EMOTION RECOGNITION USING CNN

By

A Christopher Daniel
R Logesh

1. Introduction

1.1 Purpose:

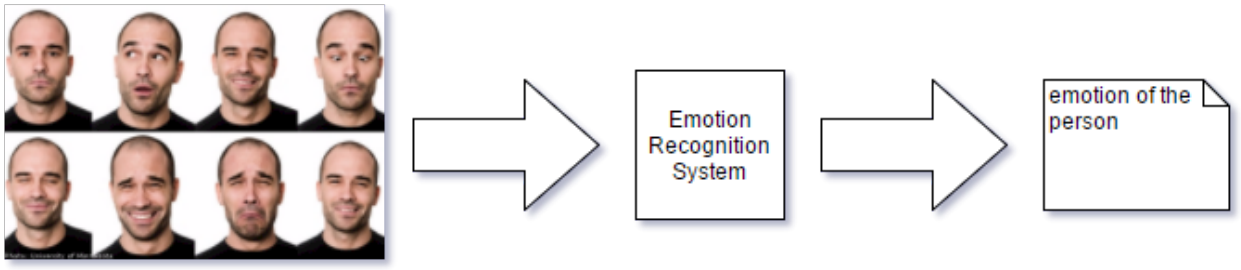
The purpose of the project is to create an emotion recognition system through facial expressions. The emotion recognition system is implemented using convolutional neural networks. This system can be extended to customer satisfaction measurement system, child care, old age care system and treating persons with hearing and speaking disorders.

1.2 Product Scope:

The software developed collects images of facial expressions from the dataset and learns from it. It takes input real-time and recognises the emotion as *Angry, Disgusted, Fearful, Happy, Sad, Surprised, and Neutral.* This software aims to predict the emotion with maximum accuracy possible.

1.3 Definitions, Acronyms & Abbreviations:

CNN	- Convolutional Neural Network (ConvNet)
GPU	- Graphics Processing Unit
CUDA	-Compute Unified Device Architecture
CuDNN	-CUDA Deep Neural Network
API	-Application Program Interface
DFD	-Data Flow Diagram
FER	-Facial Expression Recognition



2. Overall Description

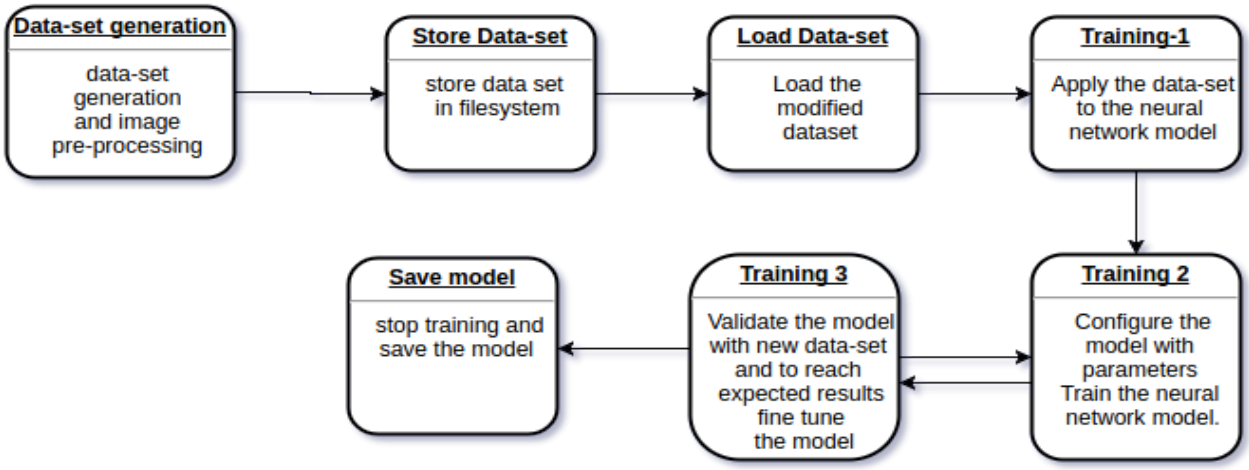
2.1 Product Perspective:

This software is a replacement of existing emotion recognition systems which uses conventional computer vision feature extractor and classifier algorithms. This software is planned to run on an embedded system which has camera, GPU, SSD storage for faster computing.

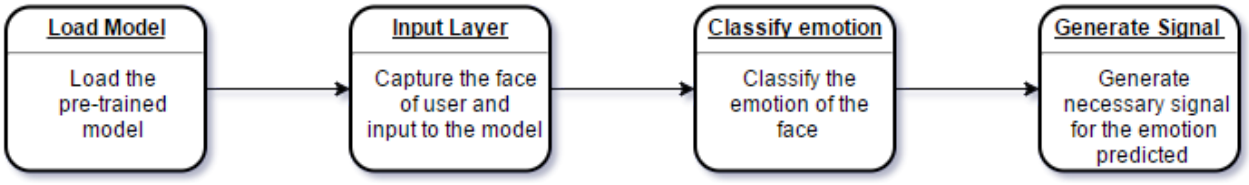
2.2 Product Functions:

- The dataset must be processed and stored in the filesystem.
- The processed dataset must be loaded to the neural network
- The neural network must be configured with parameters and loaded with data
- Continue training the neural network until it reaches the desired accuracy with fine tuning mechanism.
- The model should be validated with a test dataset.
- The model should be saved after training

- Load the pre-trained model
- Capture the user face and supply to the model
- It should classify the emotion using the face image.
- It should generate the necessary signal to the intended actor for the expression posed by the user.



1. DFD Diagram



2. DFD Diagram

2.3 Operating Environment:

The Software will run on Linux machines with python, keras, CuDNN and CUDA installed. Camera support is must. NVIDIA GPU is preferable. In order to do embedded applications Jetson TX1 Developer kit is used.

2.4 Design and implementation constraints:

- The camera must focus on the person face.
- The person should be expressive.
- The light illumination should be appropriate.

2.5 Dependencies:

The software dependencies includes,

1. Keras
2. Theano
3. OpenCV
4. CUDA
5. CuDNN
6. Pandas

2.6 Data- set:

For training the neural network, FER2013 is a dataset is taken from Kaggle.com website.
Dataset is part of the kaggle.com challenges in representation learning facial expression recognitioncontest conducted in the year 2013.
The training dataset contains 28,709 images and test dataset contains 3589 images of 48X48 images.

3. External Interface Requirements:

3.1 User Interfaces:

The GUI displays the frame captured by the camera. If a face is detected, the classified emotion will be shown as a histogram.

3.2 Hardware Interfaces:

Camera

- A camera module is present and interfaced with OpenCV library.
- Camera output is read as numpy arrays from video stream**.**

3.3 Software Interfaces:

Theano
Deep learning library written in Python.

Keras
High level API interface for Theano.

Numpy
Python library for numerical computations especially for multidimensional arrays.

OpenCV
Computer vision library.

Pandas
library for processing the dataset.

Scipy
Library for Scientific Calculations.

Scikit-Learn
Library for Machine learning.

4. System features:

4.1 Emotion classification:

Description and Priority:
This feature classifies the captured face expression as angry, sad, happy, disgusted, fearful, surprised and neutral.
It has higher priority level 7.

Stimulus / Response:
The face image detected stimulates the module to work a
returns the possibility for every emotion.

Functional requirements:
Requirement 1 - The emotion classified should be sent periodically to concerned actor.

4.2 Model Training:

Description and Priority:
The neural network model can be trained using any facial expression recognition dataset**.**
It has high priority 9 because system performance and accuracy depends on training of the neural network.

Stimulus / Response:
The model gets the dataset, architecture and hyper-parameters, allocates the resources and starts training.
It returns the final trained model which can be used for production.

Functional Requirements:
Requirement 1 - Pre-processed dataset
Requirement 2 - well defined parameters and hyper parameters
Requirement 3 – Optimizer and loss calculation algorithms

5. Non-Functional Requirements

5.1 Performance Requirements:

The system must respond to every frame captured in less than 1 second.
The camera should capture frame with good clarity for face extraction from it.
The light illumination should be optimal to capture images efficiently for analysis.

5.2 Security Requirements:

The classified emotion detail should be sent only to the concerned person.

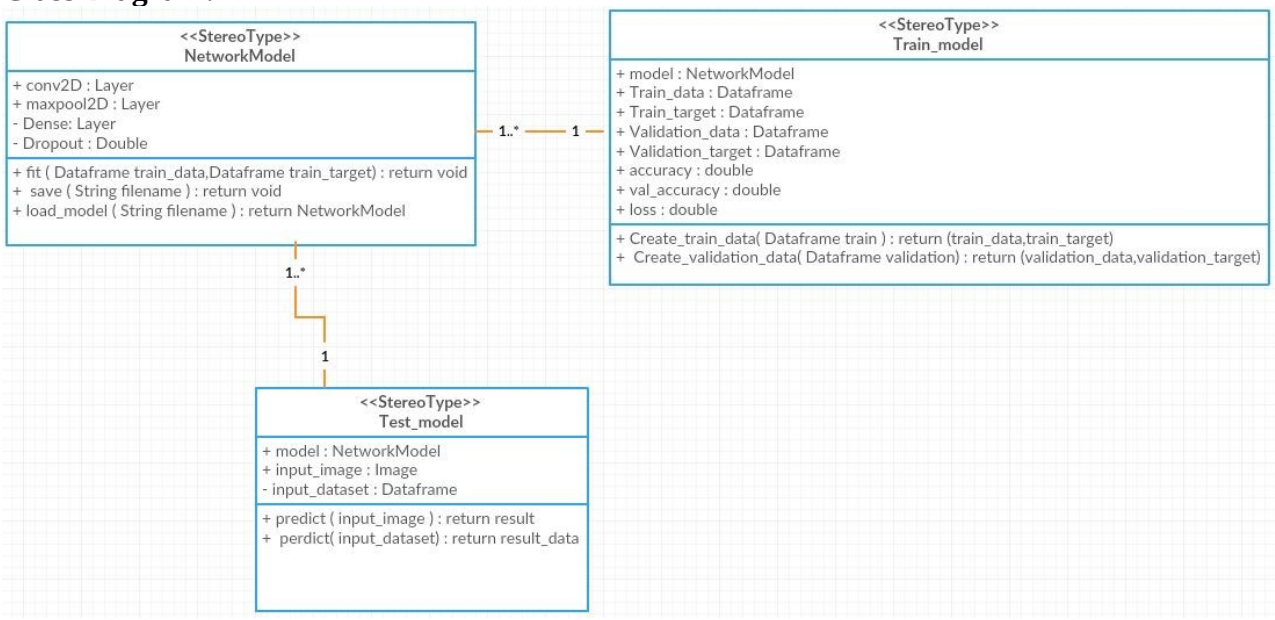
5.3 Software Quality Attributes:

The dataset should be large enough for CNN to train optimally.
The CNN hyper-parameters needs to be tuned until expected results are obtained.

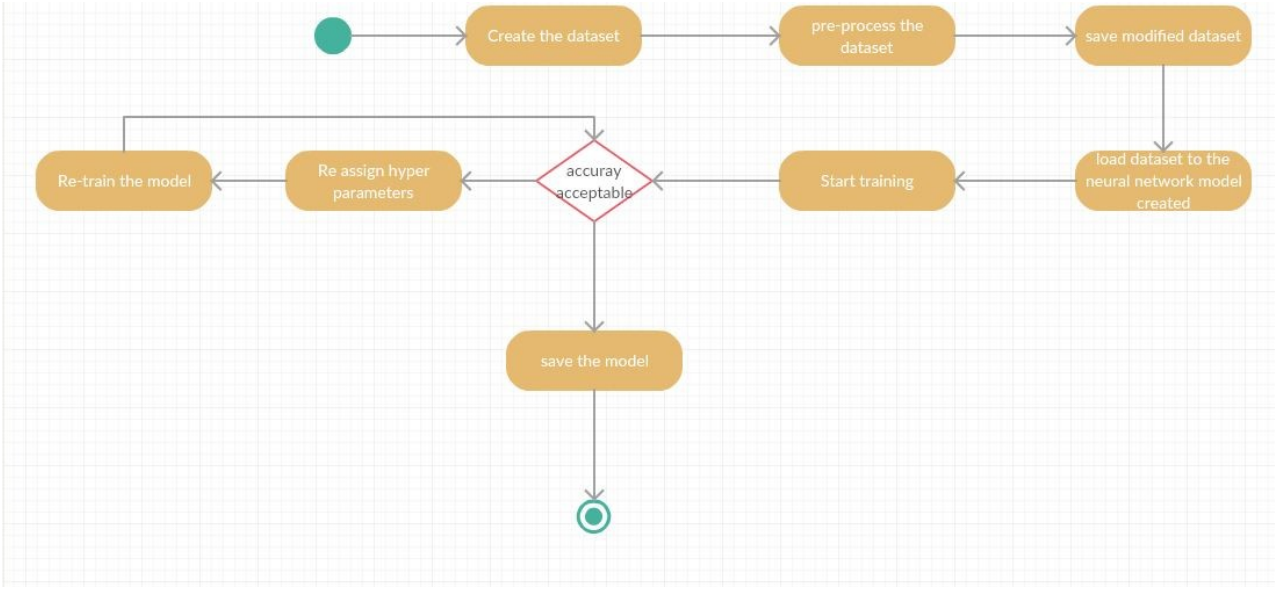
6. Other Requirements

6B Design Appendix:

Class Diagram:



Activity Diagram for training:



Activity Diagram for testing: