PROBLEM STATEMENT

Creating a facial emotion recognition system using neural networks for real-time analysis of human expressions, aimed at enhancing user engagement and experience in digital interfaces.

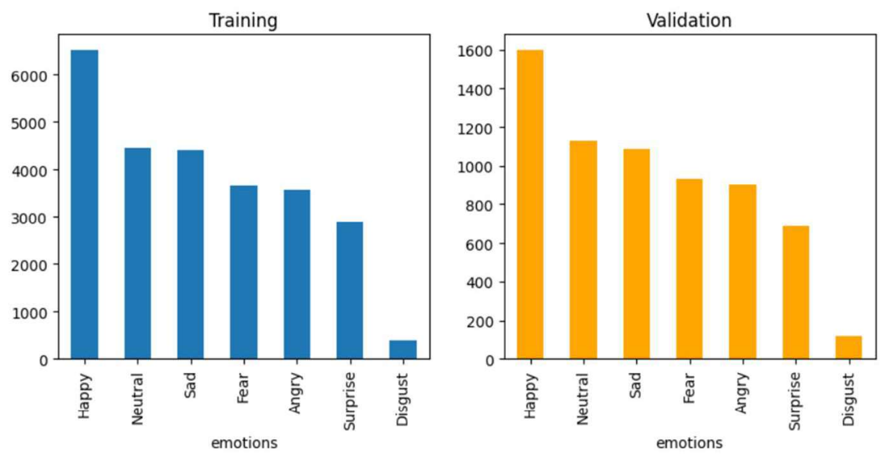
INTRODUCTION:

Facial expressions play a vital role in nonverbal communication among people, and researchers have extensively studied the perception, production, and interpretation of these expressions. As facial expressions are crucial in human interaction, the computer vision-based automatic recognition of Facial Expressions (FER) enables numerous novel applications in fields such as data analytics and human-computer interaction.

It can be used for various purposes like various recommender systems of songs and movies according to the emotion of human, and also for AI enabled therapy solutions.

DATASET:

The FER-2013 dataset has been used. This Dataset contains dataset contains 35887 images, with 4953 “Anger” images, 547 “Disgust” images, 5121 “Fear” images, 8989 “Happiness” images, 6077 “Sadness” images, 4002 “Surprise” images, and 6198 “Neutral” images. These images are 48x48 grayscale.



Problems with Dataset:

1.) Highly Unbalanced Data: The data is highly unbalanced. There is a significant lack of data on the emotion of disgust.

2.) Low Resolution: The images were of very low resolution which created a problem during the implementation of the dataset. The image resolution is [48,48] which was not significant enough that with humans it also had only 65% accuracy(Ian Goodfellow performed some small-scale experiments to estimate the human performance on this task).

Techniques Implemented to Balance the dataset:

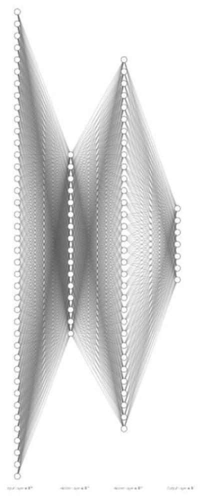
1.) SMOTE (Synthetic Minority Over-sampling Technique): This addresses class imbalance by generating synthetic samples for the minority class. It works by randomly selecting a minority class instance and computing the k-nearest neighbours for that instance. Then, it selects one of these neighbours randomly and generates a synthetic instance at a random point between the original instance and the selected neighbour.

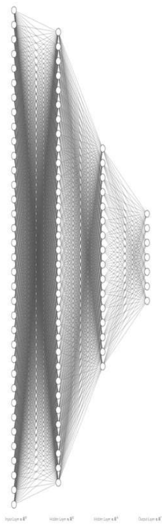
2.) Random Oversampling: This addresses class imbalance by increasing the number of instances in the minority classes by randomly duplicating existing instances until a more balanced distribution is achieved between classes. This approach involves randomly selecting instances from the minority class and replicating them, which helps to mitigate the effects of class imbalance and improve the performance of classifiers by providing more representative training data. However, it may lead to overfitting in some cases, especially when the minority class is small, as it does not introduce new information but merely duplicates existing samples.

3.) Images Augmentation: This helps to balance class distribution in image datasets by generating additional instances for minority classes through transformations like rotations, translations, scaling, flipping, and cropping. By applying these techniques to underrepresented classes, the model gains exposure to them during training, improving performance and generalization without degrading data quality.

MODELS IMPLEMENTED

Fully Connected Neural Network:

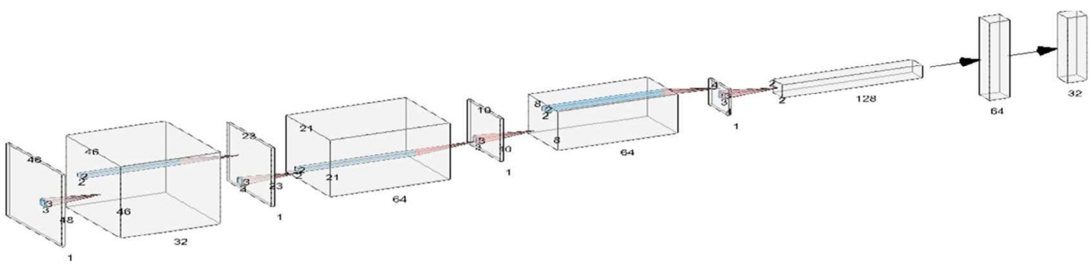




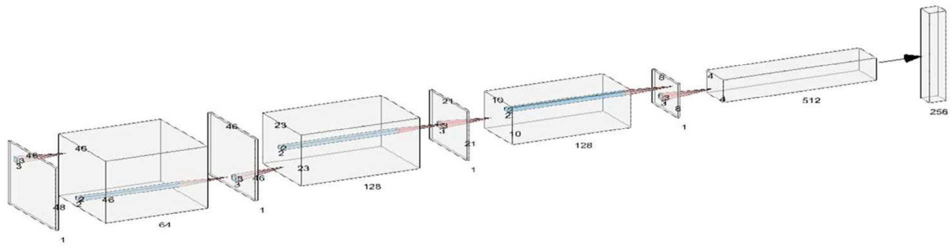
2304-256-512-7
Parameters:735495
ACCURACY:44.97%

2304-512-256-7
PARAMETERS:1324551
ACCURACY:44.08%

Convolutional Neural Network (without balancing):



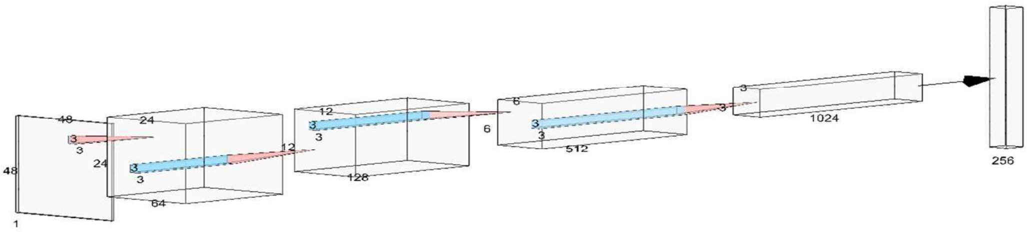
PARAMETERS:132167
ACCURACY:62.16%



PARAMETERS:948875

ACCURACY:65.29%

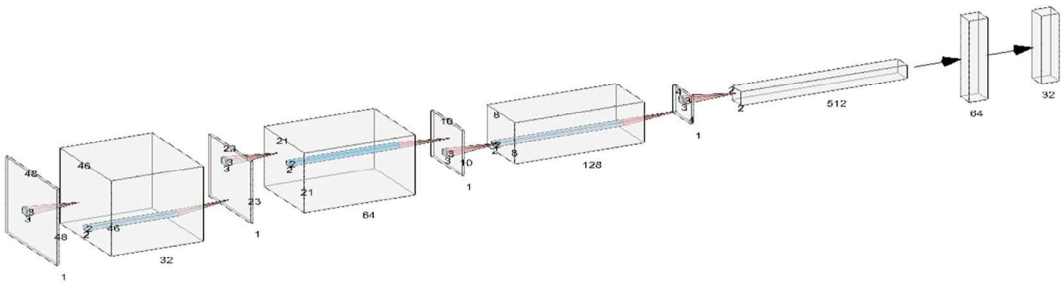
Convolutional Neural Network (SMOTE):



PARAMETERS: 5655563

ACCURACY:64.38%

Convolutional Neural Network (Over-Sampling):

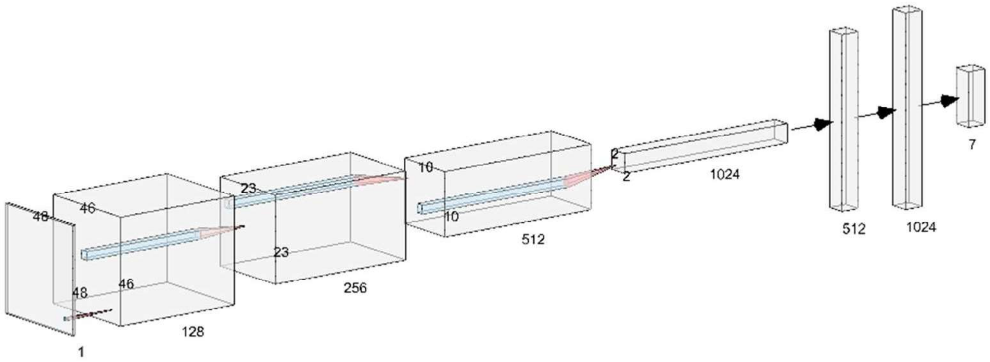


PARAMETERS: 914825

ACCURACY:49.07%

Mayank Yadav
23/CS/250

Convolutional Neural Network (Balanced by Augmentation):



PARAMETERS: 7263243

ACCURACY: 66.97%

I also tried transfer learning with the following models:

1.) InceptionV3 with 256-512-1024 dense layer:

PARAMETERS: 22991405

ACCURACY: 64.98%

2.) VGG16 with 1024 dense layer:

PARAMETERS: 15247181

ACCURACY: 43.05%

All the code can be found here:

https://www.kaggle.com/wolgwang/fer2013-facial-emotion-recognition-by-wolgwang

https://github.com/wolgwang1729/Facial-Expression-recogniser

Mayank Yadav
23/CS/250