# **Software development for Facial-Expression-Detection**

# Concept Note

# Introduction:

# 

Facial Expression or Facial Emotion Detector can be used to know whether a person is sad, happy, angry and so on only through his/her face. Due to globalization and digital divide facial expression detection has received the primary attention in order to identify the criminals and breaches. Facial expressions are the changes occurring on the human face indicating a person's internal emotional states, intents or societal communications. Depending on the expressions on the face, human face is the most principal mode of conveying and deducing affective states of human ones. In real time, facial expression detection has become a prominent research area as it plays an important role in Human Computer Interaction.

# Implementations:

# The applications of the facial expression detection can be used in

# Computer vision,

# Biometric security,

# social interaction,

# Emotional intelligence and social intelligence.

# Identify the criminals and breaches.

# Requirements:

This Project can be used to carry out such a task. It uses your Web Camera and then identifies your expression in Real Time. Yeah in real-time! Face recognition is a computer vision task of identifying and verifying a person based on a photograph of their face.

# Face Recognition using VGG

Recently, deep learning convolutional neural networks have surpassed classical methods and are achieving state-of-the-art results on standard face recognition datasets. One example of a state-of-the-art model is the VGGFace and VGGFace2 model developed by researchers at the Visual Geometry Group at Oxford.

Although the model can be challenging to implement and resource intensive to train, it can be easily used in standard deep learning libraries such as Keras through the use of freely available pre-trained models and third-party open source libraries.

## **Data-set**

We have downloaded the facial expression recognition (FER) data-set from [Kaggle challenge](https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data). The data consists of 48×48-pixel gray scale images of faces. The faces have been automatically registered so that the face is more or less centered and occupies about the same amount of space in each image. The task is to categorize each face based on the emotion shown in the facial expression in to one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral).

The training set consists of 35,888 examples. train.csv contains two columns, “emotion” and “pixels”. The “emotion” column contains a numeric code ranging from 0 to 6, inclusive, for the emotion that is present in the image. The “pixels” column contains a string surrounded in quotes for each image. The contents of this string a space-separated pixel values in row major order

# Libraries used:



**Improvements:**

While creating training datasets for models, we may come across certain challenges like

1) Imbalance problem

2) Intra-class variation

3) Occlusion

4) Contrast variation.

While imbalance problem may arise if the training dataset contains an unequal number of samples for each group of emotions leading to a bias towards a certain emotion, other problems like intra-class variation (presence of different forms of images in dataset) and occlusion (covering of the facial features like with hands) can lead to unacceptable results. Solution to the problems result from data augmentation to avoiding overfitting. Data augmentation encompasses usage of GAN(Generative Adversarial Network) models to create fictitious facial images and also includes scaling, translating, rotating exiting images to enhance the dataset.

**Model Architecture:**

**Softmax**

**Global Average Pooling 2D**

**Convolution 2D**

**Convolution 2D/Batch Normalisation**

**Convolution 2D/Batch Normalisation**

**Image**

**Repeated Four Times**

**Separable-Covolution 2D/Batch Normalisation**

**Separable-Covolution 2D/Batch Normalisation**

**Convolution 2D/Batch Normalisation**

**Maxpool 2D**

**Conclusion:**

Finally, this project will helpful for the detection of facial expressions using deep learning convolution neural network and Keras library. Deep learning helps us to create such amazing software by which we can easily detect facial expressions through a webcam in real-time. Here we have used 35800+ data sets to execute our program, if we increase the amount of data in the data set then the output will be more accurate. In robot science, this technology helps the bot to easily identify human emotions in real-time and give a quick reply to them.