**EMOTION DETECTION**

**FROM**

**UPLOADED IMAGES**

**PROJECT**

**DOCUMENT**

# **Index**

* **Objective**
* **Dataset**
* **Data Preprocessing**
* **Model Architecture**
* **Stream lit Application**

**Objective:**

The goal of this project is to build a deep learning model capable of recognizing human emotions from images. The model is trained using a dataset of facial images (FER-2013) to classify facial expressions into one of seven categories: Anger, Disgust, Fear, Happiness, Sadness, Surprise, and Neutral. The primary method used for image classification is a custom Convolutional Neural Network (CNN) with advanced data augmentation, mixup data augmentation, class balancing, and an adaptive learning rate scheduler.

**Dataset:**

The dataset used for this project is the FER-2013 dataset, which is widely used for facial emotion recognition. It contains a total of 35,887 labeled images divided into training, validation, and test sets. The dataset's classes are as follows:

* Anger
* Disgust
* Fear
* Happiness
* Sadness
* Surprise
* Neutral

**Data Preprocessing:**

* **Image resizing**: All images are resized to 48x48 pixels to standardize input dimensions.
* **Normalization**: The images are normalized using pre-trained model statistics (mean and standard deviation).
* **Augmentation**: Data augmentation techniques such as random horizontal flipping, rotation, sharpness adjustment, and brightness/contrast jittering were used to increase the model’s robustness and generalizability.
* **Class Weights**: The class weights were calculated based on the frequency of each class in the training data. This is used in the loss function to handle class imbalance by giving higher weight to underrepresented classes.
* **Data Loaders**: The data is loaded in batches using PyTorch's DataLoader with SubsetRandomSampler to handle the imbalanced dataset.
* **Balanced Sampling**: Both training and test datasets are balanced by oversampling the underrepresented classes to ensure fair evaluation.

**Model Architecture:**

**CNN Architecture**

* The CNN model consists of four convolutional blocks, each followed by batch normalization, ReLU activation, max pooling, and dropout layers. The network has the following structure:
* Block 1: Conv -> Conv -> MaxPool -> Dropout
* Block 2: Conv -> Conv -> MaxPool -> Dropout
* Block 3: Conv -> Conv -> MaxPool -> Dropout
* Block 4: Conv -> Conv -> MaxPool -> Dropout
* After feature extraction, a Global Average Pooling layer is used to reduce the number of parameters. The model ends with two fully connected layers, with the output layer having 7 neurons corresponding to the 7 classes.

**Model Parameters**

* Dropout Rate: 0.4
* Optimizer: AdamW optimizer with a learning rate of 0.0005 and weight decay (L2 regularization) of 0.01
* Loss Function: Cross-Entropy Loss with class weights to handle class imbalance
* Learning Rate Scheduler: Cosine Annealing Scheduler with a warmup phase for better convergence.

**Model Evaluation**

The performance of the model is evaluated using the following metrics:

* Accuracy: The proportion of correct predictions.
* Precision: The proportion of true positive predictions out of all positive predictions.
* Recall: The proportion of true positive predictions out of all actual positives.
* F1-Score: The harmonic mean of precision and recall, providing a balanced measure of the model’s performance.

**Stream lit Application:**

* Developed a stream lit app to detect the emotion from the uploaded image.
* Used media pipe library to extract key facial landmarks of the uploaded image .
* By using the saved model of the CNN Architecture it performs the emotion detection.
* Predicting the emotion by the user uploaded image.

**THANK YOU**