This code trains a convolutional neural network (CNN) to recognize emotions from grayscale images using TensorFlow and Keras.

Here is an overview:

**Data Preparation**

Directories:

train\_data\_dir and validation\_data\_dir specify paths to training and validation datasets.

Image Counting:

Functions to count and print the number of images in each class for both training and validation datasets.

Data Augmentation:

ImageDataGenerator for training data augmentation (rescaling, rotation, shifting, shearing, zooming, flipping).

Validation data is only rescaled.

Data Generators:

train\_generator and validation\_generator load and preprocess images from directories.

**Model Architecture**

Sequential Model:

Uses convolutional (Conv2D), max pooling (MaxPooling2D), batch normalization (BatchNormalization), and dropout (Dropout) layers.

Ends with dense layers and a softmax activation for classification into seven emotion categories.

Compilation:

Compiled with Adam optimizer (learning rate 0.0001), categorical crossentropy loss, and accuracy metric.

Training

Model Training:

Trains for up to 100 epochs with training and validation data generators.

Model Saving:

Saves the trained model to 'model\_fite.h5'.

**Evaluation and Visualization**

Visualization:

Plots training and validation accuracy and loss over epochs to visualize performance.

**Summary**

Data Augmentation: Augment training data through transformations.

Model Architecture: Builds a CNN for emotion recognition.

Image Counting: Counts and displays the number of images per class.

Visualization: Plots accuracy and loss for performance insight.