

## Mini Project: Face Emotion Recognition using Deep Learning

### Problem Statement:

This project aims to develop a deep learning model that can recognize emotions from facial expressions in images. Using a convolutional neural network (CNN), the task is to classify facial expressions into different emotional categories such as "Happy," "Sad," "Angry," "Surprised," and others. The goal is to build an emotion recognition system that can accurately predict emotions from facial images.

### Dataset Link:

[The FER-2013 Dataset \(Facial Expression Recognition\)](#)

### Project Requirements:

#### 1. Python Libraries:

- `tensorflow` or `keras` for building the model.
- `opencv` or `PIL` for image preprocessing.
- `matplotlib` for visualizations.
- `numpy` and `pandas` for data manipulation.

#### 2. Model Architecture:

- Use Convolutional Neural Networks (CNN) for feature extraction from images.
- Apply data augmentation techniques to improve model generalization.
- Implement a classifier that will output one of the emotion classes.

## Project Steps:

### 1. Data Understanding and Preprocessing:

#### Objective:

Understand the dataset structure and prepare it for training.

#### Task:

- Load the dataset and explore its structure (e.g., number of classes, image resolution, and labels).
- Perform data preprocessing:
  - Resize the images to a uniform size.
  - Normalize the pixel values to scale the images to a range of [0, 1].
  - Split the dataset into training, validation, and test sets.
  - Perform data augmentation (e.g., rotation, zoom, flip) to improve model robustness.

### 2. Model Construction:

#### Objective:

Build and compile a Convolutional Neural Network (CNN) for emotion recognition.

#### Task:

- Design a CNN model that will process the images and predict one of the emotional categories.
  - Use convolutional layers, max-pooling, and dropout for regularization.
  - Add a fully connected layer before the output layer.
  - Output layer: Softmax activation function for multi-class classification.
- Compile the model using `categorical_crossentropy` loss function and `accuracy` metric.

### 3. Model Training and Evaluation:

**Objective:**

Train the model on the training data and evaluate its performance.

**Task:**

- Train the model using the training dataset and validate it using the validation dataset.
- Use **EarlyStopping** to avoid overfitting.
- Evaluate the model using test data and report accuracy and loss.
- Plot training and validation loss/accuracy curves to monitor training progress.

### 4. Model Optimization:

**Objective:**

Improve the model performance.

**Task:**

- Experiment with different CNN architectures (e.g., adding more layers, changing kernel sizes, adding dropout).
- Tune hyperparameters such as the learning rate, batch size, and number of epochs.
- Implement techniques like data augmentation, dropout, or batch normalization to improve generalization.

### 5. Face Emotion Prediction:

**Objective:**

Make predictions on new images.

**Task:**

- Load the trained model and use it to predict the emotion on unseen images.
- Preprocess the input image (resize, normalize).
- Predict the emotion category using the trained CNN model.

- Visualize the predicted emotion with the image.

## **6. Model Evaluation:**

### **Objective:**

Evaluate the performance of the model on different metrics.

### **Task:**

- Generate a confusion matrix to show how well the model predicts each emotion.
- Calculate additional metrics like precision, recall, and F1-score for each class.
- Report the model's overall accuracy and provide analysis of misclassified emotions.

## **Expected Outcomes:**

### **1. Data Preprocessing:**

- Students will learn how to handle image datasets, resize, normalize, and augment data for deep learning models.

### **2. Deep Learning Model Development:**

- Students will build a CNN from scratch and train it to classify facial emotions accurately.

### **3. Model Evaluation:**

- Evaluate the model's performance and provide actionable insights on improving the system based on metrics like accuracy, precision, recall, and F1-score.

### **4. Prediction:**

- Students will understand how to deploy a trained model for real-time emotion prediction on new facial images.