is decided.

3.1.3 USE CASE DIAGRAM

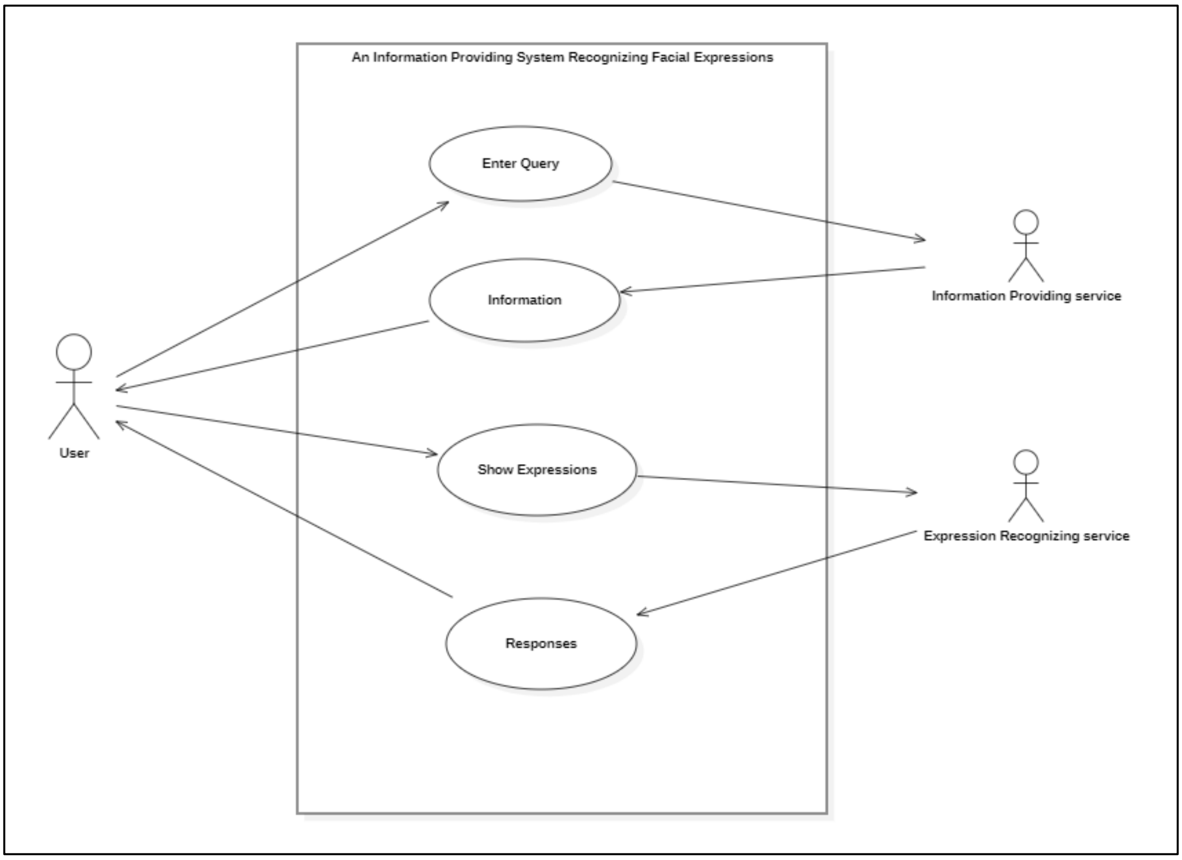

Fig. 3.3 USE CASE DIAGRAM

The figure 3.3 denotes the use case diagram. The user interacts with the system by providing a query that is matched with the most similar title to that of the query in Wikipedia to provide the required information by the information providing service. The information providing service performs the role of matching the query with the title by computing a score of similarity and provides the information. The expression providing service captures the frames collected by the webcam to identify the facial expressions, identify the predominant expression, and display a response for the user based on the predominant expression.

3.1.4. ACTIVITY DIAGRAM

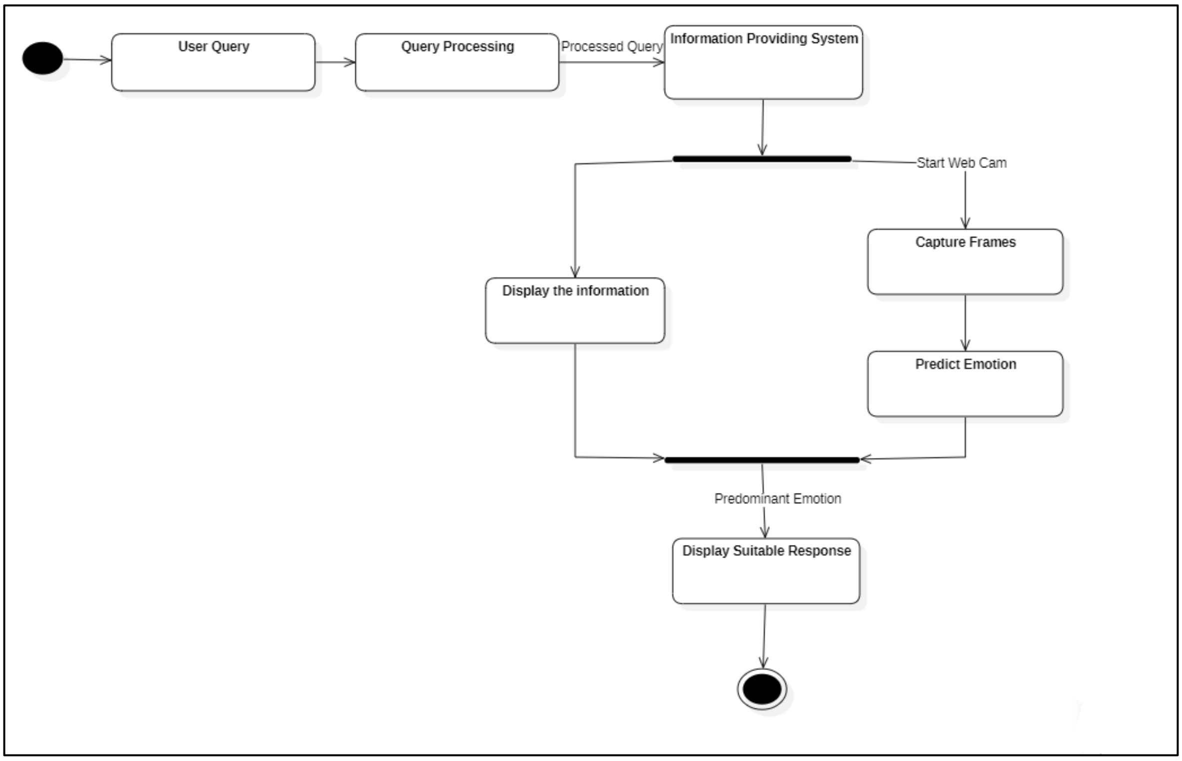

Fig. 3.4 ACTIVITY DIAGRAM

The figure 3.4 denotes the activity diagram which indicates the activities and the flow of activities performed in the proposed system. The query is inputted by the user which is processed to remove the special characters and convert it to a list of words. It is then provided to the information providing system to identify the title of Wikipedia that has the maximum similarity with it. Then, the information of that title is retrieved and displayed for the user. Parallelly, the webcam is switched on and the frames are captured. The trained model is used to predict the facial expressions and then the predominant expression is identified by performing a cumulative addition of the weights assigned to each prediction.

3.1.5. CLASS DIAGRAM

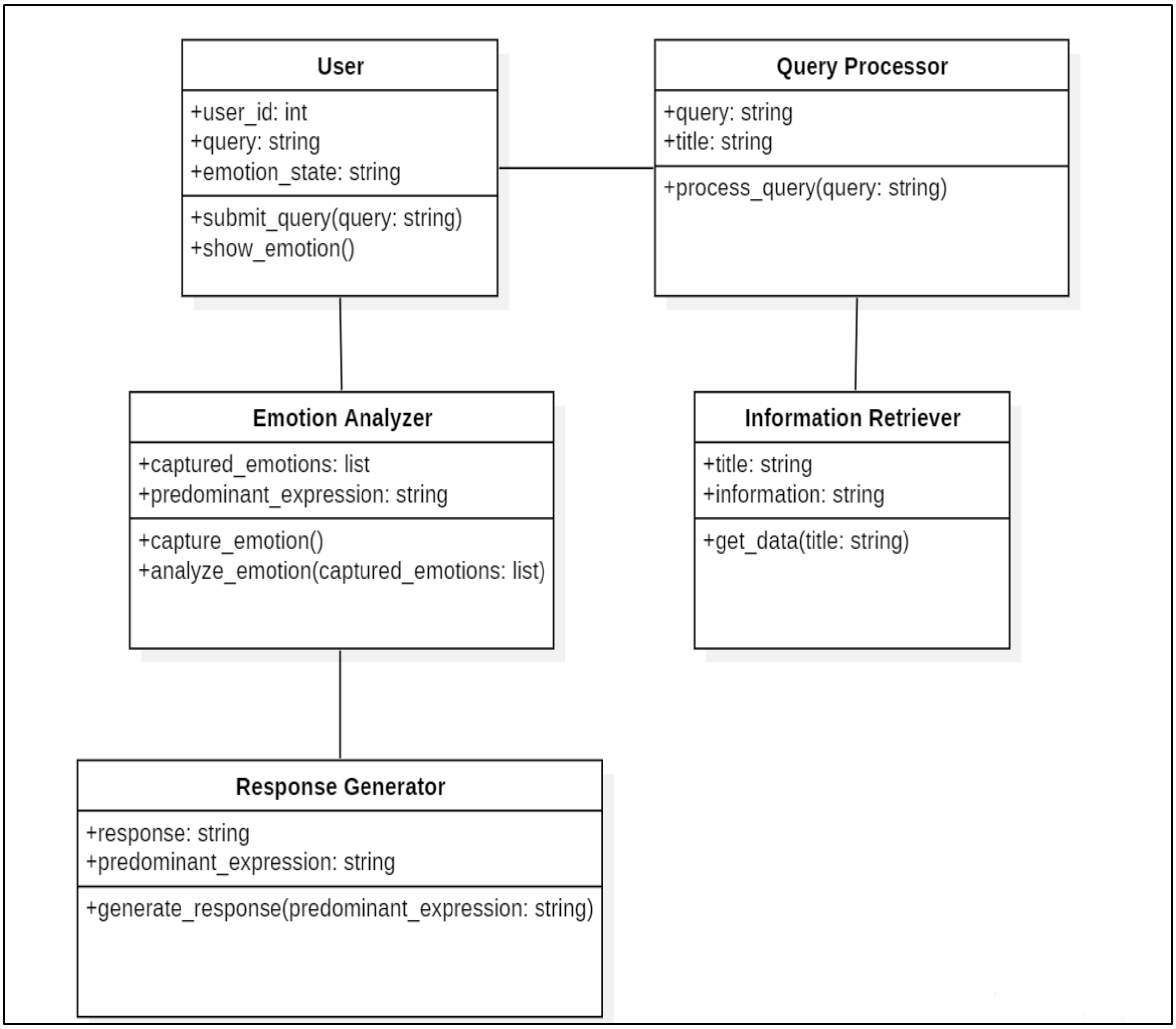

Fig. 3.5 CLASS DIAGRAM

The figure 3.5 denotes the class diagram of the proposed system. The user class interacts with the query processor and the emotion analyzer. The query processor removes the special characters from the query and converts it to a list of words. Then the title that has the maximum score of similarity with the query is identified. The information retriever gets the title from the query processor and retrieves the needed information. The emotion analyzer gets the emotion states of the users and identifies the predominant expression. The response generator class gets the predominant expression and displays a suitable response for the user so as to improve the user experience.

3.1.6. SEQUENCE DIAGRAM

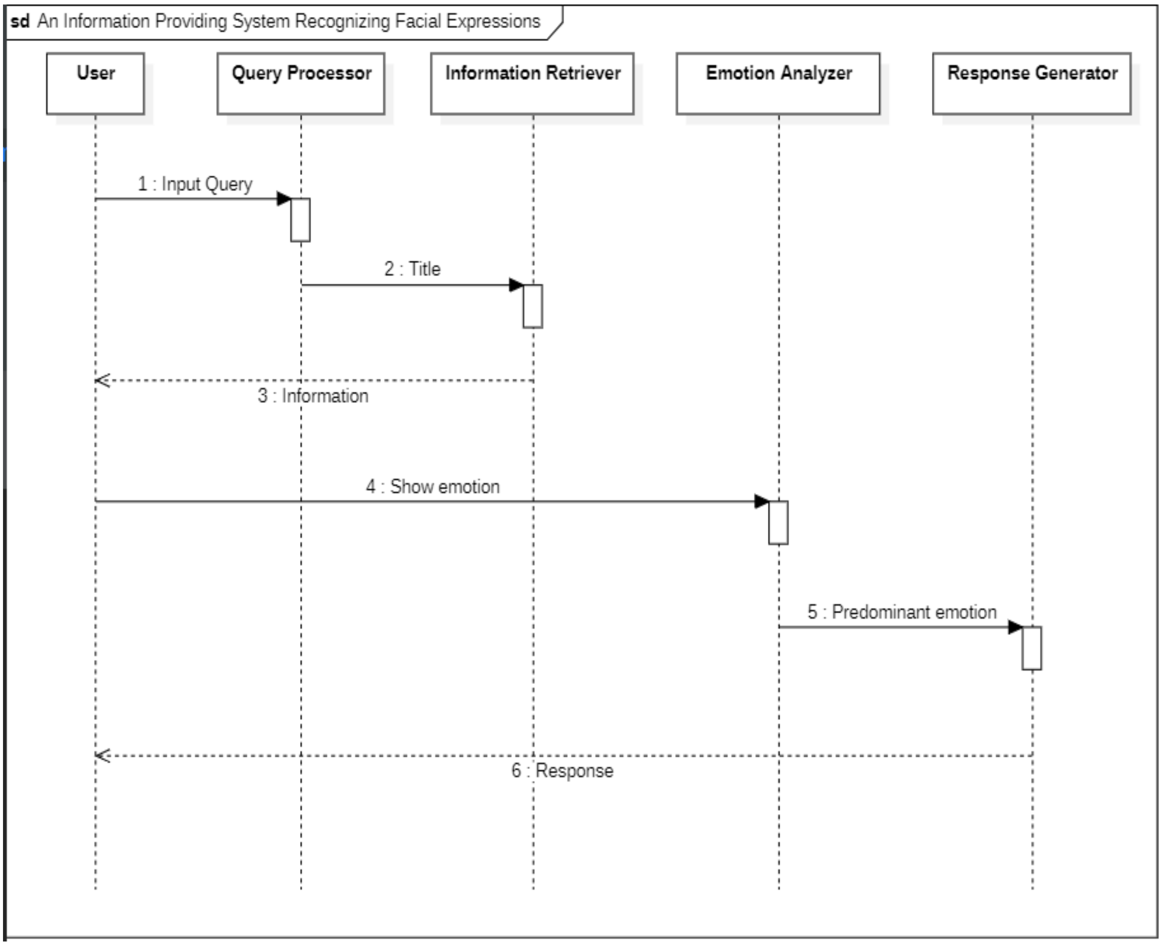

Fig. 3.6 SEQUENCE DIAGRAM

The figure 3.6 denotes the sequence diagram showing the sequence of operations that are performed in the system. Firstly, the user query is inputted which is sent to the query processor for processing the query to remove the special characters and convert it to a list of words. Then, the matching of the query with each title is performed so as to compute the score of similarity to identify the title achieves the maximum score. Then, the information is retrieved and presented to the user. While presenting, the emotion analyzer starts capturing the frames to identify the predominant expression based on cumulative addition of weights assigned to each prediction. Based on the identified predominant expression, a suitable response is provided for the user.

3.1.7. COMPONENT DIAGRAM

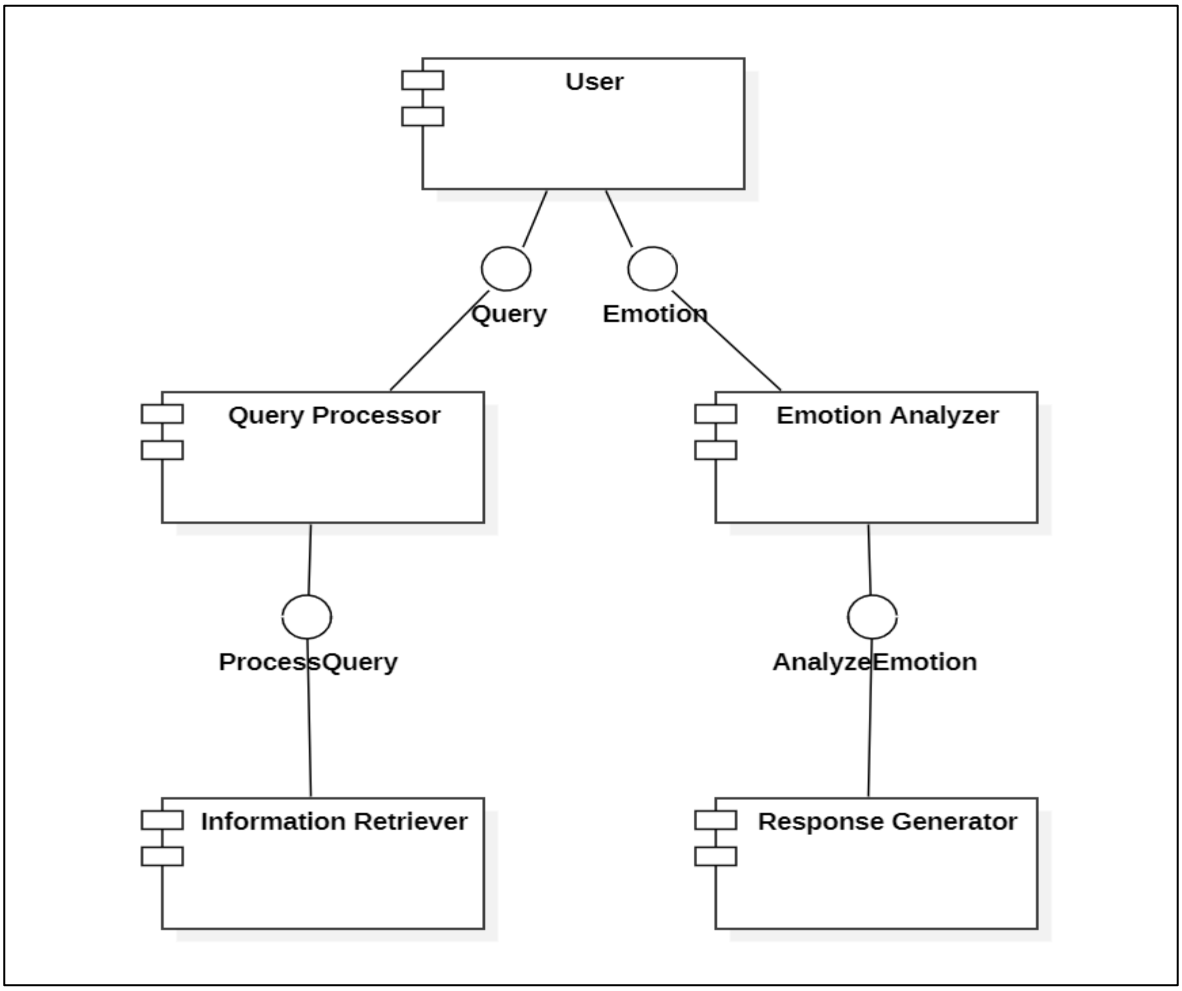

Fig. 3.7 COMPONENT DIAGRAM

The figure 3.7 denotes the component diagram that denotes the various components present in the system. The user component interacts with the query processor component by providing the query. The query processor component gets the query from the user component, processes the query, and sends the title that has the maximum similarity with the query to the information retriever component. The emotion analyzer component gets the emotion from the user component and sends the analyzed predominant expression to the response generator component. The response generator component provides a suitable response based on the predominant expression that is identified.

3.1.8. COLLOBORATION DIAGRAM

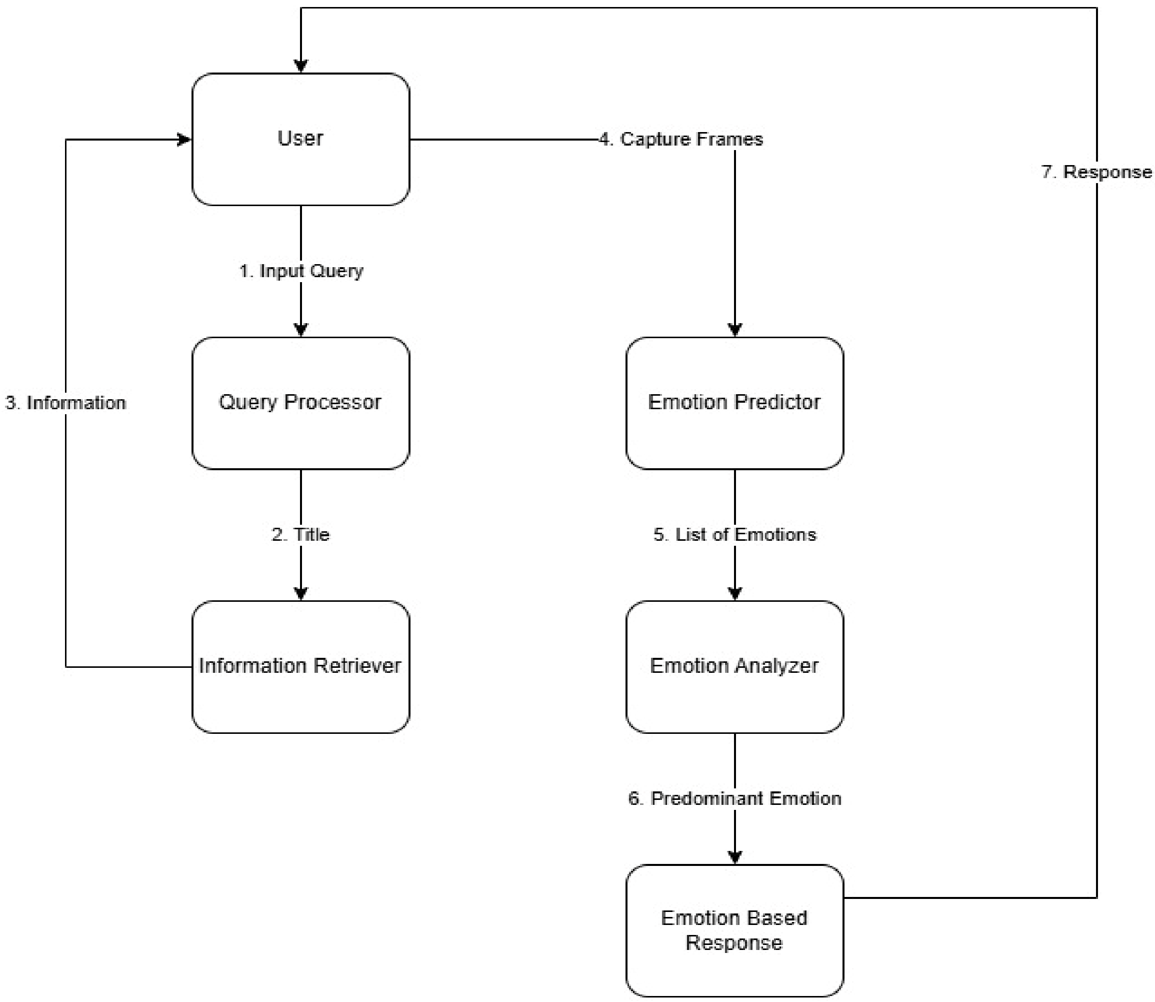

Fig. 3.8 COLLOBORATION DIAGRAM

The figure 3.8 denotes the collaboration diagram which describes on how the components collaborate and communicate with each other. The user gives a query which is then processed to remove special characters and convert it to a list of words. The matching algorithm computes a score between the query and each title to identify the title that the user is looking for. The information is retrieved using the title that is matched and is provided for the user. Then, the frames are captured and the trained deep learning model is used to predict the facial expressions. Finally, the predominant expression is identified to provide an emotion-based response for the user.

3.2 DEVELOPMENT ENVIRONMENT

3.2.1 HARDWARE REQUIREMENTS

Table 3.1 displays the recommended hardware requirements needed in order to run the project in a seamless manner.

PROCESSOR	Intel CORE i7
RAM	16 GB
HARD DISK	512 GB SSD
GRAPHICS PROCESSING UNIT	GPU P100
CAMERA	1280 x 720 (720p)

Table 3.1 Hardware Requirements

3.2.2 SOFTWARE REQUIREMENTS

Table 3.2 displays the software that is needed to develop the project and execute it in a seamless manner.

OPERATING SYSTEM	Windows 11
FRONT END	HTML, CSS, JavaScript, Bootstrap
BACK END	Python 3.10.8
SERVER	Xampp
DATABASE	MySQL

Table 3.2 Software Requirements

CHAPTER 4
MODULE DESCRIPTION

4.1 METHODOLOGY

4.1.1. INFORMATION RETRIEVAL MODULE

This module is a part of the entire system that handles the part of inputting a query from the user in the textual format till providing the requested information. The query that is inputted is processed to remove all the special characters and leave it only with numbers, alphabets, and spaces. The acquired string is then split into individual words using space as the delimiter. The list of words in the query is compared with each title of Wikipedia which is also segmented into individual words using underscore as the delimiter as that is the format in which the list of titles is released by Wikipedia. The fraction of words in the query which are present in the title and the fraction of words in the title which are present in the query is identified. The mean value of both the fraction values is calculated which is considered to be the score of similarity between the query and a title. In an effort to reduce the latency in reaching the suitable title, the query is only checked against those titles that have their starting letters as the initial letters of any of the words in the query. The title that achieves the maximum score of similarity with the query id identified and that is passed to the Wikipedia API (Application Programming Interface) as an argument to retrieve the information that is present in the page with this title. The complete information on the requested page is retrieved in the form of a string. The introduction part of the complete information is only extracted and displayed for the user. The information is displayed letter by letter by introducing latency between printing each letter so as to provide a sensation of reading for the user and it might be more suitable to capture frames and predict the facial expressions of the user throughout this process where the user seems to read the displayed content. This module performs the role of analysing the user query to retrieve the right information and display it for the user in a short period of time.

4.1.2. FACIAL EXPRESSION RECOGNITION MODULE

A deep learning model was trained on a convolutional neural network with residual blocks to predict the facial expressions of the user in real-time when the retrieved information is being displayed. The FER-2013 dataset which consisted of around 28000 images had a great class imbalance as there were only 400 images for the 'disgust' class against 7200 images for the 'happy' class. All the classes did not have equal number of images and there was also a great difference between the class that had the highest number of images and the class that had the lowest number of images. These differences might not make the model equally learn the features of all the classes and the model might become more biased towards one particular class just because it has seen more images of it during the training phase. To avoid that, data augmentation was performed to increase the number of images in the classes that had lesser images and also include varieties in the images so as to make the model handle a variety of situations in real-time. Three types of methods were adopted to increase the size of the dataset and ensure balance in the number of images of all the classes: background addition, brightness adjustment, and contrast adjustment. All the images of the FER-2013 dataset seem like zoomed images focusing only on the faces of the persons. But the images that are collected in real-time would have background parts, which the model might find it difficult to handle if it had not seen any image with background parts during the training phase. So, the existing images in the dataset were resized to half of its original size and placed onto the center of an image of the original size that had all its pixel values of the same random value that is chosen between 0 and 255. The next method used was brightness adjustment which included selecting a random value between -127 to 127 and adding the random value to all the pixels of an existing image to create a new image which had a different level of brightness. The third method was contrast adjustment which included selecting a value between 0.1 and 2.0 and multiplying all the pixels of an existing image by this value to get an image with a different contrast, which indicates the difference between the dark regions of an image and light regions of an image. After applying these techniques to the existing dataset, the size of the dataset was increased to 50613 images. Then, an architecture for a convolutional neural network that contained pooling layers, convolution layers, batch normalization layers, dropout layers, and residual blocks was designed. A residual block allows an input layer to bypass some layers and add its information to the final layer. This can be used when only the difference between the input and the output needs to be learnt instead of the complete transformation that happens in between. Dropout layers are used to avoid overfitting of the model on the training data by memorizing the features of the training data alone, which makes it difficult for the model to generalize and handle unseen situations. Dropout layers randomly drop some percentage of the neurons in each layer so as to prevent the network from relying on particular neuron that might possibly memorize a specific piece of information. This is followed by the fully connected layers that are dense connections finally leading to the output layer of seven neurons each outputting a probability value between 0 and 1 using the Softmax function. The seven neurons indicate the seven classes: happy, angry, fear, disgust, neutral, surprise, and sad which the model classifies each image frame into. Whichever class gets the highest probability score is the final prediction of the model for an image frame. When the information is being displayed, the images frames are captured by the camera every 2 seconds and the facial expression predictions are done by the trained model to store the predictions in an array. The first half of the predictions are given a weightage of 0.5 and the second half of the predictions are given a weightage of 1.0. A score is computed for each class by performing a cumulative addition of weights assigned to each prediction. Whichever class gets the maximum score is decided to be the predominant expression of the user throughout the reading time. Based on this identified predominant expression, the needs and expectations of the user are understood and suitable responses are provided so as to modify the presented content in order to suit the requirements of the user.

CHAPTER 5

IMPLEMENTATION AND RESULTS

The list of Wikipedia titles contains around 16 million records. Traversing through this huge list is time consuming and the system might not provide the requested information in a reasonable time if appropriate methods are not used. Only traversing through the titles that had their starting letters as the starting letters of any of the words in the query, reduced the time to retrieve information by around 75% compared to traversing through the entire list of 16 million titles. The model trained on the convolutional neural network using residual blocks has achieved an accuracy of 65.11% on the 7178 test images across all the seven classes. Table 5.1 denotes the values obtained for various performance metrics.

Metric	Values
Accuracy	65.11%
Precision	66.47%
Recall	62.61%
F1-score	63.89%

Table 5.1 Performance Metrics

The increased dataset was separated into 80% training data and 20% validation data. The accuracy reached on the validation data was monitored throughout the period of training. If the accuracy on the validation data did not improve for 10 continuous epochs, then the early stopping method was used to stop the training process as it is an indication of when the model starts to memorize the features present in the training data instead of learning the relationships between the input image and the output category. Both the training accuracy and validation accuracy start to increase as the number of epochs starts progressing with the validation accuracy even recording higher values than training accuracy at some stages in the initial phase. Then, as the training progresses, the accuracy on the training data starts to increase than the accuracy on the validation data and the validation accuracy starts to attain a stable state without increasing after 34 epochs. Figure 5.1 shows the trends of the accuracy on the training data and the accuracy on the validation data during the training process of the model.

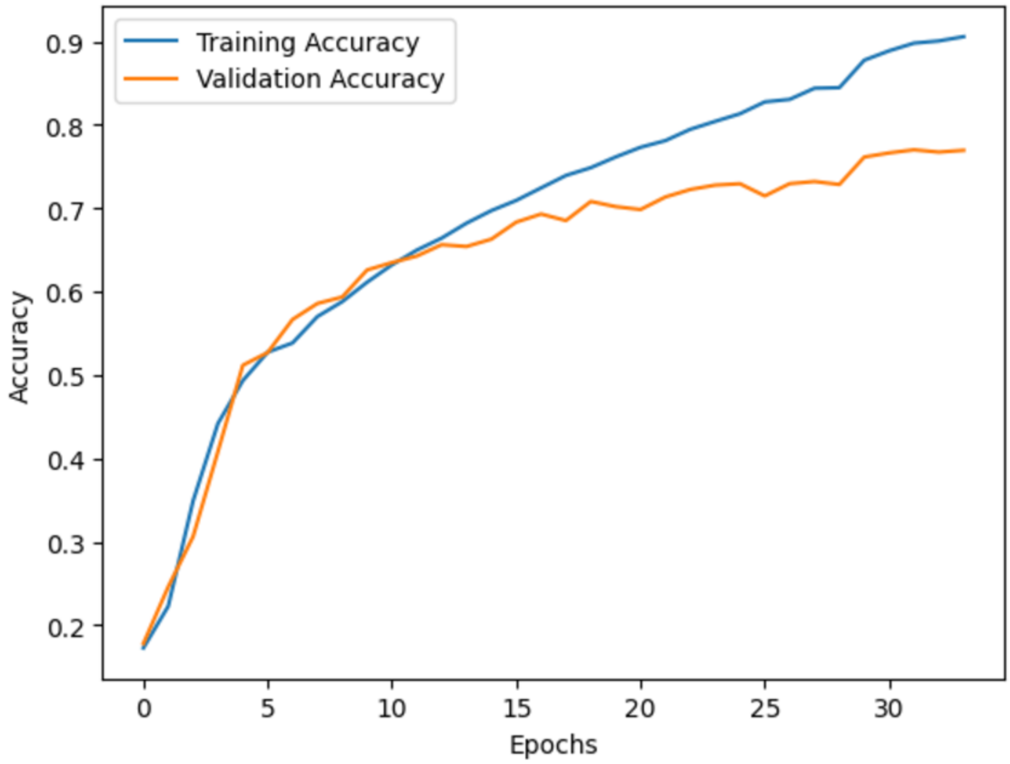

Fig. 5.1. Trends of accuracy

The learning rate was initialized to 0.0005. The loss on the validation data was also monitored to check if it does not improve for continuous 5 epochs. If the loss on the validation data did not decrease for five consecutive epochs, then the learning rate was reduced by a factor of 0.1 so as to prevent erratic updates when the model is reaching a minimum point on the graph of the loss function. The learning rate was reduced by a factor of 0.1 after 27 epochs and after 34 epochs. Figure 5.2 depicts the values of the learning rate that were maintained during the training process of the model on the dataset.

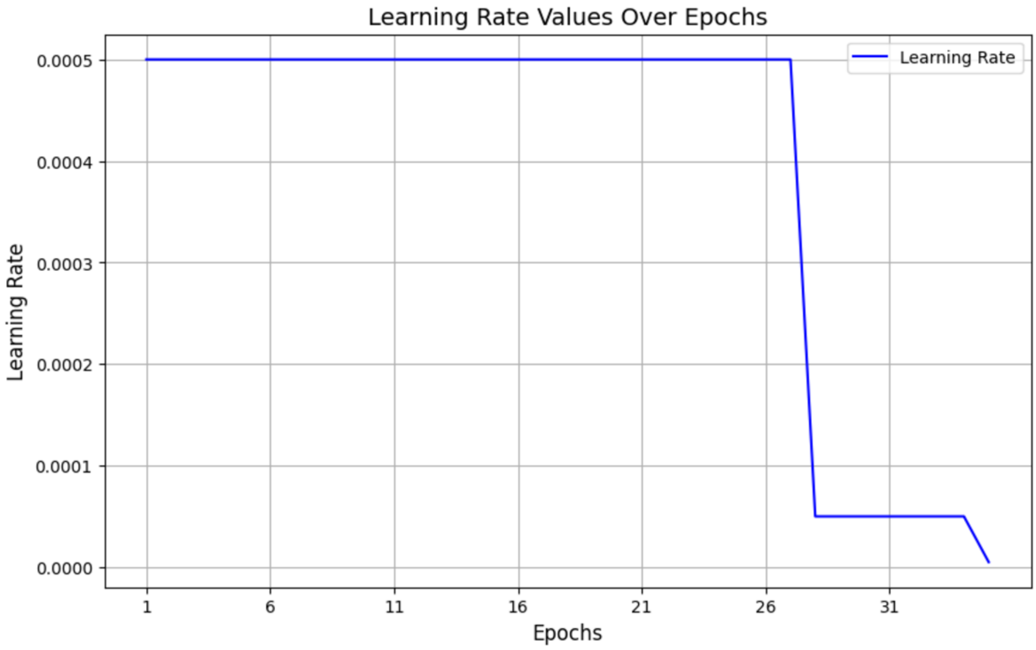

Fig 5.2. Learning rate over epochs

Figure 5.3 shows a page of the developed application where the requested information is retrieved for the user and a response based on the facial expression of the user is also provided.

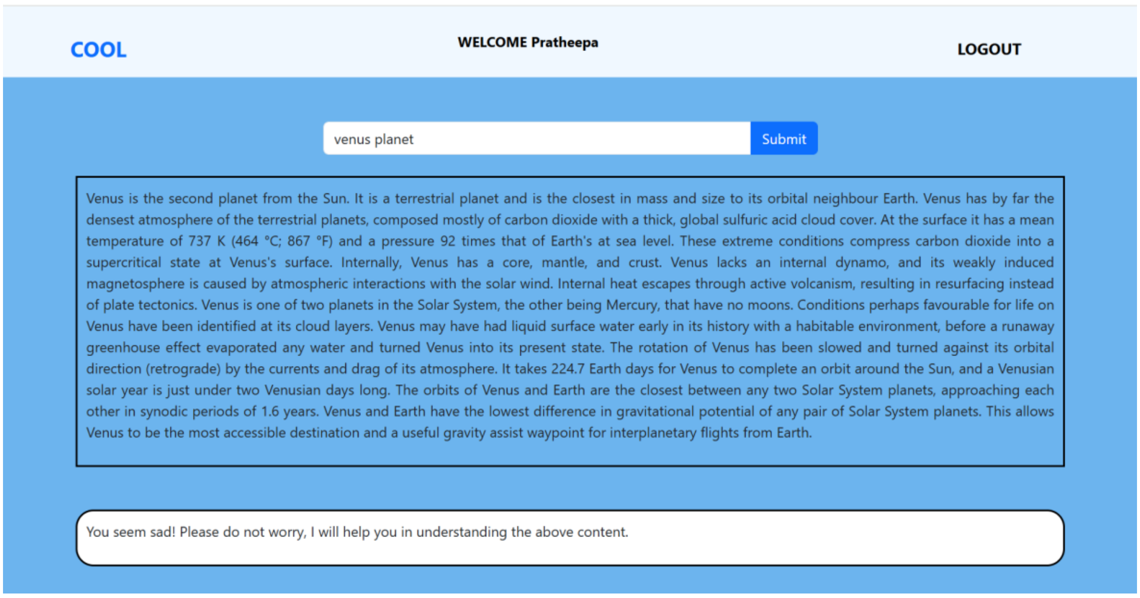

Fig 5.3 Output Screenshot

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1. CONCLUSION

The project aims to not only provide the requested information but also recognize the facial expressions of the user while reading through the provided content. The predominant expression of the user throughout the learning process is used to provide suitable responses and can be used to adapt the content so as to cater to the needs or expectations of the user. This system provides a rich user experience and also aims to provide an effective learning process for the user.

6.2. FUTURE WORK

As part of the work allocated for phase II, we aim to provide the functionality of modifying the provided information based on the predominant expression of the user. Future work will focus on recognizing expressions in occluded images and inclusion of additional data sources and advanced algorithms to retrieve information which would help to provide the user with the desired content and reduce the latency of delivering the information.

APPENDIX – I

LIST OF PUBLICATIONS:

1. **PUBLICATION STATUS:** IN REVIEW PROCESS

   TITLE OF THE PAPER: AN APPROACH TO DEVELOP THE INFORMATION PROVIDING SYSTEM BY RECOGNIZING FACIAL EXPRESSIONS

   AUTHORS: DR. P. KUMAR, DR. SENTHIL PANDI S, PRATHEEPA R, MANNURU SHREEYA

   NAME OF THE CONFERENCE: INTERNATIONAL CONFERENCE ON ADVANCES IN COMPUTER SCIENCE, ELECTRICAL, ELECTRONICS, AND COMMUNICATION TECHNOLOGIES

APPENDIX – II

```
from tensorflow.keras.models import load_model
import numpy as np
import wikipedia
import time
import threading
import cv2
file=open("wikipedia_titles","r",encoding="utf-8")
model=load_model("finalmodel.h5",compile=False)
classes=["angry","disgust","fear","happy","sad","neutral","surprise"]
def preprocessing(s):
    newstring=""
    for i in s:
        if i.isalnum() or i==" " or i=="_":
            newstring+=i
    return newstring
def printing(result):
    for i in result:
        print(i,end="")
        time.sleep(0.01)
```

```
predictedlabels=[]
def captureframes():
    if camera.isOpened():
        captured, frame=camera.read()
        if captured:
            frame=frame[120:360,160:480]
            step1=cv2.resize(frame,(160,160))
            step2=cv2.resize(step1,(96,96))
            frame=cv2.resize(step2,(48,48))
            temp=[]
            for i in frame:
                l=[]
                for j in i:
                    l.append(int(0.114*j[0]+0.587*j[1]+0.299*j[2]))
                temp.append(l)
            frame=np.array(temp)
            temp=[]
            for i in frame:
                l=[]
                for j in i:
                    x=[j,j,j]
```

```
            l.append(x)
          temp.append(l)
        frame=np.array(temp)
        x=[]
        x.append(frame)
        x=np.array(x)
        x=x/255.0
        predictions=model.predict(x,verbose=0)
        predictedlabels.append(np.argmax(predictions[0]))
def getlimit(s):
  c=0
  for i in range(len(s)):
    if s[i]=="\n":
      c+=1
    if c==3:
      index1=i
      break
  index2=s.index("==")
  return min(index1,index2)
query=input("What do you like to know about?\n")
query=preprocessing(query.lower()).split()
```

```
maximum_score=0
print("Loading....")
list_of_titles=[title for title in file]
title_firstletters=[title[0] for title in list_of_titles]
titles=[]
for i in query:
    if i[0].isalpha():
        lastindex=len(title_firstletters)-1-(title_firstletters[::-
1].index(i[0].upper()))
        titles+=list_of_titles[title_firstletters.index(i[0].upper()):lastindex+1]
for title in titles:
    processed_title=preprocessing(title.lower()).strip().split("_")
    a=0
    for i in processed_title:
        if i in query:
            a+=1
    b=0
    for i in query:
        if i in processed_title:
            b+=1
    score=((a/len(processed_title))+(b/len(query)))/2
    if score>=maximum_score:
```

```
        maximum_score=score
        answer_query=title
if maximum_score==0:
    print("Sorry! I don't have that information.")
else:
    try:
        page=wikipedia.page(answer_query)
        result=page.content
        endindex=getlimit(result)
        result=result[:endindex]
        camera=cv2.VideoCapture(0)
        printingthread=threading.Thread(target=printing,args=(result,))
        printingthread.start()
        while printingthread.is_alive():
            captureframes()
            time.sleep(2)
        c=0
        while c<=3:
            captureframes()
            time.sleep(2)
            c+=1
```

```
camera.release()
score_eachlabel=[0]*7
if len(predictedlabels)>0:
    midvalue=int(len(predictedlabels)/2)
    for i in predictedlabels[:midvalue]:
        score_eachlabel[i]+=0.5
    for i in predictedlabels[midvalue:]:
        score_eachlabel[i]+=1
    finalprediction=classes[np.argmax(score_eachlabel)]
else:
    finalprediction="neutral"
print("\n\n")
if finalprediction=="neutral":
    print("If you need any information about anything else, please let me know!")
elif finalprediction=="happy":
    print("You seem so interested! I will provide you with some more information.")
elif finalprediction=="sad":
    print("You seem sad! Please do not worry, I will help you in understanding the above content.")
elif finalprediction=="surprise":
```

```
            print("You seem so enthusiastic! I will provide you with some
interesting facts about the topic.")
        elif finalprediction=="angry":
            print("You don't seem happy! I will provide the information in a
manner that would help you in understanding.")
        elif finalprediction=="fear":
            print("Please don't be worried! I will let you know of some positive
aspects about the topic.")
        elif finalprediction=="disgust":
            print("I see that you are not satisfied with the content! Please let me
know on what specific information I can provide.")
    except wikipedia.DisambiguationError:
        print("Please be more specific on the information that you need")
    except wikipedia.PageError:
        print("Sorry! I don't have that information.")
```

REFERENCES

[1] M. Aly, A. Ghallab and I. S. Fathi, "Enhancing Facial Expression Recognition System in Online Learning Context Using Efficient Deep Learning Model," in IEEE Access, vol. 11, pp. 121419-121433, 2023, doi: 10.1109/ACCESS.2023.3325407.

[2] W. E. Villegas-Ch, J. García-Ortiz and S. Sánchez-Viteri, "Identification of Emotions From Facial Gestures in a Teaching Environment With the Use of Machine Learning Techniques," in IEEE Access, vol. 11, pp. 38010-38022, 2023, doi: 10.1109/ACCESS.2023.3267007.

[3] G. Zhao, H. Yang and M. Yu, "Expression Recognition Method Based on a Lightweight Convolutional Neural Network," in IEEE Access, vol. 8, pp. 38528-38537, 2020, doi: 10.1109/ACCESS.2020.2964752.

[4] J. Liu, H. Wang and Y. Feng, "An End-to-End Deep Model With Discriminative Facial Features for Facial Expression Recognition," in IEEE Access, vol. 9, pp. 12158-12166, 2021, doi: 10.1109/ACCESS.2021.3051403.

[5] N. Zhou, R. Liang and W. Shi, "A Lightweight Convolutional Neural Network for Real-Time Facial Expression Detection," in IEEE Access, vol. 9, pp. 5573-5584, 2021, doi: 10.1109/ACCESS.2020.3046715.

[6] J. Pu and X. Nie, "Convolutional Channel Attentional Facial Expression Recognition Network and Its Application in Human–Computer Interaction," in IEEE Access, vol. 11, pp. 129412-129424, 2023, doi: 10.1109/ACCESS.2023.3333381.

[7] P. Ganesan, S. Kumar Jagatheesaperumal, I. Gobhinath, V. Venkatraman, S. N. Gaftandzhieva and R. Z. Doneva, "Deep Learning-Based Interactive Dashboard for Enhancing Online Classroom Experience Through Student Emotion Analysis," in IEEE Access, vol. 12, pp. 91140-91153, 2024, doi: 10.1109/ACCESS.2024.3421282.

[8] M. D’incà, C. Beyan, R. Niewiadomski, S. Barattin and N. Sebe, "Unleashing the Transferability Power of Unsupervised Pre-Training for Emotion Recognition in Masked and Unmasked Facial Images," in IEEE Access, vol. 11, pp. 90876-90890, 2023, doi: 10.1109/ACCESS.2023.3308047.

[9] M. Shi, L. Xu and X. Chen, "A Novel Facial Expression Intelligent Recognition Method Using Improved Convolutional Neural Network," in IEEE Access, vol. 8, pp. 57606-57614, 2020, doi: 10.1109/ACCESS.2020.2982286.

[10] E. Uzun, "A Novel Web Scraping Approach Using the Additional Information Obtained From Web Pages," in IEEE Access, vol. 8, pp. 61726-61740, 2020, doi: 10.1109/ACCESS.2020.2984503.

[11] G. Suddul, C. Lillmond and S. Armoogum, "A Smart Virtual Tutor with Facial Emotion Recognition for Online Learning," 2022 IEEE Zooming Innovation in Consumer Technologies Conference (ZINC), Novi Sad, Serbia, 2022, pp. 67-72, doi: 10.1109/ZINC55034.2022.9840683.

[12] B. Fang, X. Li, G. Han and J. He, "Facial Expression Recognition in Educational Research From the Perspective of Machine Learning: A Systematic Review," in IEEE Access, vol. 11, pp. 112060-112074, 2023, doi: 10.1109/ACCESS.2023.3322454.

[13] A. V. Savchenko, L. V. Savchenko and I. Makarov, "Classifying Emotions and Engagement in Online Learning Based on a Single Facial Expression Recognition Neural Network," in IEEE Transactions on Affective Computing, vol. 13, no. 4, pp. 2132-2143, 1 Oct.-Dec. 2022, doi: 10.1109/TAFFC.2022.3188390.

[14] R. Miao, H. Kato, Y. Hatori, Y. Sato and S. Shioiri, "Analysis of Facial Expressions to Estimate the Level of Engagement in Online Lectures," in IEEE Access, vol. 11, pp. 76551-76562, 2023, doi: 10.1109/ACCESS.2023.3297651.

[15] R. Bhargava, R. Lobo, R. Shah, N. Shah and S. Nair, "Easier Web Navigation Using Intent Classification, Web Scraping and NLP Approaches," 2022 5th International Conference on Advances in Science and Technology (ICAST), Mumbai, India, 2022, pp. 286-290, doi: 10.1109/ICAST55766.2022.10039559.

[16] Y. He, Y. Zhang, S. Chen and Y. Hu, "Facial Expression Recognition Using Hierarchical Features With Three-Channel Convolutional Neural Network," in IEEE Access, vol. 11, pp. 84785-84794, 2023, doi: 10.1109/ACCESS.2023.3303402.

[17] D. -H. Lee and J. -H. Yoo, "CNN Learning Strategy for Recognizing Facial Expressions," in IEEE Access, vol. 11, pp. 70865-70872, 2023, doi: 10.1109/ACCESS.2023.3294099.

[18] Y. Tang, X. Zhang, X. Hu, S. Wang and H. Wang, "Facial Expression Recognition Using Frequency Neural Network," in IEEE Transactions on Image Processing, vol. 30, pp. 444-457, 2021, doi: 10.1109/TIP.2020.3037467.

[19] S. Li, X. Li, L. Shang, X. Jiang, Q. Liu, C. Sun, Z. Ji and B. Liu, “HopRetriever: Retrieve Hops over Wikipedia to Answer Complex Questions”, AAAI, vol. 35, no. 15, pp. 13279-13287, May 2021.

[20] X. Yang, Y. Fang and C. Raga Rodolfo, "Graph Convolutional Neural Networks for Micro-Expression Recognition—Fusion of Facial Action Units for Optical Flow Extraction," in IEEE Access, vol. 12, pp. 76319-76328, 2024, doi: 10.1109/ACCESS.2024.3406037.

AN INFORMATION PROVIDING SYSTEM RECOGNIZING FACIAL EXPRESSIONS_copy.docx

ORIGINALITY REPORT

12%	6%	9%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Bei Fang, Xian Li, Guangxin Han, Juhou He. "Facial Expression Recognition in Educational Research From the Perspective of Machine Learning: A Systematic Review", IEEE Access, 2023 Publication	1%
2	www.researchgate.net Internet Source	1%
3	Jing Pu, Xinxin Nie. "Convolutional Channel Attentional Facial Expression Recognition Network and Its Application in Human–Computer Interaction", IEEE Access, 2023 Publication	1%
4	William Eduardo Villegas-Ch, Joselin García-Ortiz, Santiago Sánchez-Viteri. "Identification of Emotions From Facial Gestures in a Teaching Environment With the Use of Machine Learning Techniques", IEEE Access, 2023 Publication	1%

An Approach to Develop the Information Providing System by Recognizing Facial Expressions

Kumar P
Department of CSE
Rajalakshmi Engineering College
Chennai, India
kumar@rajalakshmi.edu.in

Senthil Pandi S
Department of CSE
Rajalakshmi Engineering College
Chennai, India
mailtosenthil.ks@gmail.com

Pratheepa R
Department of CSE
REC, Chennai, India
210701192@rajalakshmi.edu.in

Mannuru Shreeya
Department of CSE
REC, Chennai, India
210701148@rajalakshmi.edu.in

***Abstract*— Facial expressions are very important in understanding one's state of mind. We aim to incorporate this notion in our learning assistant application in order to provide a rich user experience. The facial expression of the user is recognized with the help of a model trained on the FER-2013 dataset using a convolutional neural network. A query from the user is matched with an appropriate title from the list of titles released by Wikipedia using a matching algorithm based on the percentage of words that match in both the query and a title. The information under the title that achieves the maximum score is retrieved and provided for the user. Simultaneously, the facial expressions of the user are captured so as to analyze the level of understanding of the content for the user and display suitable responses. The percentage of correct predictions given by the proposed model on 7178 unseen images is 65.11%.**

Keywords— Face Expression, Expression Recognition, Information System, Attention Mechanism.

I. INTRODUCTION

In today's world and era of growing technology, self-learning in the online mode is increasing every day. Various kinds of resources are made accessible to people all across the world. Wikipedia serves as a huge resource for information about a lot of things and it releases a copy of all the titles and the available content in the form of dumps regularly. This project uses the list of titles to match the query of the user with the appropriate title and display the needed information. Facial expressions are a very important index in understanding the mindset of a person. A human teacher takes all efforts to also look at the facial expressions of the students apart from delivering content in order to modify the lecture and make the students understand better. Likewise, recognizing facial expressions while delivering information can prove to be a very important tool in improving the learning process for the user.

The image is received through a webcam that turns on while the information is being displayed. The quantity of data used to train the expression recognizing model is increased by introducing variations in the existing data in order to avoid class imbalance and also make the model generalize well and perform better in unseen situations. The model is trained on the increased data of around 50000 images using a convolutional neural network along with methods deployed to prevent the model from learning to memorize the properties of the data that it is being trained on. The model classifies any image into one of the seven categories: angry, happy, disgust, sad, surprise, fear, and neutral. The system gives less weightage to the predictions made in the first half of the time period spent in capturing the frames from the webcam, and gives more weightage to the predictions made in the second half. The final prediction is decided by looking at which category has accumulated the maximum number of points that is calculated by summing the weights for each prediction. Suitable responses are displayed for the user based on the final prediction.

II. LITERATURE SURVEY

In the paper [1], the enhanced version of the ResNet-50 model incorporates a Convolutional Block Attention Mechanism (CBAM) to improve expression recognition accuracy. The algorithm in uses a pre-trained machine learning model to recognize six different emotions. A laboratory was set up to capture facial expressions during practical sessions, with images captured in batches of 350 per student. 7000 images were collected, corresponding to six categories: commitment, boring, frustration, focused, interested, and neutral. The work in proposes a lightweight emotion recognition model based on DenseNet architecture, which uses highly linked convolution layers and model compression methods. The proposed model in contains three stages: pre-processing of data, extraction of features, and classification. The ResNet network is replaced in the first half of the VGG network, and the classification layer uses the Softmax function to give the prediction. The research in [2] presents a model for recognizing emotions using Multi-task cascaded convolutional networks (MTCNN) detection, trained using the ADAM optimizer. A FER algorithm for human-computer interaction systems is designed in to enable robots understand facial expressions like humans. The pruning strategy that is proposed has good model acceleration, reducing memory usage and improving classification accuracy. The regenerative Generative Adversarial Network (GAN) is used in [3] to recreate obscured areas of students' faces, essential for precise emotional interpretation. The dashboard offers a user-friendly interface for educators, with visualizations associated with different emotions. Teachers can access real-time feedback and historical data monitoring to adjust their teaching methods. The Convolutional Autoencoder (AE) is the proposed method in [4] for recognizing partial occlusions in images caused by objects like sunglasses and hands. The encoder's transpose version is the decoder, which reconstructs the original image using the latent space. The AE is used to extract features and a Multilayer Perceptron (MLP) that is a linear classifier is trained, which classifies positive and negative emotions. The Fuzzy C-Means clustering technique is used in for facial detection using the Haar-Like algorithm. Feature extraction is performed using a trained CNN model, and the retrieved characteristics are used for emotion classification using a Support Vector Machine (SVM). The paper [5] introduces a web scraping approach called "UzunExt" that improves the efficiency of extracting content from web pages. UzunExt uses string searching techniques to locate the start and end tags of the desired content, achieving faster content extraction than traditional methods. The paper [6] presents a recurrent neural network using long short term memory algorithm to predict the most plausible outcome for a query. The query is loaded into a JSON document, tokenized, and converted into numeric configuration. Keras is used for creating a sequential model for a deep convolutional neural network trained on the FER 2013 dataset. The paper [7] discusses the various facial expression identification methods used in educational research. A lightweight CNN model for face recognition is trained in using video stills of faces. The model's training process includes a separate process for face recognition and fine-tuning for emotion categorization. A study in [8] involving nine video lectures revealed that students' engagement levels are crucial for improving learning outcomes. Participants were instructed to press a key when they noticed an auditory target, with an average loudness of 70 db and white noise of 0.66 db. Facial features were recorded during the experiment, and it was found that deviation from the lecture can be understood by increased nose wrinkling. Facial expressions can predict learners' attention levels, with video camera-captured facial features predicting reaction times (RTs), which are representative of attentional states. The Naive Bayes classification model is utilized in due to its effectiveness. It uses segment embedding and token embedding to process questions, while BERT, a pre-trained model, responds to user requests using incoming information. The scraping process uses BeautifulSoup (BS4) and Selenium Web Driver. The paper presents a three-channel convolutional neural network to increase the accuracy of facial expression identification. Augmentation techniques like random horizontal flipping and brightness adjustment were applied to increase the volume of data. The study in [9] introduces a divide and conquer convolutional neural network (CNN) learning approach to improve facial expression recognition accuracy. The approach involves identifying, focusing, and normalizing face regions, and adjusting the ResNet-18 model to accommodate modifications like lowering filter sizes and extracting more relevant information from face photos. In the paper, a FreNet model is a lightweight deep learning model that uses the Discrete Cosine Transform (DCT) to convert images into the frequency domain. This process focuses on capturing low-frequency components, which contain critical information about the image. The methodology outlined in [10] focuses on multi-hop open-domain question answering (QA) by using both structured and unstructured information from Wikipedia. The study presents a new retrieval technique named HopRetriever, intended to collect scattered reasoning evidence from several sources. It involves defining a "hop", which is the combination of a hyperlink and its corresponding outbound content. The system systematically acquires additional documents by traversing a sequence of "hops" between them, enhancing the evidence compilation at each stage. This enables HopRetriever to identify relevant documents that aid in addressing intricate inquiries. The paper presents a graph convolutional neural network designed for recognizing micro-expressions. It uses Facial Action Units (FAUs) to analyze detailed changes in facial expressions, and the Optical Flow Method (OFM) to estimate pixel motion in facial images.

III. PROPOSED MODEL

The proposed system mainly contains two components: (A) Information Retrieval and (B) Recognizing Facial Expressions. We will discuss these two components in detail elucidating on the steps adopted to build them.

Information Retrieval: Our work aims to handle queries related to providing information about any given topic. Wikipedia acts as a huge resource of information for a wide variety of topics and it also releases the list of titles and all the available content in the form of dumps in a regular manner. The list of titles released can be used to perform a matching algorithm between the user query and each title so as to find what information the user is seeking. The user query might be in natural language indicating the needs and might not always contain the exact order of words as present in the actual title. So, performing an algorithm to match the query with the actual title as present in the information resource is necessary to retrieve the content that the user is asking for. The algorithm traverses through the titles to compute a score for each of them by comparing the user query and each title. The title that attains the highest score is chosen indicating that it matches the user query to the maximum extent and conveying that the user has asked for this specific information. The score is calculated by computing the average of the fraction of words of the title that are present in the query and the fraction of words of the query that are present in the title. The formula is given by:

$$score = (a/b + c/d) / 2$$

where 'score' indicates the value of similarity between the user query and a title, 'a' indicates the count of words in the title that are present in the query, 'b' indicates the count of words in the title, 'c' indicates the count of words in the query that are present in the title, and 'd' indicates the count of words in the query.

The algorithm only traverses through the subset of the list of titles that begin with the starting alphabets of all the words in the user query to reduce the time taken to retrieve the content and improve the efficiency. The content is displayed letter by letter with a time interval so as to create a situation of capturing the facial expressions of the user while reading through the provided information. Displaying the information and recording the facial expressions are implemented as two different threads that execute simultaneously. While the content is getting printed, the frames are captured from the webcam at a regular interval of 2 seconds and the expression predictions for each frame are stored.

Recognizing Facial Expressions: A deep neural network model was trained on the FER-2013 dataset using a convolutional neural network to identify facial expressions. The FER dataset has a lot of positive aspects on having images that cover facial expressions of people from different angles and containing images for seven different classes: angry, happy, disgust, sad, surprise, fear, and neutral. The images in the dataset are of shape (48,48,3) with height of 48 pixels, width of 48 pixels, and three channels defining the color of a pixel using the values of red, green, and blue. All the three values of a pixel are identical leading to every image in the dataset appearing with only shades of gray and not containing any other colors. One major issue that needs to be addressed before training any classification model is ensuring class balance. The 'happy' class contains around 7200 images whereas the 'disgust' class contains only around 400 images. This huge difference might make the model not completely capture or learn the features of the 'disgust' class and also might make the predictions to be more biased towards the 'happy' class. Hence, some methods are used to increase the amount of data in the unbalanced classes to ensure that the model learns the features of every class by giving equal importance to all the categories of expressions. The methods used to increase data in the unbalanced classes that have less images are:

Zoom Out: All the images in the dataset contain only the faces of humans and do not have any background parts. But the images in real-time would contain background parts which might make the model not to be effectively trained against those situations. Therefore, introducing images with some background parts in the training phase is necessary to train the model well to adapt to various situations. An image is resized to half its size and a random value is chosen in the range of [0, 255]. The random number is decided as the value of all the three channels that form a pixel and a new image is created of the original size of an image in the dataset with all the pixels of the same values. Then, the resized image is placed onto the center of the new image thus forming an image that seems to contain a background apart from the face.

Fig.1 Model Data Augumentation

(ii) Adjusting Brightness: The images that would be delivered for prediction in real-time would be having different brightness levels. Therefore, training images with varying brightness levels is necessary to accommodate all the situations. A random value is chosen in the range of [-127, 127]. This value is added to all the pixels in the image to produce a modified image of a different brightness level. If the values of the pixel go out of the range of [0, 255], they are clipped to the nearest extreme value.

(iii) Adjusting Contrast: Contrast of an image is the difference between the dark regions and the light regions. If the difference is more, the image is said to have a high contrast and the objects would be visible well against the background. If the difference is less, the image is said to have a low contrast. The images given in real-time can have varying contrast. Therefore, including images of varying contrast in the training phase would help the model to handle different situations. A random value is chosen in the range of [0.1, 2.0]. All the pixel values in an image are multiplied by this random value to get different pixel values that may have higher contrast or lower contrast based on the random value chosen. A value greater than 1.0 would increase the contrast as it would widen the difference between pixel values. A value lesser than 1.0 would lead to an image with a low contrast as it would narrow the difference between pixel values. If any value goes out of the range of [0, 255], then it is clipped back to the nearest extreme value. These three techniques not only increase the amount of data to attain class balance, but also introduces variations in the training data that makes the model handle varied situations. After application of these techniques to the images present in the unbalanced classes, each class contains around 7200 images with the total images in the dataset to be 50613.

Fig.2 Model Contrast Adjustment

A model is trained on this entire dataset using a convolutional neural network that would learn the biases, the weights, and the filter values of the network in order to provide a prediction for any new image. The architecture of the neural network contains convolution layers that uses filters to span through the image with a defined kernel size, compute the dot product of the filter values and the pixel values, and use the ReLU activation function to help the model learn the relationships between the input and the output that are not linear, and produce feature maps of multiple channels. The count of channels in a layer are defined by the count of filters used in the previous layer. The training is done in batches in a single epoch. A batch size of 32 is used during the training process. Batch normalization layers are used to make the data in a particular channel across all the images of a batch to have a variance of 1 and a mean of 0. Max-pooling layers are used to capture the most important features in a generated feature map and also reduce the number of features so as to prevent the model from just memorizing the data on which it was trained. If the model learns a large number of features, it will likely tend to learn only the features in the provided data on which it is trained and might not be able to generalize well for unseen situations. Dropout layers are used to drop some percentage of the neurons in some of the layers to prevent the model from relying only on some specific features to decide the prediction. Residual blocks are used to perform a transformation of a layer using convolution and batch normalization layers and add the initial layer to the transformed layer to provide the final layer. This helps in creating a shortcut and passing the information from the earlier layers to bypass one or more layers and feed the information into later layers. The layer for convolution operation, the layer for normalizing the extracted feature values, residual block layer, the layer to capture the important features, and a dropout layer is considered to be a block and four such blocks are used in succession. It is followed by a global average pooling layer that computes the mean value of all the values in each feature map and generates a one-dimensional vector that then proceeds onto the dense layers. The dense layers also have batch normalization layers and dropout layers used along with them. The final layer is the layer of seven neurons where each neuron predicts the probability of each possible class using the softmax function. The final output is returned as a vector of seven values with each value denoting the probability of a class. The class that has the maximum probability is considered to be the resulting prediction.

The maximum number of epochs is decided to be 60 and an early stopping method is used to stop the process of training when the model starts to overfit the data on which it is being trained. The data is separated into 80% training data and 20% validation data by training on the training data and monitoring the loss values and the accuracy values of the validation data. If the loss on the data used for validation does not decrease after consecutive 10 epochs, the process of training is halted as it means that the model has started to memorize or overfit the data on which it is getting trained and might not be able to give accurate predictions on unseen data. If the loss on the data that is used for validation does not decrease for consecutive 5 epochs, it is understood that the model is approaching a minimum point, and therefore, the learning rate is reduced by a factor of 0.1 so that the further weight updates would be minimal and would not lead to any erratic updates, which would make the model converge at the minimum point. A combination of early stopping and adaptive learning rate methods is used to stop the model from overfitting and also help the model converge at the minimum point. Adam optimizer is used to update the biases, the weights, and the filter values of the network.

This trained model is used to predict the facial expressions of the image captured from the webcam every 2 seconds while displaying the retrieved information. The predictions are stored in the form of an array. The first half of the predictions are given a weightage of 0.5 and the second half of the predictions are given a weightage of 1.0.

The total weightage value for each class is computed so as to find the final prediction. The class that gets the maximum weightage denotes the predominant expression of the user and is considered to be the final prediction of the expression of the user while reading through the content.

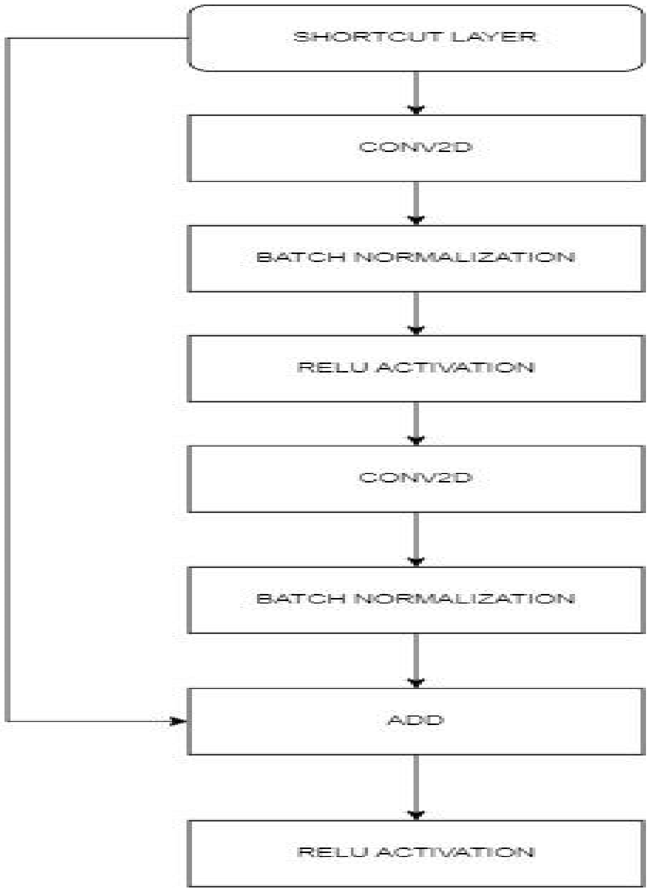

Fig.3 Proposed Model Work Flow

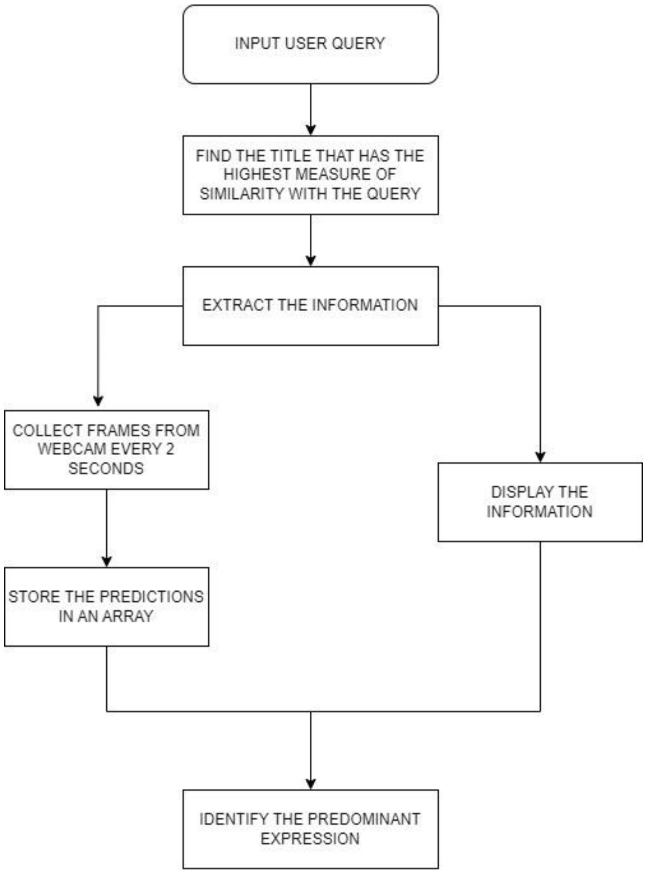

Fig.4. Model Architecture

IV. RESULT

A model was trained using a convolutional neural network before performing data augmentation on the initial 28709 training images from the FER-2013 dataset. The model recorded an accuracy of 44.10% with literally no predictions for the 'disgust' class due to the low count of images present in that category. There were only around 400 images present for the 'disgust' class against 7200 images for the 'happy' class, leading to a great imbalance. Figure 5 represents the confusion matrix of the above model.

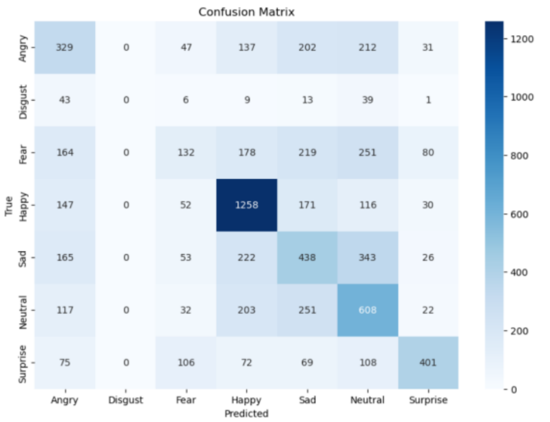

Figure 5. Confusion matrix of the model trained prior to data augmentation

The amount of data was increased by introducing variations to the present data in the form of zoom out operation, adjusting brightness and contrast of the images and creating new images. This led to every class containing around 7200 images leading to class balance. Then, a model was trained on this increased dataset of 50613 images.

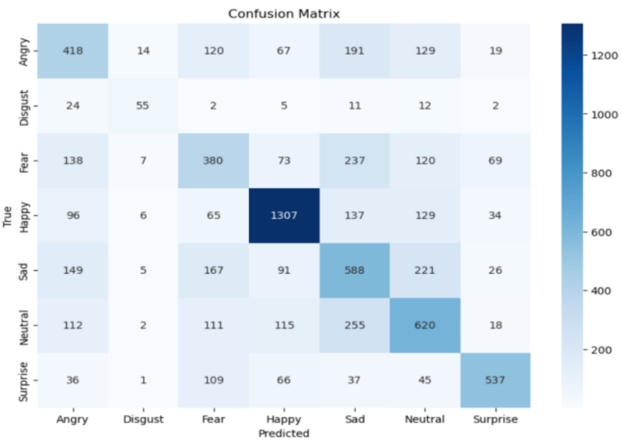

Figure 6. Confusion matrix of the model trained after data augmentation

Figure 6 depicts the confusion matrix of the above model which was trained after increasing the size of the dataset and attaining class balance. Now, there were images that were classified as the 'disgust' class and there were improvements in the correct predictions of the other classes as well, leading to the percentage of correct predictions to be 54.40%.

The above model recorded an accuracy of 95% on the training data whereas recorded much lower accuracy on the test data, indicating overfitting. Therefore, dropout layers were introduced in the fully connected layers to tackle the memorization of the data on which the model is getting trained and the accuracy of the further trained model improved to 57.99%. Figure 7 depicts the confusion matrix of the above model which was trained after introducing the dropout regularization technique.

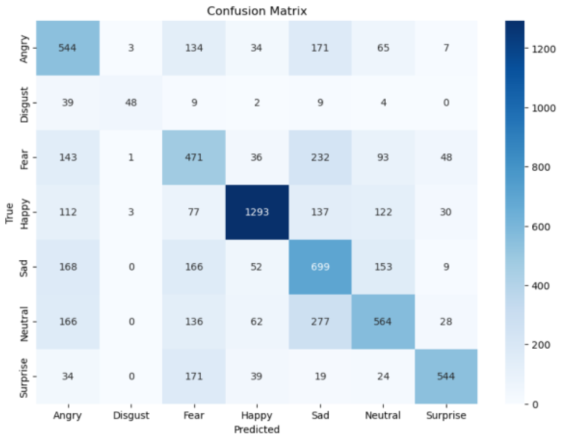

Figure 7. Confusion matrix of the model after introducing dropout layers

Then, residual blocks were introduced to make the information at a layer bypass one or more layers and use it at later layers. Dropout layers were introduced both in the convolutional part and the fully connected part.

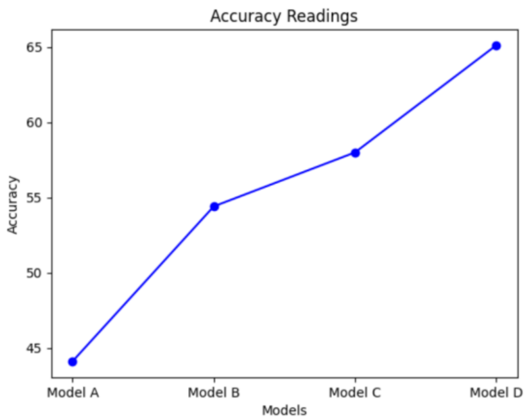

Figure 8. Accuracy readings of different models

The number of filters at each layer in the convolution part were made to increase sequentially which would lead to more and more features being captured in a hierarchical manner. Early stopping method was deployed to further tackle optimization by stopping the process of training when the model started to overfit the data on which it is being trained. The learning rate was reduced whenever the model moves on a plateau region in the loss function which would lead to effective convergence. These were the changes made to attain the proposed model that recorded an accuracy of 65.11%.

Figure 8 shows the different accuracy readings that were recorded in training multiple models which eventually led to the proposed model. Model A denotes the model that was trained before data augmentation. Model B denotes the model that was trained after increasing the size of the dataset to attain class balance. After observing overfitting, model C was trained after including dropout layers in the fully connected part of the model. Model D denotes the proposed model that constitutes residual blocks, dropout layers in both the convolutional part and the fully connected part, early stopping method, and adaptive learning rate method that reduces learning rate in plateau regions to help the model converge better.

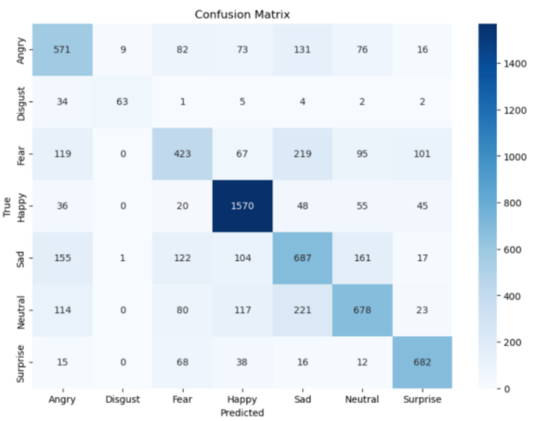

Figure 9. Confusion matrix of the proposed model

The proposed model has recorded an accuracy of 65.11% in recognizing facial expressions from 7178 unseen images. Figure 9 denotes the confusion matrix of the proposed model on the unseen images.

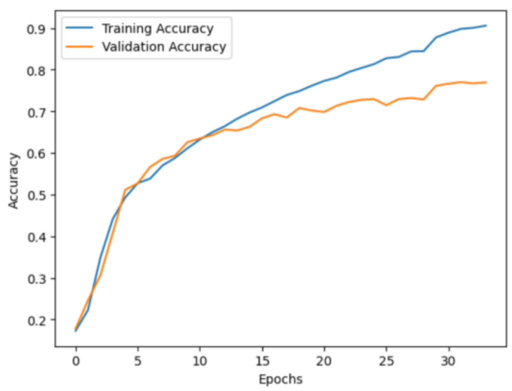

Figure 10. Trends of accuracy on the training data and validation data during training phase

The training dataset is separated into 80% training data and 20% validation data with the training being performed on the training data and the accuracy and loss being monitored on the unseen validation data. Figure 10 shows the trends of the accuracy obtained on the training data and the validation data throughout the training period. Initially both the training accuracy and validation accuracy seem to be of the same values with the validation accuracy even recording a higher accuracy at some stages. As the training progresses, the model starts to extensively learn the training data and therefore, the training accuracy keeps on increasing than the validation accuracy. The training stops with 34 epochs as the loss on the validation data does not decrease for consecutive 10 epochs and the model is saved with the best weights that have been achieved so far to get to the minimum possible loss value.

V. CONCLUSION

The proposed system is an information providing system that not only provides the required information for the user but also recognizes the facial expressions of the user while reading through the provided content. This information about the predominant expression of the user throughout the learning process is used to provide suitable responses and can also be used for the modification of the provided content so as to cater to the needs of the user. This system provides a rich user experience and also aims to provide an effective learning process for the user.

REFERENCES

[1] M. Aly, A. Ghallab and I. S. Fathi, "Enhancing Facial Expression Recognition System in Online Learning Context Using Efficient Deep Learning Model," in IEEE Access, vol. 11, pp. 121419-121433, 2023, doi: 10.1109/ACCESS.2023.3325407.

[2] W. E. Villegas-Ch, J. García-Ortiz and S. Sánchez-Viteri, "Identification of Emotions From Facial Gestures in a Teaching Environment With the Use of Machine Learning Techniques," in IEEE Access, vol.11,pp.38010-38022, 2023, doi: 10.1109/ACCESS. 2023.3267007.

[3] G. Zhao, H. Yang and M. Yu, "Expression Recognition Method Based on a Lightweight Convolutional Neural Network," in IEEE Access, vol. 8, pp.38528-38537, 2020, doi: 10.1109/ACCESS.202 0.2964752.

[4] J. Liu, H. Wang and Y. Feng, "An End-to-End Deep Model With Discriminative Facial Features for Facial Expression Recognition," in IEEE Access, vol. 9, pp.12158-12166, 2021, doi: 10.1109/ACCESS.202 1.3051403.

[5] N. Zhou, R. Liang and W. Shi, "A Lightweight Convolutional Neural Network for Real-Time Facial Expression Detection," in IEEE Access, vol.9, pp. 5573-5584, 2021, doi: 10.1109/ACCESS.2020.30467 15.

[6] J. Pu and X. Nie, "Convolutional Channel Attentional Facial Expression Recognition Network and Its Application in Human–Computer Interaction," in IEEE Access, vol. 11, pp. 129412-129424, 2023, doi: 10.1109/ACCESS.2023.3333381.

[7] P. Ganesan, S. Kumar Jagatheesaperumal, I. Gobhinath, V. Venkatraman, S. N. Gaftandzhieva and R. Z. Doneva, "Deep Learning-Based Interactive Dashboard for Enhancing Online Classroom Experience Through Student Emotion Analysis," in IEEE Access, vol. 12, pp. 91140-91153, 2024, doi: 10.1109/ACCESS.2024.3421282.

[8] M. D'incà, C. Beyan, R. Niewiadomski, S. Barattin and N. Sebe, "Unleashing the Transferability Power of Unsupervised Pre-Training for Emotion Recognition in Masked and Unmasked Facial Images," in IEEE Access, vol. 11, pp. 90876-90890, 2023, doi: 10.1109/ACCESS.2023.3308047.

[9] M. Shi, L. Xu and X. Chen, "A Novel Facial Expression Intelligent Recognition Method Using Improved Convolutional Neural Network," in IEEE Access, vol. 8, pp. 57606-57614, 2020, doi: 10.1109/ACCESS.2020.2982286.

[10] E. Uzun, "A Novel Web Scraping Approach Using the Additional Information Obtained From Web Pages," in IEEE Access, vol. 8, pp. 61726-61740, 2020, doi: 10.1109/ACCESS.2020.2984503.

Pratheepa Paper

ORIGINALITY REPORT

12%	8%	11%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	ijarsct.co.in Internet Source	1%
2	Mouna Kaouni, Fatima Lakrami, Ouidad Labouidya. "Integrating Artificial Intelligence and Natural Language Processing in E-Learning Platforms: A Review of Opportunities and Limitations", 2023 7th IEEE Congress on Information Science and Technology (CiSt), 2023 Publication	1%
3	ijcs.stmikindonesia.ac.id Internet Source	1%
4	Submitted to University of Zambia Student Paper	1%
5	Submitted to University of Bridgeport Student Paper	1%
6	www.ijraset.com Internet Source	1%
7	Ankit Tomar, Saksham Gera, Santosh Kumar, Bhasker Pant. "HHFER: A Hybrid Framework	1%