Introduction

We propose a model for online education that utilizes live input from students to provide real-time feedback to teachers, enabling dynamic adaptation of teaching methods based on student comprehension levels. Data preprocessing involves loading and normalizing images into numpy arrays and tensors, followed by encoding into category vectors. The neural network architecture comprises convolution layers for feature extraction and dense linear layers for classification, with a higher dropout rate to prevent overfitting. We aim to make a sophisticated model with more than 4 million parameters.

Literary Review

PCA and Deep Learning

A novel method combines Principal Component Analysis (PCA) for dimensionality reduction in facial emotion recognition with deep learning for optimal facial feature extraction. Because significant variation is preserved, using PCA's compression function on deep learning features increases accuracy and computing efficiency. Empirical findings exhibit the method's potential to improve facial emotion detection systems, as it outperforms direct feature extraction with the VGG-Face model [6].

Face Detection and Recognition

This paper highlights face detection and recognition, addressing the need for automated comprehension and assessment of image and video datasets, particularly in face identification, appearance recognition, and human-computer interaction applications. Emphasizing the significance of facial recognition in biometrics [1].

Iris Recognition

The research provides a noise reduction technique for iris recognition systems to target localized high-frequency information, such as eyelashes and eyelids in segmented iris areas utilizing radial suppression. This method performs better in obtaining lower equal error rates, improves segmentation accuracy, and includes an automated iris identification system prototype. It also uses a one-dimensional Log Gabor wavelet for iris feature extraction[2].

Emotion Recognition Algorithms

The paper compares five algorithms for real-time emotion recognition from facial images, including deep learning (AlexNet, Affdex CNN, FER-CNN) and conventional methods (SVM, MLP) utilizing Histogram of Oriented Gradients features. It evaluates these approaches on four basic emotions. It demonstrates the superiority of the proposed FER-CNN model, highlighting its potential for diverse applications such as medicine, e-learning, entertainment, and marketing[10].

Research Gaps

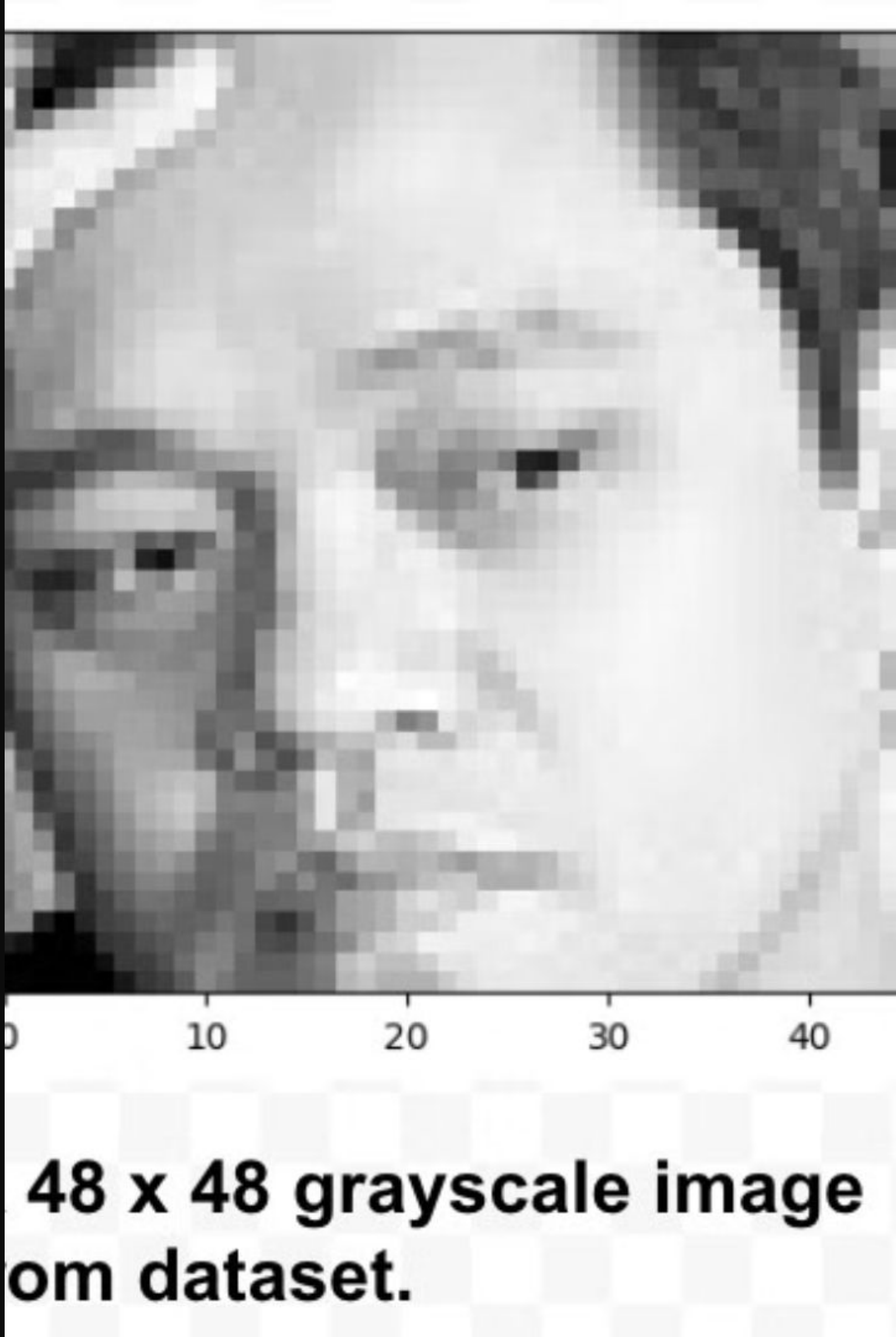
Limited Dataset: The major research gap is lack of reliable dataset. The solution for this is to generate the dataset . One of the other possible solution to the above is make a dataset. One of the best dataset yet has about 36,000 images split in 7 classes. other datasets have videos which presents its own challenge of extracting the faces.

A sample subset showing all the classes.

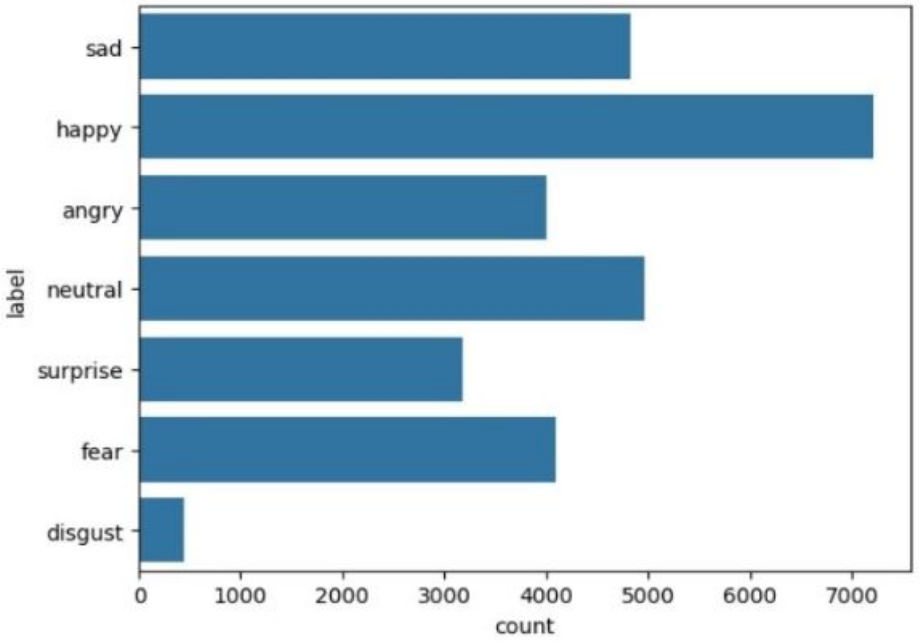
Dataset

The dataset is called Facial Expression Dataset (FER-2013). The dataset contains 7 classes of emotions, **Fear, Neutral, Angry, Happy, Surprise, Disgust, Sad.** The dataset was divided into two parts: train and test.

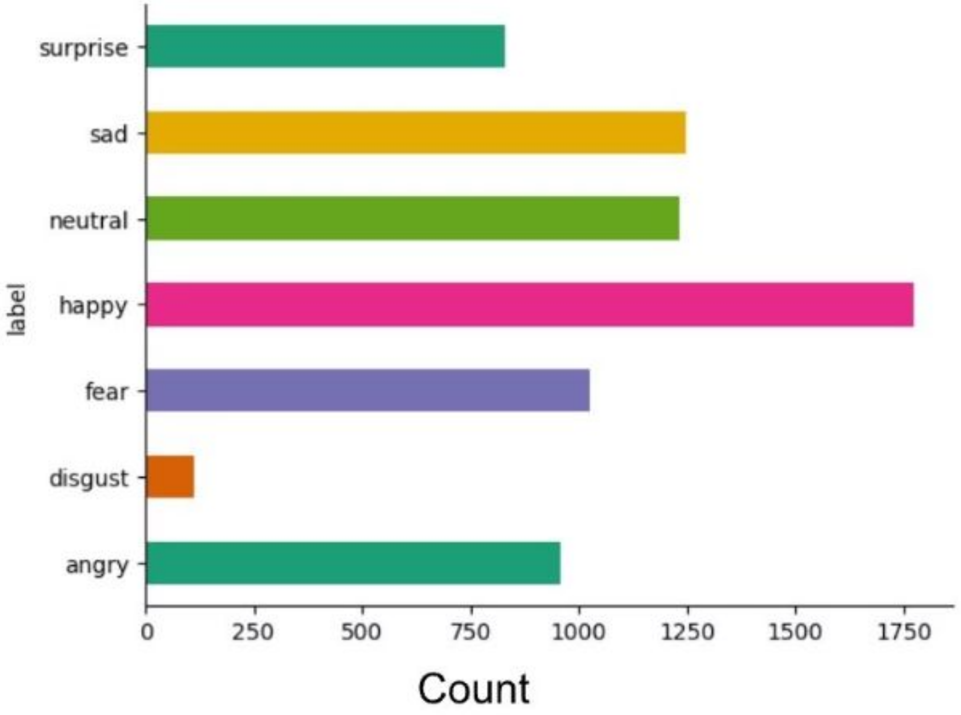
The train : test split was 80 : 20 %(28709 : 7178).



Data set distribution



This graph shows the distribution of various classes among train directory.

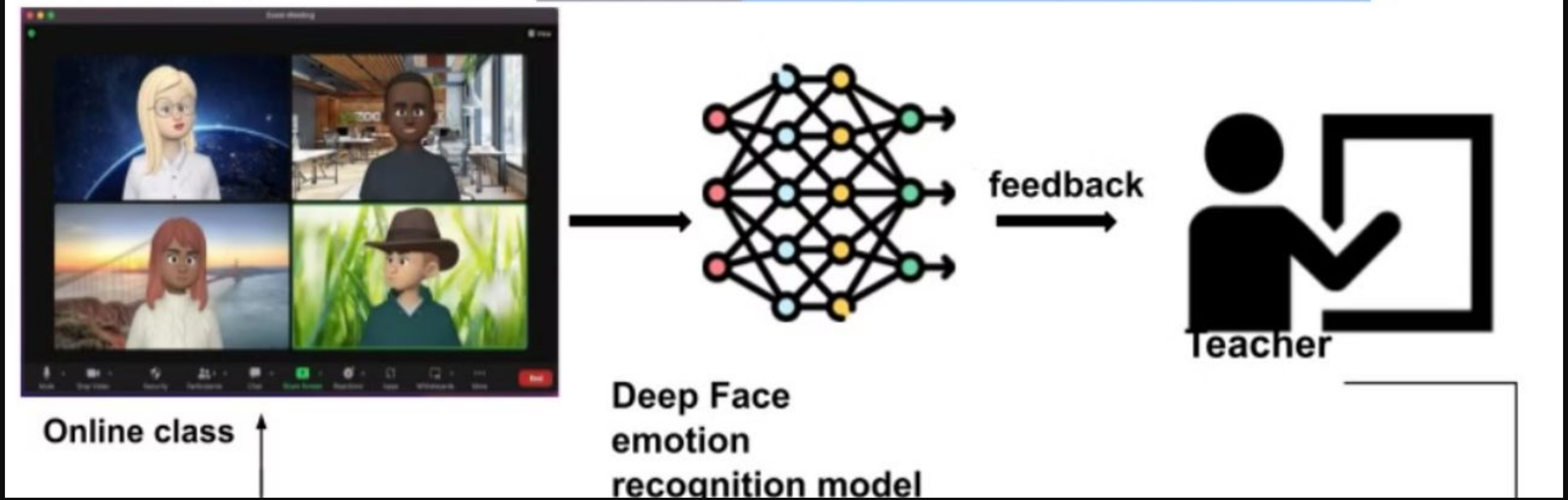


Test dataset distribution.

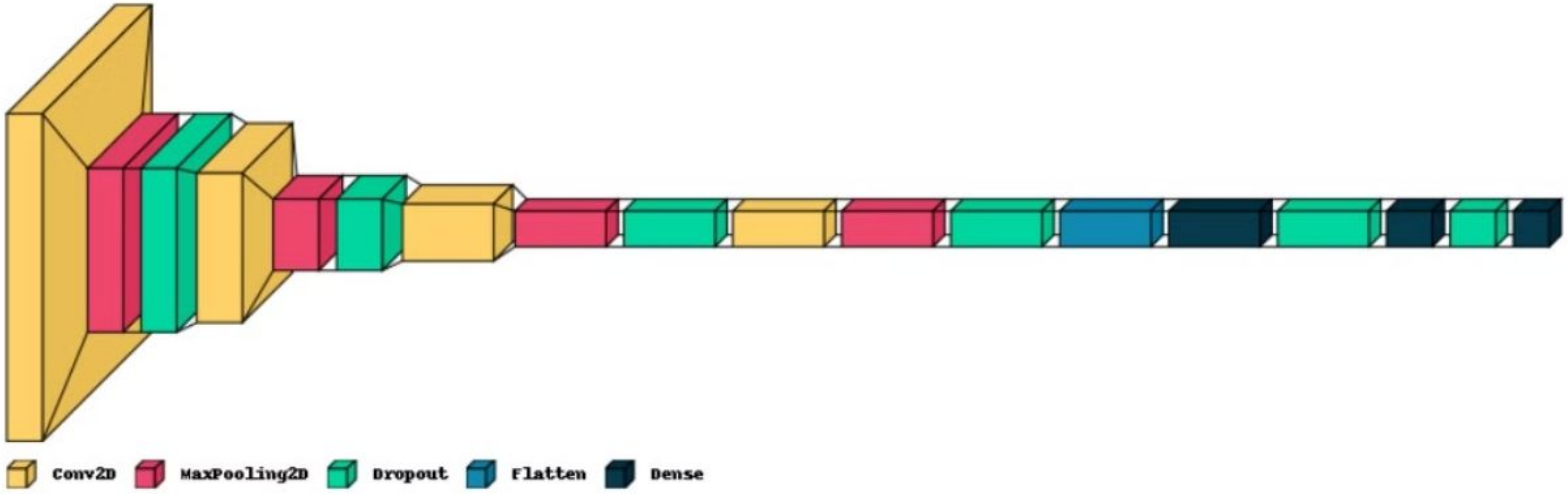
Methodology

We propose a model which can take live inputs from the class and then give feedback to the teacher such that the teacher can dynamically engage the class based on the feedback from deep neural networks. This is more suitable for online classes where understanding the student requirements is vastly different from offline classes. If a tough concept causes the students to be confused then the teacher can immediately get feedback and immediately change the methodology of teaching.

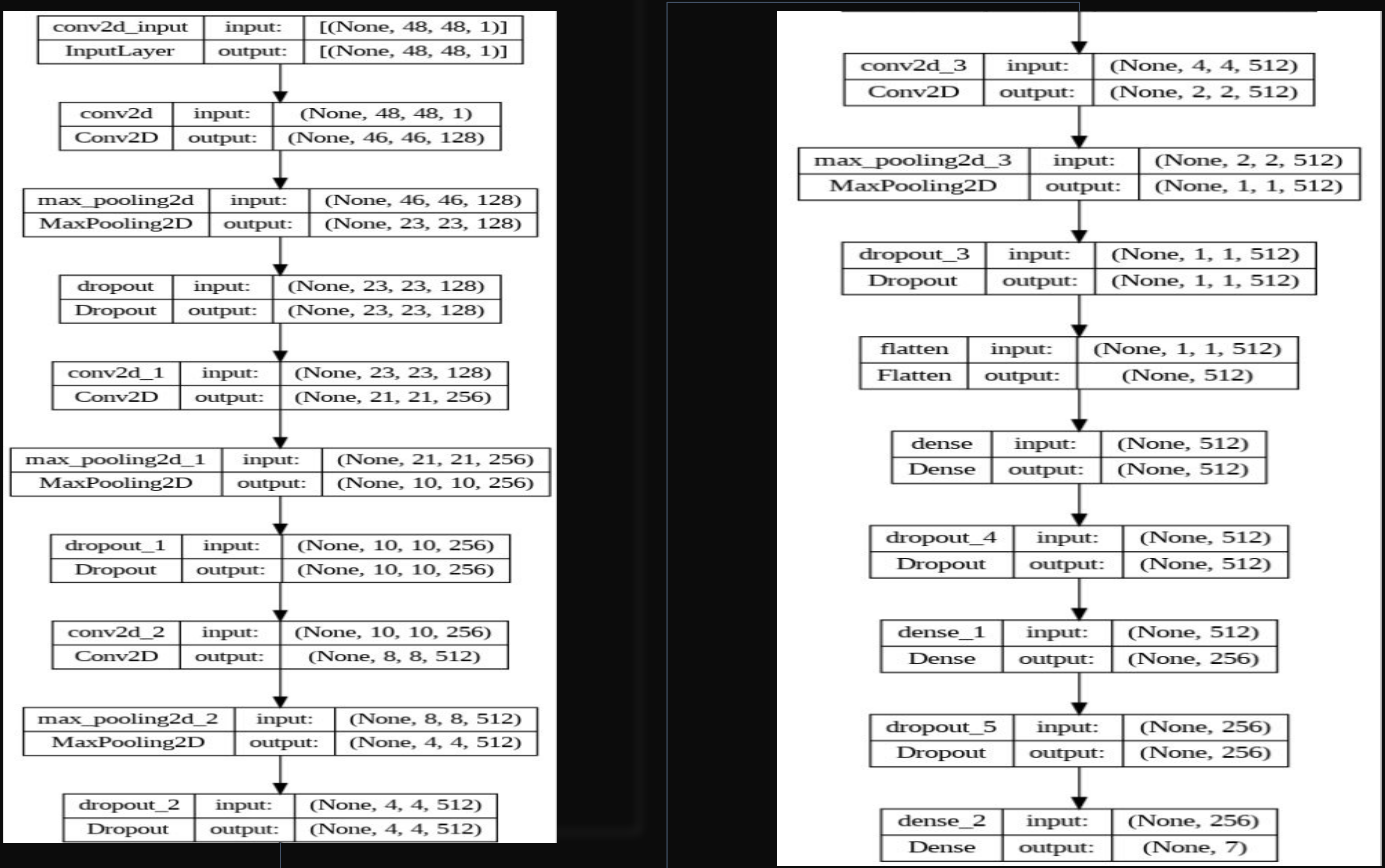
All the images were loaded to a numpy array then to tensor. Each image was normalized. We then encoded the images into category vectors.


Online class
Deep Face
emotion
recognition model
feedback
Teacher

Deep CNN Model:



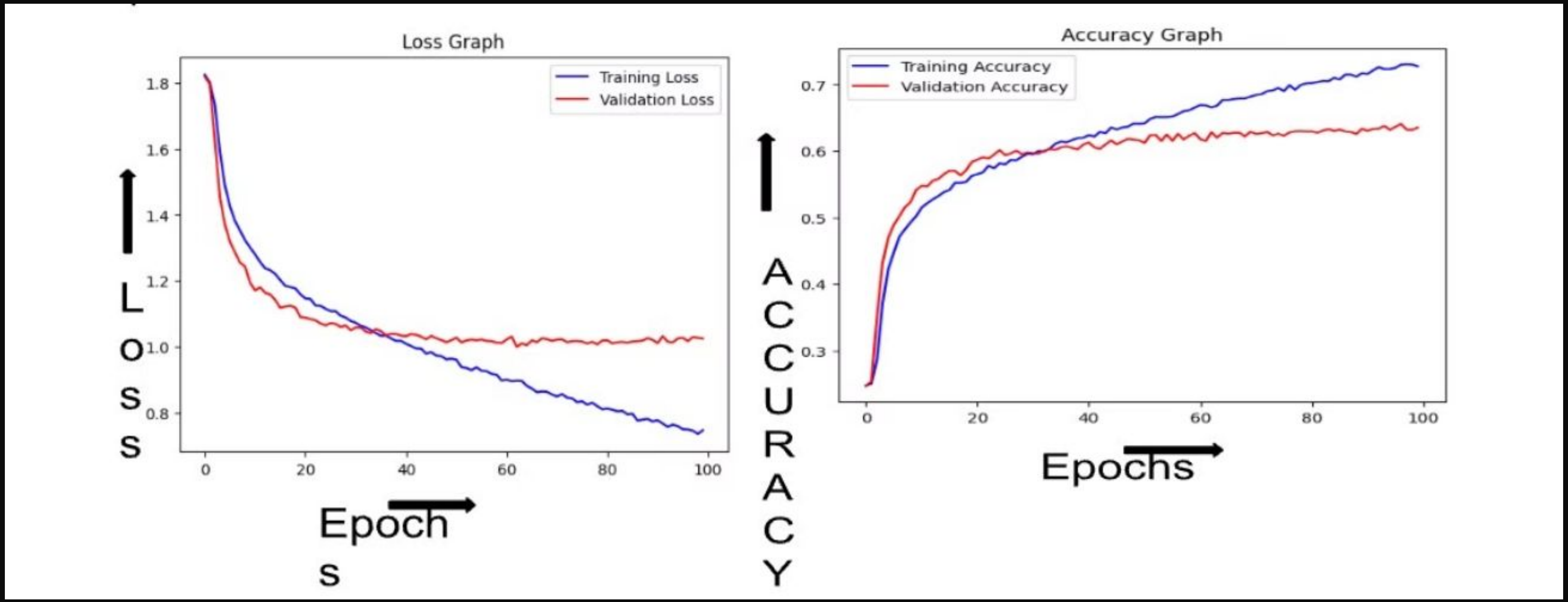
Visual representation of the model architecture.


conv2d_input | input: | [(None, 48, 48, 1)]
InputLayer | output: | [(None, 48, 48, 1)]
conv2d | input: | (None, 48, 48, 1)
Conv2D | output: | (None, 46, 46, 128)
max_pooling2d | input: | (None, 46, 46, 128)
MaxPooling2D | output: | (None, 23, 23, 128)
dropout | input: | (None, 23, 23, 128)
Dropout | output: | (None, 23, 23, 128)
conv2d_1 | input: | (None, 23, 23, 128)
Conv2D | output: | (None, 21, 21, 256)
max_pooling2d_1 | input: | (None, 21, 21, 256)
MaxPooling2D | output: | (None, 10, 10, 256)
dropout_1 | input: | (None, 10, 10, 256)
Dropout | output: | (None, 10, 10, 256)
conv2d_2 | input: | (None, 10, 10, 256)
Conv2D | output: | (None, 8, 8, 512)
max_pooling2d_2 | input: | (None, 8, 8, 512)
MaxPooling2D | output: | (None, 4, 4, 512)
dropout_2 | input: | (None, 4, 4, 512)
Dropout | output: | (None, 4, 4, 512)
conv2d_3 | input: | (None, 4, 4, 512)
Conv2D | output: | (None, 2, 2, 512)
max_pooling2d_3 | input: | (None, 2, 2, 512)
MaxPooling2D | output: | (None, 1, 1, 512)
dropout_3 | input: | (None, 1, 1, 512)
Dropout | output: | (None, 1, 1, 512)
flatten | input: | (None, 1, 1, 512)
Flatten | output: | (None, 512)
dense | input: | (None, 512)
Dense | output: | (None, 512)
dropout_4 | input: | (None, 512)
Dropout | output: | (None, 512)
dense_1 | input: | (None, 512)
Dense | output: | (None, 256)
dropout_5 | input: | (None, 256)
Dropout | output: | (None, 256)
dense_2 | input: | (None, 256)
Dense | output: | (None, 7)

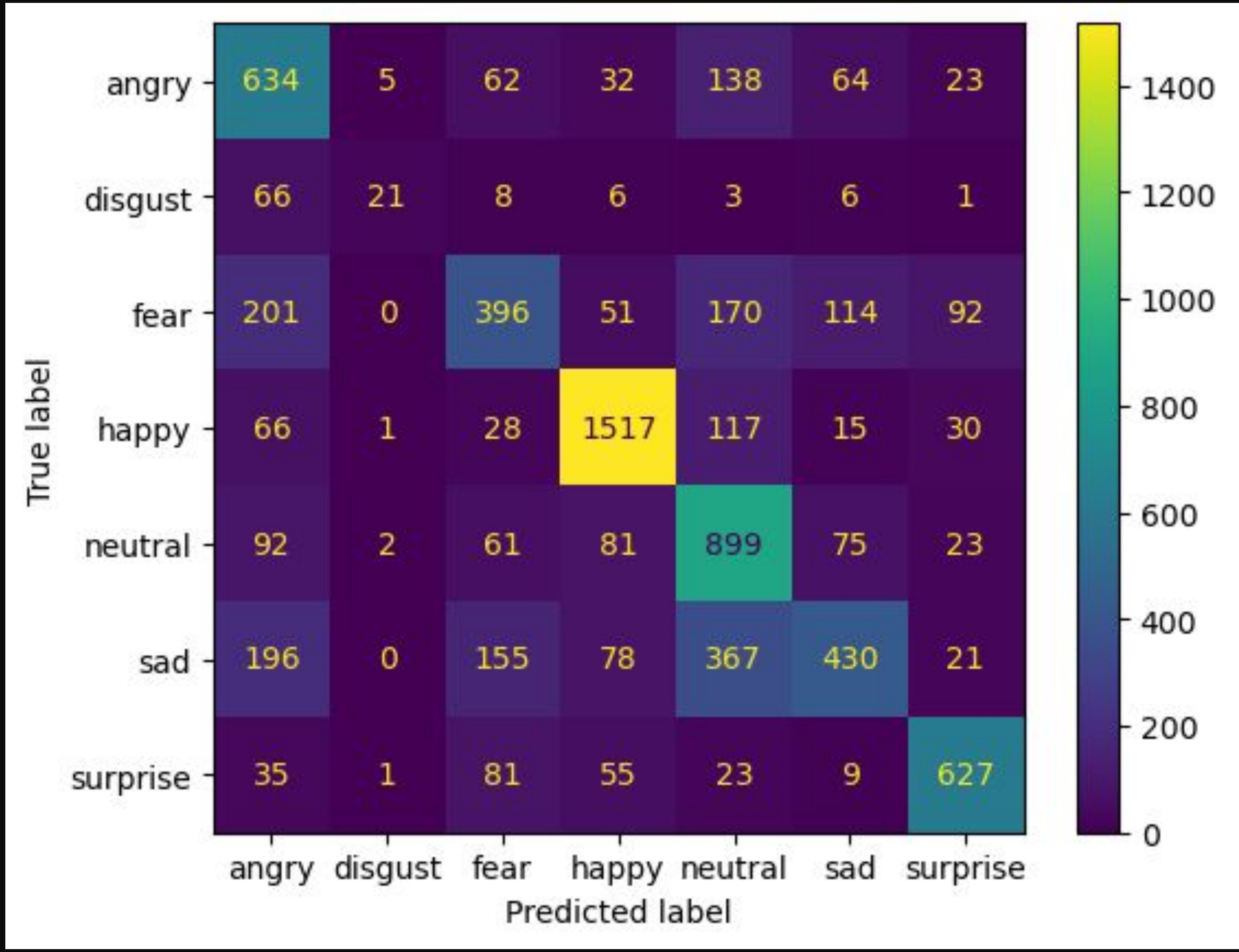
Cont..

The methodology to make the neural network is fairly intuitive. The first few Convolution layers of the neural network are for feature extraction and the next dense linear layers are dedicated to classification. In the implementation of this model, to curb overfitting and to optimize training, a dropout of 40 % was chosen as we have a fairly large neural network with over 4,232,199 trainable parameters. Convolution kernel size of 3 x 3 was chosen as it is a common practice to choose a smaller size and then try with larger kernel size of 5 x 5 or 11 x 11. To introduce non linearity the standard Rectified Linear Unit (ReLU) was opted for. The convolution layer and the max pooling layer kept decreasing the image from 48 x 48 to 1 x 1 while doubling the feature maps every convolution layer. Starting from 128 feature maps in the first convolution layer to 512 in the third convolution layer doubling every convolution layer. The later linear layers are for classification and the successive linear layer size keeps reducing in size till the size is equal to the number of classes. At the final output layer we use the softmax activation function to get a definitive answer. The standard T4 gpu of Google Colab was used for this project.

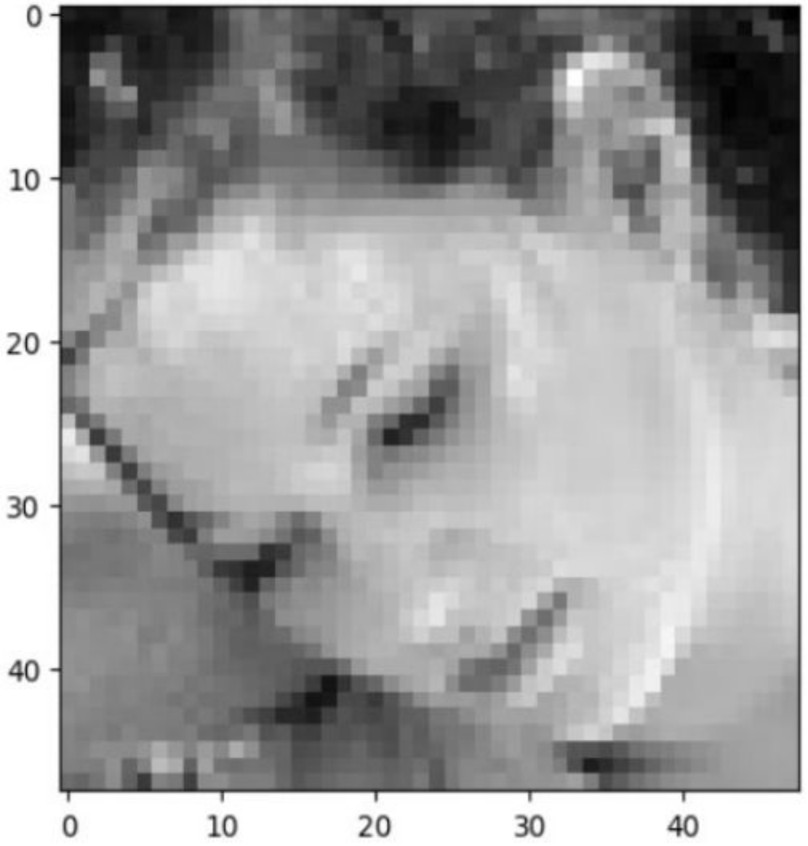
Results:



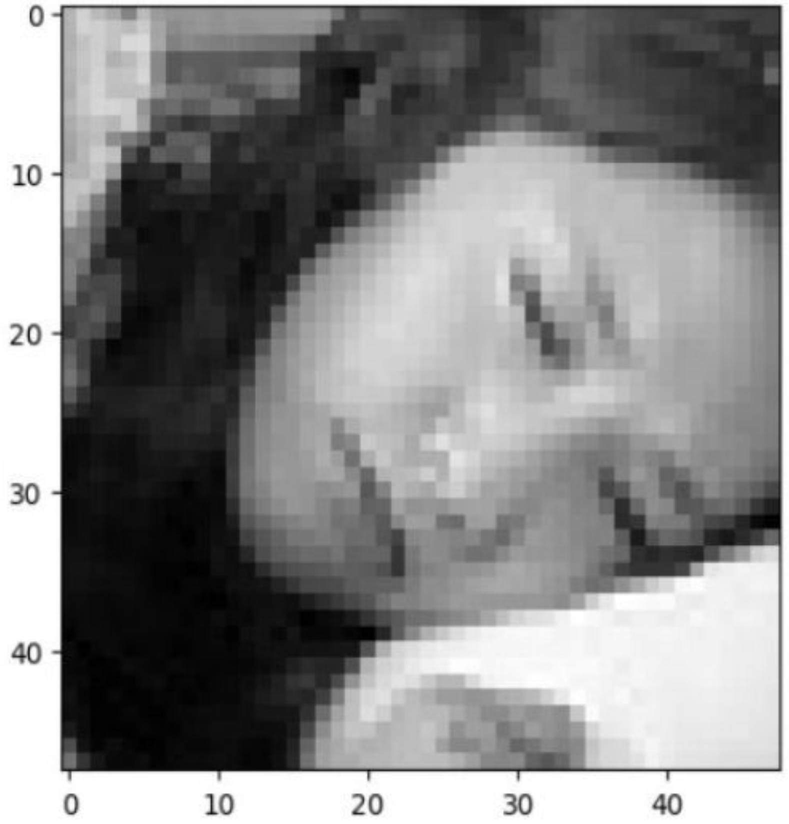
We achieved a test accurate of 68%



angry
disgust
fear
happy
neutral
sad
surprise
True label
Predicted label
634 5 62 32 138 64 23
66 21 8 6 3 6 1
201 0 396 51 170 114 92
66 1 28 1517 117 15 30
92 2 61 81 899 75 23
196 0 155 78 367 430 21
35 1 81 55 23 9 627
0
200
400
600
800
1000
1200
1400



Original Output: sad
Predicted Output: neutral



Original Output: sad
Predicted Output: sad

Comparison:

Sr No	Dataset	Model	Accuracy
1.	JAFFE	DenseNet161	96.51%
2.	FER2013	CNN	71%
3.	FER2013	Improved ResNet	83%
4.	FER2013	Our Model	68%

Proposed Approach Limitations

Expand Dataset Exploration

Our study utilized two datasets, CK+ and FER2013, for emotion detection, but further exploration of additional datasets is necessary for robustness. Expanding dataset size, particularly with larger subsets of CK+ and FER2013, could improve emotion recognition outcomes.

Develop Task-Specific Models

Future research could develop task-specific CNN models tailored for emotion recognition systems.

Incorporate Alternative Evaluation

Incorporating alternative evaluation methodologies like k-fold cross-validation and statistical analysis could enhance the validation of our approach.

Future Directions

1 K-Fold Cross Validation

We divide the model to multiple sets , use on to train the models and the rest are used to test, every time we use different set to train and repeat it k times.

2 Optimize Model Architecture

Ongoing optimization efforts could refine our model architecture, potentially incorporating new blocks for improved efficiency and accuracy.

3 Multimodal Emotion Recognition

Future research could expand emotion recognition to include speech and body gestures, exploring innovative approaches for practical applications.

Conclusion

In conclusion, while showing promising results, our model for emotion recognition also presents avenues for further enhancement and exploration. The limitations identified, such as the need for additional datasets, underscore the importance of ongoing research to ensure the robustness and generalizability of our approach. By addressing these limitations and incorporating them in future directions, such as optimization efforts and exploration of real-world applications, our model holds the potential to contribute significantly to the field of emotion recognition. Through continued collaboration and innovation, we can advance the capabilities of our model and pave the way for its integration into practical settings, ultimately benefiting society at large.

Thankyou

Collaboration

Thank you for your time and effort in working with us on this project. Your valuable input and expertise have been instrumental in shaping the final outcome.

Innovation

We appreciate your willingness to explore new ideas and push the boundaries of what's possible. Your innovative thinking has been a driving force behind the success of this endeavor.

Progress

This project has been a journey of growth and learning for all of us. We are grateful for your commitment to seeing it through and helping us achieve our goals.