EMOTIONAL INTELLIGENCE ANALYSIS

Dr. IMTIYAZ KHAN
Assistant Professor CS and AI Dept MJCET OU Hyderabad TS India
MOHAMMED MUBEEN UDDIN
BE CSE AI DS CS and AI Dept MJCET OU Hyderabad TS India
SYED ABDUL MUBASHIR
BE CSE AI DS CS and AI Dept MJCET OU Hyderabad TS India

Abstract: *Emotional Intelligence analysis is a tool used to understand and analyse a person's emotions based on facial recognition. It uses advanced algorithms to detect and interpret facial expressions and provides detailed information about a person's emotional state. The tool used in the number of the current face, including the person's smile and the specific behaviour that emerges. This data is then processed and analysed to create an index bar from depression to happiness that provides an overview of the emotional state of the person being examined. The model Emotional Intelligence Analysis is designed to be versatile at the same time, making it ideal for use in groups or public places where searching and interpreting emotions is difficult.In addition, the device captures the emotions of mask wearers, as well as easy in Multiple faces Emotion Recognition. Overall, emotional analysis is a powerful tool for understanding and interpreting a person's emotional state. Using facial recognition, the device can provide a better understanding of a person's emotions, making it useful for many applications in psychology, medicine, and business.*

INTRODUCTION

Emotional Intelligence Analysis is a rapidly evolving field that has gained considerable interest due to its potential to provide valuable insights into human behaviour and decision-making processes. One of the most promising techniques in Emotional Intelligence analysis is facial emotional recognition, which involves the use of computer vision algorithms to detect and classify human emotions based on facial expressions.The ability to accurately perceive and understand the emotions of oneself and others, and to regulate one's own emotions accordingly, is crucial for effective communication, social interactions, and decision-making. Facial Emotion Recognition is a technology used for analysing sentiments by different sources, such as pictures and videos. It belongs to the family of technologies often referred to as 'affective computing', a multidisciplinary field of research on computer's capabilities to recognize and interpret human emotions and affective states and it often builds on Artificial Intelligence technologies. Facial expressions are forms of non-verbal communication, providing hints for human emotions.

With the increasing use of artificial intelligence (AI), interest in emotional intelligence is also increasing. To improve human-computer interaction (HCI) and make it more effective, machines need to be able to understand their environment, especially the thoughts of the same people. With the help of cameras and sensors, machines can detect the state of the environment. In recent years, deep learning (DL) algorithms have proven to be very effective in capturing environmental states. Emotional intelligence is important for machines to better serve their purposes, as it provides information about the inner state of humans.Deep learning techniques can be used to analyse facial images to determine a person's mood. This can lead to more efficient and effective interactions between humans and machines and can have a significant impact in many applications such as healthcare, education and customer-facing services. For this reason, the development of artificial intelligence using artificial intelligence has gained importance in recent years.

FACIAL EMOTIONAL RECOGNITION

FER typically has four steps. The first is to detect a face in an image and draw a rectangle around it and the next step is to detect landmarks in this face region. The third step is extracting spatial and temporal features from the facial components. The final step is to use a Feature Extraction (FE) classifier and produce the recognition results using the extracted features. Figure 1.1 shows the FER procedure for an input image where a face region and facial landmarks are detected. Facial landmarks are visually salient points such as the end of a nose, and the ends of eyebrows and the mouth as shown in Figure 1.2. The pairwise positions of two landmark points or the local texture of a landmark are used as features. Table 1.1 gives the definitions of 64 primary and secondary landmarks . The spatial and temporal features are extracted from the face and the expression is determined based on one of the facial categories using pattern classifiers.

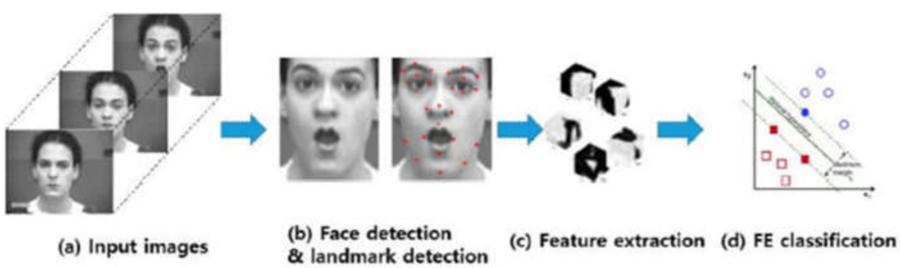


Figure 1.1 FER procedure for an image [9].

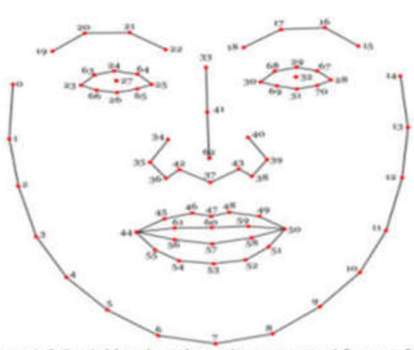


Figure 1.2 Facial landmarks to be extracted from a face.

Primary landmarks		Secondary landmarks	
Number	Definition	Number	Definition
16	Left eyebrow outer corner	1	Left temple
19	Left eyebrow inner corner	8	Chin tip
22	Right eyebrow inner corner	2-7,9-14	Cheek contours
25	Right eyebrow outer corner	15	Right temple
28	Left eye outer corner	16-19	Left eyebrow contours
30	Left eye inner corner	22-25	Right eyebrow corners
32	Right eye inner corner	29,33	Upper eyelid centers
34	Right eye outer corner	31,35	Lower eyelid centers
41	Nose tip	36,37	Nose saddles
46	Left mouth corner	40,42	Nose peaks (nostrils)
52	Right mouth corner	38-40,42-45	Nose contours
63,64	Eye centers	47-51,53-62	Mouth contours

Table 1.1 Definitions of 64 primary and secondary landmarks.

The DL-based FER method directly learns the end of the input image, reducing reliance on facial physics-based models and other methods. Among the DL models, Convolutional Neural Networks (CNNs) are the most popular. With CNN, the input image is filtered by a convolutional algorithm to create a unique map. This map is then fed into a fully connected process and according to the results of the finite classifier, the face map is considered to belong to the group. The data used for this model is the Facial Emotion Recognition 2013 (FER 2013) dataset.This is an open source document created for a project and then shared publicly for a Kaggle competition. It has 35,000 face images in 48×48 grayscale with various emotions. Five emotions were used for this project: happiness, anger, neutrality, sadness and fear. We used a mental test that detects many faces in a photo and calculates their emotions. Bar analyses faces and uses special formulas to find out happiness, sadness, anger, surprise, etc. gives points for emotions.Results are shown moving from negative to positive, allowing users to easily compare and understand the emotions in the images. This capability provides more insightful analytics that can be used in a variety of applications such as market research, client advocacy, and human resource management.

DATASET PREPARATION

The FER 2013 dataset is well known and was used in the Kaggle competition. The data must be prepared for input to the CNN because there are some issues with this dataset as discussed below. The input to the model should be an array of numbers, so images must be converted into arrays.

Some dataset challenges are given below:

- **Imbalance**: Imbalance is when one class has many more images than another class. This results in the model being biassed towards one class. For example, if there are 2000 images for the happy expression and 500 images for the fear expression, then the model will be biassed towards the happy expression. Data augmentation is done to avoid this problem. Data augmentation increases the amount of data using techniques like cropping, padding, and horizontal flipping.

- **Contrast variation:** Some images in the dataset can be too dark and some can be too light. Since images contain visual information, higher contrast images have more information than lower contrast images. A CNN takes images as input, automatically learns image features and classifies the images into output classes. Thus, variations in image contrast affect CNN performance. This problem can be solved by changing the images to focus on the faces.
- **Intra-class variation:** Some images in the dataset are not human faces as there are drawings and animated faces. The features in real and animated faces differ and this creates confusion when the model is extracting landmark features. Model performance will be better if all images in the dataset are human faces so other images should be removed.

The images used for training should be free from the above issues. Thus, manual filtering of the 35,000 images in the FER 2013 dataset was done and 7,074 images from five classes were selected, 966 for angry, 859 for fear, 2477 for happy, 1466 for neutral and 1326 for sad.

Python Libraries Used :

NumPy: Numerical Python (NumPy) is an open source Python library used for working with arrays and matrices. An array object in NumPy is called nd.array. CNN inputs are arrays of numbers and NumPy can be used to convert images into NumPy arrays to easily perform matrix multiplications and other CNN operations.

OpenCV: OpenCV is an open source library for CV, ML and image processing. Images and videos can be processed by OpenCV to identify objects, faces and handwriting. When it is integrated with a library such as NumPy, OpenCV can process array structures for analysis. Mathematical operations are performed on these array structures for pattern recognition.

Dlib: Dlib v19.2 uses a Maximum-Margin Object Detector (MMOD) with CNN based features. Training with this library is simple and a large amount of data is not needed. After labelling landmarks in an image, it learns to detect them. This also has an inbuilt shape and frontal face detector.

Math: Common mathematical functions are defined in the math library. These include trigonometric functions, representation functions, logarithmic functions and angle conversion functions. CNN operations require adding and multiplying arrays which can be done using this library.

Image to arrays :

An image is represented by values (numbers) that correspond to the pixel intensities. The array module in NumPy (nd.array) is used to convert an image into an array and obtain the image attributes. Figure 2.2 shows an image in the sad class from the FER 2013 dataset converted into a NumPy array.

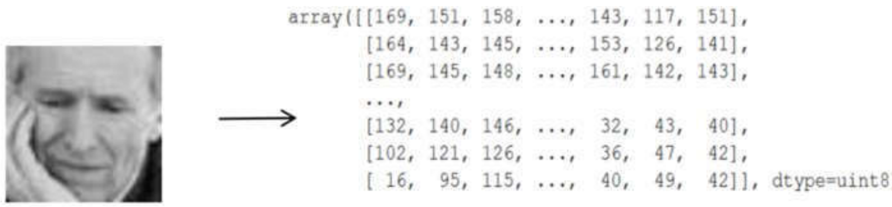


Figure 2.2 Sad image from the FER 2013 dataset converted into an array.

Image to landmarks :

The Dlib library is used to detect facial landmarks. This process consists of two steps, localising the face in an image and detecting the facial landmarks. The frontal face detector from Dlib is used to detect the face in an image. A rectangle on the face is obtained which is defined by the top left corner and the bottom right corner coordinates. The Dlib shape predictor is used to extract the key facial features from an input image. An object called landmarks which has two arguments is passed. The first argument is an image in which faces will be detected and the second specifies the area where the facial landmarks will be obtained. This area is represented by the coordinates of the rectangle. Figure 2.3 shows the 64 landmarks detected in an image.

Figure 2.3 Landmarks detected on a face.

CONVOLUTIONAL NEURAL NETWORKS

The fundamental building block of a NN is a neuron. Figure 3.1 shows the structure of a neuron. Forward propagation of information through a neuron happens when inputs X1 to Xm are multiplied by their corresponding weights and then added together. This result is passed through a nonlinear activation function along with a bias term which shifts the output. The bias is shown as in Figure 3.1. For an input vector X= X1,X2 ,X3…,Xm and weight vector W=W1 W2,W3 ,...Wm, the neuron output is $\hat{Y} = g(W0+\sum Xi.Wi)$. The output is between 0 and 1 which makes it suitable for problems and with probabilities. The purpose of the activation function is to introduce nonlinearities in the network since most real world data is nonlinear. The use of a nonlinear function also allows NNs to approximate complex functions.

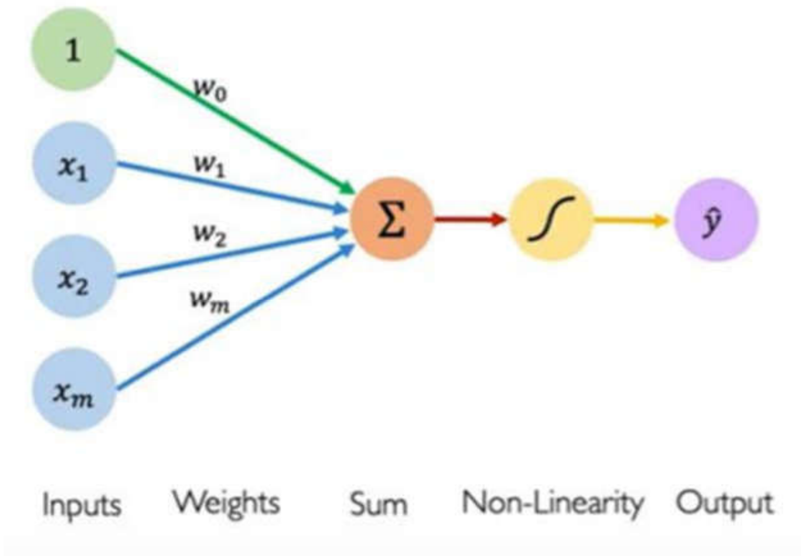


Figure 3.1 The basic structure of a neuron

Neurons can be combined to create a multi output NN. If every input has a connection to every neuron it is called dense or fully connected. Figure 3.2 shows a dense multi output NN with two neurons. A deep NN has multiple hidden layers stacked on top of each other and every neuron in each hidden layer is connected to a neuron in the previous layer. Figure 3.3 shows a fully connected NN with 5 layers.

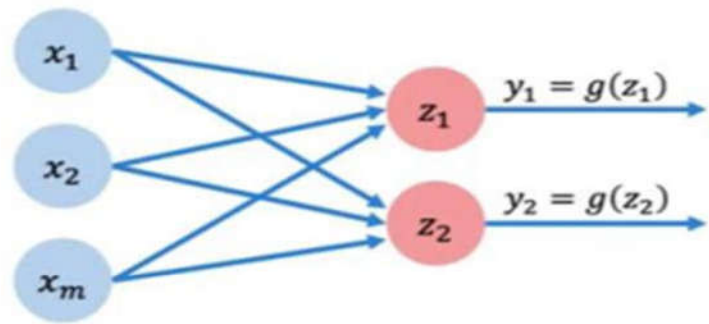


Figure 3.2 A multi output NN with two neurons

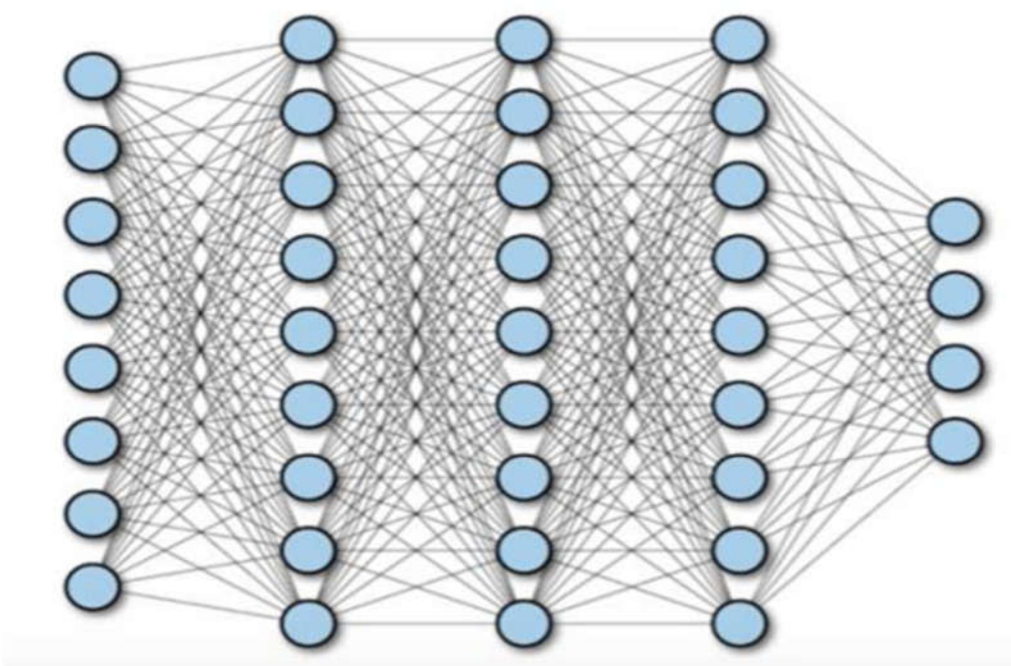


Figure 3.3 A fully connected NN

CNN concept:

A CNN is a DL algorithm which takes an input image, assigns importance (learnable weights and biases) to various aspects/objects in the image and is able to differentiate between images. The preprocessing required in a CNN is much lower than other classification algorithms. Figure 3.4 shows the CNN operations. The architecture of a CNN is analogous to that of the connectivity pattern of neurons in the human brain and was inspired by the organisation of the visual cortex. One role of a

CNN is to reduce images into a form which is easier to process without losing features that are critical for good prediction. This is important when designing an architecture which is not only good at learning features but also is scalable to massive datasets. The main CNN operations are convolution, pooling, batch normalisation and dropout which are described below.

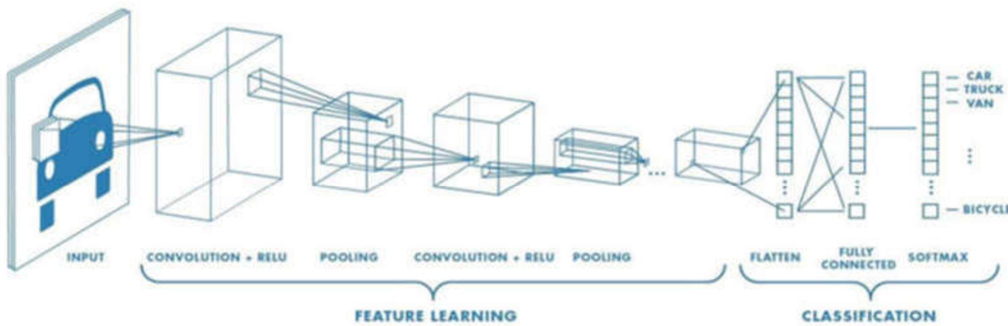


Figure 3.4 The CNN operations

CNN architecture:

ML models can be built and trained easily using a high-level Application Programming Interface (API) like Keras. In this report, a sequential CNN model is developed using TensorFlow with the Keras API since it allows a model to be built layer by layer. TensorFlow is an end-to-end open source platform for ML. It has a flexible collection of tools, libraries and community resources to build and deploy ML applications. Figure 3.5 shows the structure of a CNN where conv. denotes convolution.

The CNN Model has four phases. At the end of each phase, the size of the input image is reduced. The first three phases have the same layers where each starts with a convolution and ends with dropout. The first phase of the model has an input layer for an image of size 48 × 48 (height and width in pixels) and convolution is performed on this input. The convolution parameters are the same for all convolution layers in the network except the number of kernels. An He-normal initializer is used which randomly generates appropriate values for the kernel. The number of kernels is 64 in the first phase. Then, batch normalisation is performed to obtain the inputs to the next layer. Convolution and batch normalisation are repeated in the following layers. In the next layer, max pooling is performed with pool size 2 × 2, so the output size is 24 × 24. Dropout is performed next at a rate of 0.35. The second phase has 128 kernels and 0.4 dropout rate. Max pooling in the second phase gives an output of size 12 × 12. The third phase has 256 kernels with 0.5 dropout rate. Max pooling in the third phase reduces the size of the output to 6 × 6. The final phase starts with a flatten layer followed by dense and output layers. Classifying the five emotions requires the data to be a one-dimensional array. The flatten layer converts the two-dimensional data into a one-dimensional array. The flattened output is fed to the dense layer which applies the SoftMax function.

Training the model:

To train the model, the train-test split() function is used. This function splits the dataset into training and testing sets. The training data is not used for testing. A training ratio of 0.90 means 90% of the dataset will be used for training and the remaining for testing the model. The Learning Rate (LR) is a configurable parameter used in training which determines how fast the model weights are calculated. A high LR can cause the model to converge too quickly while a small LR may lead to more accurate weights (up to convergence) but takes more computation time. The number of epochs is the number of times a dataset is passed forward and backward through the NN. The dataset is divided into batches to lower the processing time and the number of training images in a batch is called the batch size

TEXT ON DISPLAY

In our implementation of emotional intelligence analysis, we have used OpenCV to count the number of faces and smiles in the input frames. This information is then displayed in the output frame along with the bounding boxes around each detected face. We start by checking for the required libraries and opening the default camera to capture faces. We use the dlib library to get the coordinates of each face and count the number of faces in the frame.

To detect the faces, we use the Haar-cascades classifier, which is a pre-trained model for detecting features in images. We use face, eye, and smile Haar-cascades for our implementation, which need to be downloaded and placed in the working directory. We process each frame of the live feed from the camera, convert it to grayscale, and apply the Haar-cascade classifiers to detect the faces.

For each detected face, we check for smiles using OpenCV's smile detection. We use the cv2.rectangle function to draw the bounding box around each detected face in the output frame, along with the smile count for that face. We define the region of interest for the face using roi_gray and roi_color, which allows us to isolate the face in the grayscale image and the original frame, respectively. Overall, our implementation uses a combination of computer vision and machine learning techniques to detect faces and smiles in the input frames, providing a powerful tool for emotional intelligence analysis

ANALYSIS BAR

The emotional intelligence analysis model we have developed provides an index bar that displays the emotional state of the people in the input frames. This index bar is placed at the bottom of the screen and is calculated based on the number of smiles, the number of faces, and their respective emotions. We divide the sum of emotional states and smiles by the total number of faces, resulting in a percentage that is used to determine the position of the index bar. The bar is then mapped to different colour codes representing various emotional states, including light green for happiness, red for sadness, yellow for neutrality, dark green for excitement, and orange for shock.

The index bar works for multiple faces, allowing us to determine the emotional state of all people in the input frames. The bar is developed using OpenCV, making use of its functionalities for drawing bars and handling colours. The emotional intelligence analysis model provides a useful tool for understanding the emotional states of people and can be used in various applications such as in customer service, healthcare, and entertainment.

RESULTS AND DISCUSSION

Evaluation metrics:

Accuracy, loss, precision, recall and F-score are the metrics used to measure model performance. These metrics are defined below.

Accuracy: Accuracy is given by

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Loss: Categorical cross-entropy is used as the loss function and is given by

$$\text{Loss} = -\sum_{c=1}^{m}\left(y_{o,c}\log(p_{o,c})\right)$$

where y is a binary indicator (0 or 1), p is the predicted probability and m is the number of classes (happy, sad, neutral, fear, angry)

Confusion matrix: The confusion matrix provides values for the four combinations of true and predicted values, True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). Precision, recall and F-score are calculated using TP, FP, TN, FN. TP is the correct prediction of an emotion, FP is the incorrect prediction of an emotion, TN is the correct prediction of an incorrect emotion and FN is the incorrect prediction of an incorrect emotion. Consider an image from the happy class. The confusion matrix for this example is shown in Figure 4.1. The red section has the TP value as the happy image is predicted to be happy. The blue section has FP values as the image is predicted to be sad, angry, neutral or fearful. The yellow section has TN values as the image is not sad, angry, neutral or fear but the model predicted this. The green section has FN values as the image is not happy but was predicted to be happy

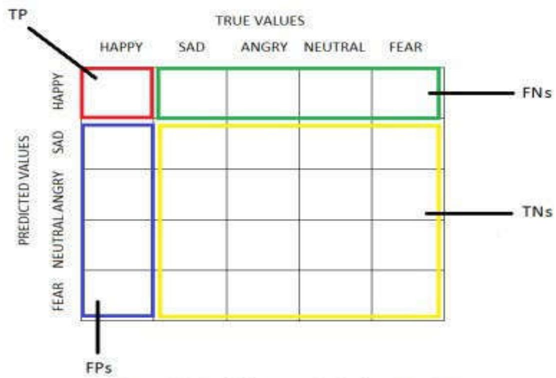


Figure 4.1 Confusion matrix for five emotions.

Recall: Recall is given by

Recall = TP /(TP+FN)

Precision: Precision is given by

Precision = TP/(TP+FP)

F-score: F-score is the harmonic mean of recall and precision and is given by

F-score= (2xRecallxPrecision) /(Recall+Precision)

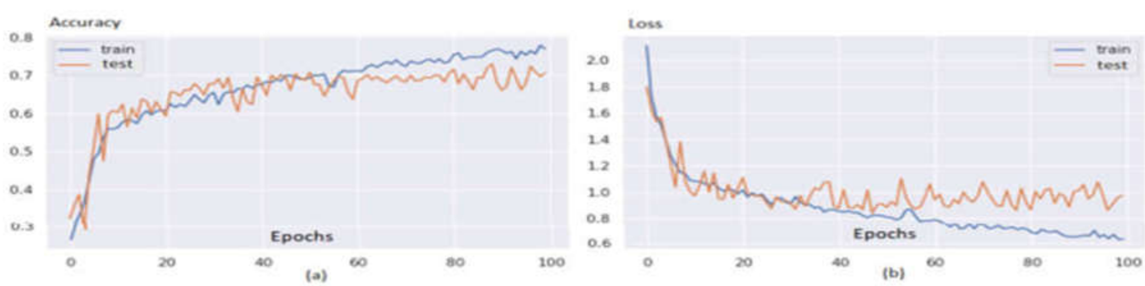


Figure 4.2 Results of one trial after training Model 1 for 100 epochs (a) accuracy and (b) loss.

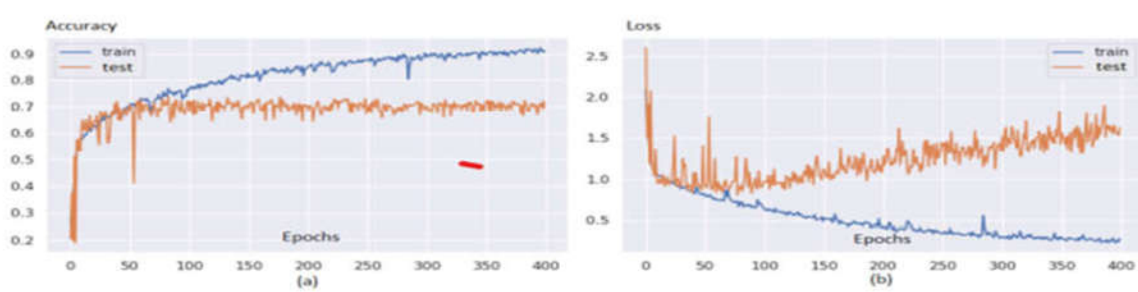
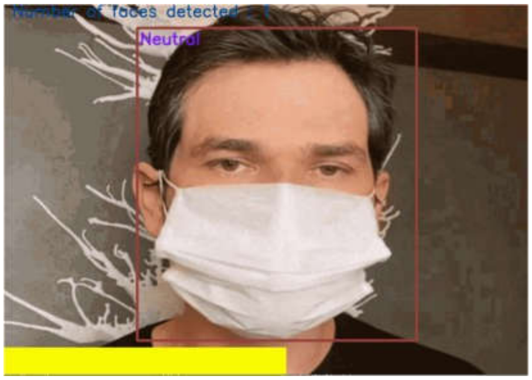
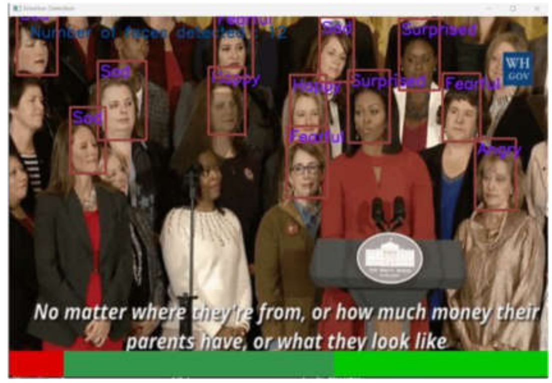
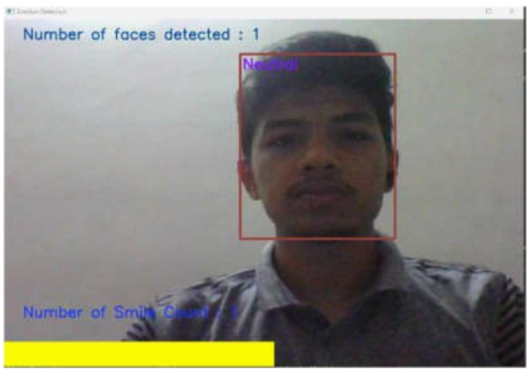


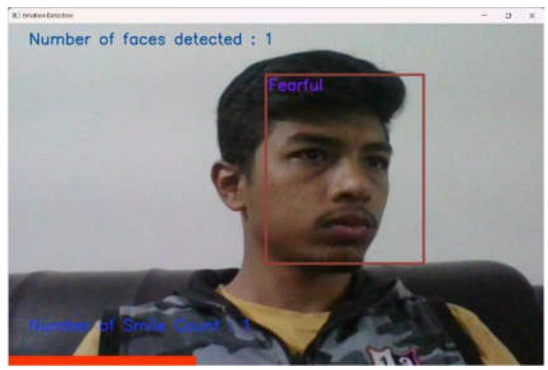
Figure 4.3 Results of one trial after training Model 1 for 400 epochs (a) accuracy and (b) loss.

Output Images :









CONCLUSION

In conclusion, the development of CNN models for facial expression recognition is a promising field of research. In this study, two CNN models were developed and evaluated for their accuracy in recognizing emotions from facial images. The models were trained and tested using the FER 2013 dataset, which included seven different emotions. The results showed that the models performed well, with an average accuracy of 80.0% and average loss of 0.63 for the five emotions and 72.2% and average loss of 0.85 for the seven emotions. Happy emotion had the best precision, recall and F Score of 0.88 for CNN Model and precision 0.87, recall 0.90 and F-score of 0.88 for CNN Model. The neutral emotion had the worst performance with precision 0.56, recall 0.64 and F-score 0.60 for CNN Model and precision 0.56, recall 0.68 and F-score 0.61 for Model . These findings suggest that facial expression recognition can be used as a measure of emotional intelligence and that controlling cognitive ability is important in studies of ability EI. Moreover, the proposed models outperformed other approaches in the literature, demonstrating their potential for real-world applications. As the field of AI continues to advance rapidly, emotional intelligence analysis will become increasingly important for improving human-computer interaction and making it more natural. The use of DL techniques such as CNNs, combined with emotional intelligence analysis, holds great promise for enhancing the ability of machines to understand the emotional states of humans and better serve their purpose.

FUTURE WORK:

Emotional Intelligence Analysis is a promising field that can greatly benefit from the incorporation of other types of neural networks such as Recurrent Neural Networks (RNNs). To improve accuracy, techniques such as the Capsnet algorithm for pattern recognition could be considered for feature extraction. While DL-based approaches require a large labelled dataset, significant memory, and long training and testing times, simpler solutions should be developed with lower data and memory requirements to improve the accuracy and robustness of facial expression recognition models, especially for more complex emotions. This will pave the way for exploring the applications of such models in fields such as healthcare, education, and entertainment, among others. Overall, Emotional Intelligence Analysis is a rapidly evolving field with great potential for future advancements and developments.

REFERENCES

[1] Multi angle optimal pattern-based deep learning for automatic facial expression recognition – Deepak Kumar Jain,Zhang Zhang, Kaiqi Huang.

[2] Facial Emotion Recognition Using Transfer Learning in the Deep CNN- M. A. H. Akhand, Shuvendu Roy, Nazmul Siddique, Md Abdus Samad Kamal, Tetsuya Shimamura .

[3] Real Time Expression Detection of Multiple Faces Using Deep Learning- Shreya Mishra, Shubham Kumar, Shubham Gautam, Jaspreet Kour.