Real-Time Facial Expression Detection Using Convolutional Neural Network

A PROJECT REPORT

Submitted by

ANJU A (TKM19MCA004)

to

The APJ Abdul Kalam Technological University

In partial fulfillment of the requirments for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

Thangal Kunju Musaliar College of Engineering
Kerala

DEPARTMENT OF COMPUTER APPLICATIONS

MAY 2022

DECLARATION

I undersigned hereby declare that the project report on‘’Real-Time Facial Expression Detection Using Convolutional Neural Network” , submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Dr.Nadera Beevi S. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that we have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University..

Kollam

22/05/2022

Anju A

DEPARTMENT OF COMPUTER APPLICATIONS

TKM COLLEGE OF ENGINEERING

C E R T I F I C A T E

This is to certify that, the project report entitled **"Real-Time Facial Expression Detection Using Convolutional Neural Network"** is submitted by **Anju A (TKM19MCA004)** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Master of Computer Applications, is a bonafide record of the project work carried out by her under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Internal Supervisor | Head of the Department | External Examiner

ACKNOWLEDGEMENT

First and foremost I thank GOD almighty and my parents for the success of this project. I owe sincere gratitude and heart full thanks to everyone who shared their precious time and knowledge for the successful completion of my project.

I am extremely grateful to **Prof. Dr. Fousia.M.Shamsudeen**, Head of the Department, for providing me with best facilities.

I would like to thank my project guide **Prof.Dr.Nadera Beevi S** , Department of Computer Applications, who motivated me throughout the work of my project.

I profusely thank all other faculty members in the department and all other members of TKM College of Engineering, for their guidance and inspirations throughout my course of study.

I owe my thanks to my friends and all others who have directly or indirectly helped me in the successful completion of this project.

Anju A

ABSTRACT

This project involves identification of facial expressions that reveal human emotions can help computers to better assess the human state of mind, so as to provide a more customized interaction. We explore the recognition of human facial expressions through a deep learning approach using a Convolutional Neural Network (CNN) algorithm. The system uses a labelled data set containing around 32,298 images with multiple facial expressions for training and testing. The pretraining phase involves a face detection subsystem with noise removal, including feature extraction. The generated classification model used for prediction can identify seven emotions of the Facial Action Coding System (FACS). The Facial Action Coding System (FACS) refers to a set of facial muscle movements that correspond to a displayed emotion. Using FACS, we are able to determine the displayed emotion of a participant. This analysis of facial expressions is one of very few techniques available for assessing emotions in real-time.

Contents

List of Figures

Chapter 1

Introduction

The rapid growth of artificial intelligence has contributed a lot to the technology world. As the traditional algorithms failed to meet the human needs in real time, Machine learning and deep learning algorithms have gained great success in different applications such as classification systems, recommendation systems, pattern recognition etc. Emotion plays a vital role in determining the thoughts, behaviour and feeling of a human. An emotion recognition system can be built by utilizing the benefits of deep learning and different applications such as feedback analysis,etc. can be implemented with good accuracy. The main focus of this work is to create a Deep Convolutional Neural Network (DCNN) model that classifies 7 different human facial emotions. The model is trained, tested and validated using the manually collected image dataset. In the field of computer science, machine learning is one of the emerging technologies that considered having an impact of 90 percentage in the next 4 years. Deep learning, a subset of machine learning uses artificial neural network, which is an algorithm inspired from the human brain. Convolutional Neural Network (CNN) is a class of deep neural network that uses convolution as the mathematical operation. As the dataset consists of images, the system uses a 2D CNN for the recognition task. The proposed deep convolutional neural network is trained not only to classify 7 different human facial emotions, but also to yield a good accuracy.

1.1 Existing System

In the existing system, image processing techniques is used to detect emotions, which is not accurate. Some of the existing work includes machine learning algorithms, in which facial features is extracted from the input image by using image preprocessing techniques. Thus, it leads to lower the accuracy. Also, it classifies emotions to a limited number of emotions i.e., 2 or 3 emotions. Many of the existing methods used still images and emotion were perceived by measuring the dimensions of lips and eyes. Currently, the system failed to identify whether the person is stressed or not based on the predicted emotion as well as the amount of stress in the input image, i.e., stress ratio or happiness ratio. Existing work make predictions on the uploaded input facial image, which is not real- time stress detection.

1.2 Proposed System

In this proposed system, Deep learning is used for emotion recognition. The image dataset contains number of images with different emotions are used for training the CNN network. Trained model predicts the emotions in one of 7 categories. CNN network consists of different layers. These layers together extract the features from the input image. The first step is to identify the facial region of the input image. Then, it is given to the CNN network for training.

1.3 Objectives

The main objectives of the project are as follows:

- The system aims to develop a real-time implementation of emotion detection.
- To predict emotions from the frame extracted from the webcam feed.

Chapter 2

Literature Survey

Literature review is the comprehensive study and interpretation of literature that relates to a particular topic. When one uses literature review research questions are identified, then one seek to answer this research questions by searching for and analyzing relevant literature. Some importance of literature reviews is that new insights can be developed by the re-analyzing the results of the study.

2.1 Real-Time Facial Expression Detection

Here, we take some of the papers related to Real-Time Facial Expression Detection USING Convolutional Neural Network,

1. Deep Convolutional Neural Networks For Facial Emotion Recognition[M Geetha Yadav , Rajasekhar Nennuri , Roshna Sanjana K , Sai Ramana Deekonda , Shivangi Solanki]
For facial emotion recognition, this paper proposes a two-layer convolution network model. The model uses the image dataset to classify five different facial emotions. The model employs two convolution layers, with dropouts between them. The first convolution layer receives the input image, which is resized to 32 x 32 pixels. The model has comparable training and validation accuracy, indicating that it has the best fit and is generalizable to the data. The activation function used here is ReLU (Rectified Linear Unit), which reduces negative values to zero while maintaining positive values. This feature map is applied to a pooling layer with a pool size of 2 x 2 in order to

reduce the size without sacrificing any detail. The model reduces the loss function using an Adam optimizer, and it has been checked to have an accuracy of 78.04 percent.

2.Emotion Recognition using Deep Neural Network with Vectorized Facial Feature[G. Yang, J. S Saumell, J Sannie]

In this paper, proposed a DNN model which uses vectorized facial features as input. Vectorized facial feature for facial expression will be introduced. The vectorized facial feature can be used to build a DNN (Deep Neural Network) for emotion recognition. The proposed facial feature model can not only reflect facial expressions correctly, it can also be used for DNN with high efficiency. To test the efficiency of such method, a DNN is trained to recognize some universal expressions. Compared with other computer vision powered system, vectorized facial features can achieve similar accuracy as other machine learning algorithms (CNN). Yet, it reduces the data as well as the time required for training. Such advantages can significantly increase the speed of building applications involving emotion recognition. The model can predict different emotions with an accuracy of 84.33 percent.

3.Sign Language Recognition System Using Deep Neural Network[S. Suresh, H. T. P Mithun and M. H. Supriya]

This paper proposes a basic 2 layer convolutional neural network (CNN) to classify sign language image datasets. The classifier was found to perform with varying lighting and noisy image datasets. This model has classified 6 different sign languages using two different optimizers, SGD and Adam with an accuracy of 99.12 percent and 99.51 percent respectively. More accuracy is obtained when using the Adam optimizer. Future work of this Sign Language Recognition System can be extended to improve the performance by tuning the hyperparameters and implement a sign language recognition system from video sequence using CNN LSTM. This sign language recognition system can also be made to control certain devices such as home robot. The two CNN models developed have different type of optimizers, the Stochastic Gradient Descent (SGD) and Adam.

4.Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine[Kun Han, Dong Yu, Ivan Tashev]

In this paper we propose to utilize deep neural networks (DNNs) to extract high level features from raw data and show that they are effective for speech emotion recognition. We first produce an emotion state probability distribution for each speech segment using DNNs. We then construct

utterance-level features from segment-level probability distributions. These utterancelevel features are then fed into an extreme learning machine (ELM), a special simple and efficient single-hidden-layer neural network, to identify utterance-level emotions. The experimental results demonstrate that the proposed approach effectively learns emotional information from low-level features and leads to 20 percent relative accuracy improvement compared to the state-of-the-art approaches

5.Video-Based Emotion Recognition using CNN-RNN andC3D Hybrid Networks[Yin Fan, Xiangju Lu, Dian Li, Yuanliu Liu]

In this paper, present a video-based emotion recognition system. The core module of this system is a hybrid network that combines recurrent neural network (RNN) and 3D convolutional networks (C3D) in a late-fusion fashion.Also added an audio classifier into the system. RNN and C3D encode appearance and motion information in different ways. Specifically, RNN takes appearance features extracted by convolutional neural network (CNN) over individual video frames as input and encodes motion later, while C3D models appearance and motion of video simultaneously. Combined with an audio module, system achieved a recognition accuracy of 59.02 percentage without using any additional emotion-labeled video clips in training set. Used CNN-LSTM and C3D networks to simultaneously model video appearances and motions. Especially, found that the combination of the two kinds of networks can give impressive results, which demonstrated the effectiveness of the method.

Chapter 3

Methodology

3.1 System Architecture

The system architectural design is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components. The system architecture is shown in Figure 3.1.

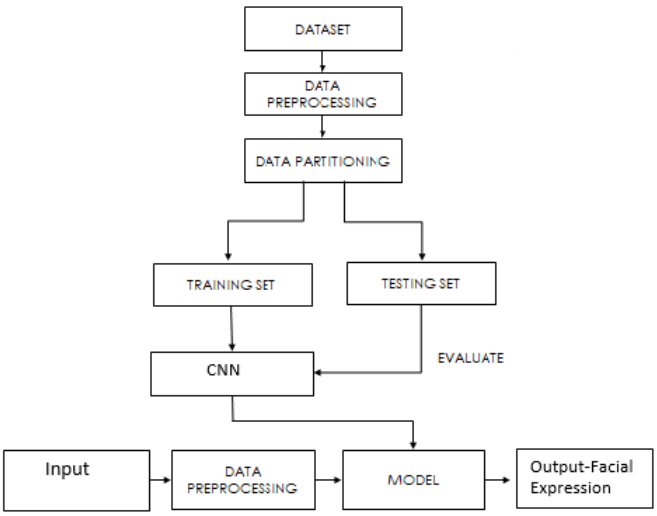


Figure 3.1: System Architecture

3.2 Data set

- The dataset contains face images of different persons with different facial expressions. It includes 7 categories of facial expressions such as, angry, disgusted, fearful, happy, neutral, sad, and surprised.
- Dataset contains total of 35,887 images with .png extension in which images are classified in 7 categories. Angry- 4953 images, disgusted- 547 images, fearful- 5121 images, happy- 8989 images, neutral- 6198 images, sad- 6077 images, surprised- 4002 images.

3.3 Data Preprocessing

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. The aim of pre-processing is an improvement of the data for further processing. Pre-processing of data is carried out before model is built and training process is executed. Since the data are images, thus image preprocessing steps are used. Image rescaling is done to convert all images to a common scale. Image scaling refers to the resizing of a digital image. RGB to Grayscale conversion is also done. Haarcascade technique is used to detect faces in each image or each frame in the video. It is an Object Detection Algorithm used to identify faces in an image or a real time video. The algorithm uses edge or line detection features proposed by Viola and Jones.

3.4 Data Partitioning

The overall dataset provided by FER-2013 contains 35,887 images with .png extension that fall under the 7 classes. The dataset was originally split up into training and testing data. 80 percentage of the images in the dataset for training set and remaining 20 percentage images for testing set. The main difference between the two is that the training data contained the ground truth labels of each image, while the testing images did not. This allows us to evaluate the model and also test the model against unseen data and apply our own testing measures.

3.5 Feature Extraction

Feature extraction is a type of dimensionality reduction where a large number of pixels of the image are efficiently represented in such a way that interesting parts of the image are captured effectively. In deep neural networks (CNN), feature extraction is not manual. The network itself learns to extract features while training. Inside the CNN network, convolution layers extracts features from the input image.

3.6 Model development

A CNN architecture is formed by a stack of distinct layers that transform the input volume into an output volume (e.g. holding the class scores) through a differentiable function. It includes the following layers: .

3.6.1 Convolutional layer (CNV)

The convolutional layer is the core building block of a CNN. The layer's parameters consist of a set of learnable filters (or kernels), which have a small receptive field, but extend through the full depth of the input volume. During the forward pass, each filter is convolved across the width and height of the input volume, computing the dot 16 product between the entries of the filter and the input and producing a 2- dimensional activation map of that filter. As a result, the network learns filters that activate when it detects some specific type of feature at some spatial position in the input. Stacking the activation maps for all filters along the depth dimension forms the full output volume of the convolution layer. Every entry in the output volume can thus also be interpreted as an output of a neuron that looks at a small region in the input and shares parameters with neurons in the same activation map. The convolutional layer is shown in figure 3.2.

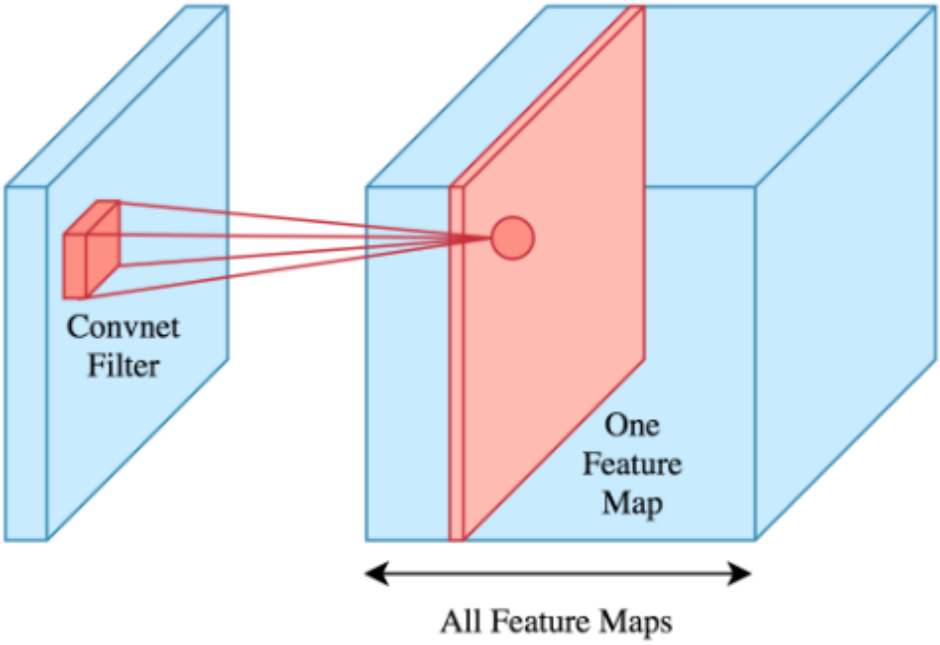


Figure 3.2: Convolution Layer

3.6.2 Pooling layer (PL)

Another important concept of CNNs is pooling, which is a form of nonlinear down-sampling. There are several non-linear functions to implement pooling among which max pooling is the most common. It partitions the input image into a set of nonoverlapping rectangles and, for each such sub-region, outputs the maximum. The intuition is that the exact location of a feature is less important than its rough location relative to other features. The pooling layer serves to progressively reduce the spatial size of the representation, to reduce the number of parameters and amount of computation in the network, and hence to also control overfitting. It is common to periodically insert a pooling layer between successive convolutional layers in a CNN architecture. The pooling operation provides another form of translation invariance. The pooling layer operates independently on every depth slice of the input and resizes it spatially. The most common form is a pooling layer with filters of size 2x2 applied with a stride of 2 down samples at every depth slice in the input by 2 along both width and height, discarding 75 percentage of the activations. In this case, every max operation is over 4 numbers. The depth dimension remains unchanged. This is shown in Fig.3.3. In addition to max pooling, the pooling units can use other functions, such as

average pooling or L2-norm pooling. The pooling layer is shown in figure 3.3.

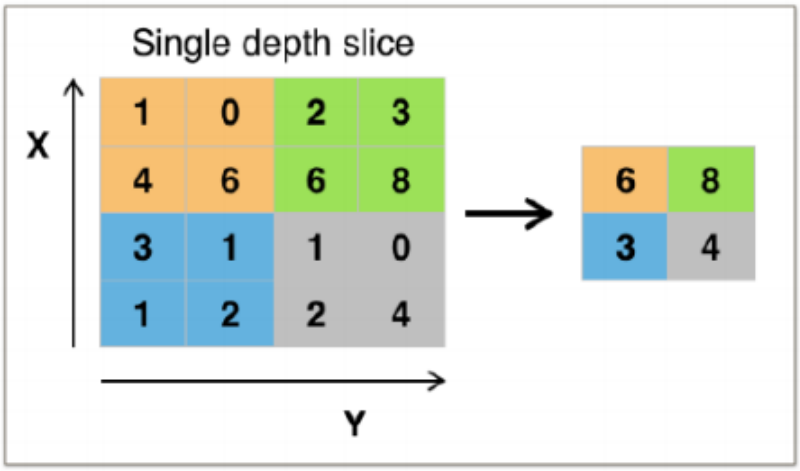


Figure 3.3: Pooling Layer

3.6.3 Fully connected layer (FC)

Finally, after several convolutional and max pooling layers, the high-level reasoning in the neural network is done via fully connected layers. Neurons in a fully connected layer have connections to all activations in the previous layer, as seen in regular neural networks. Their activations can hence be computed with a matrix multiplication followed by a bias offset. The fully connected layer is shown in figure 3.4.

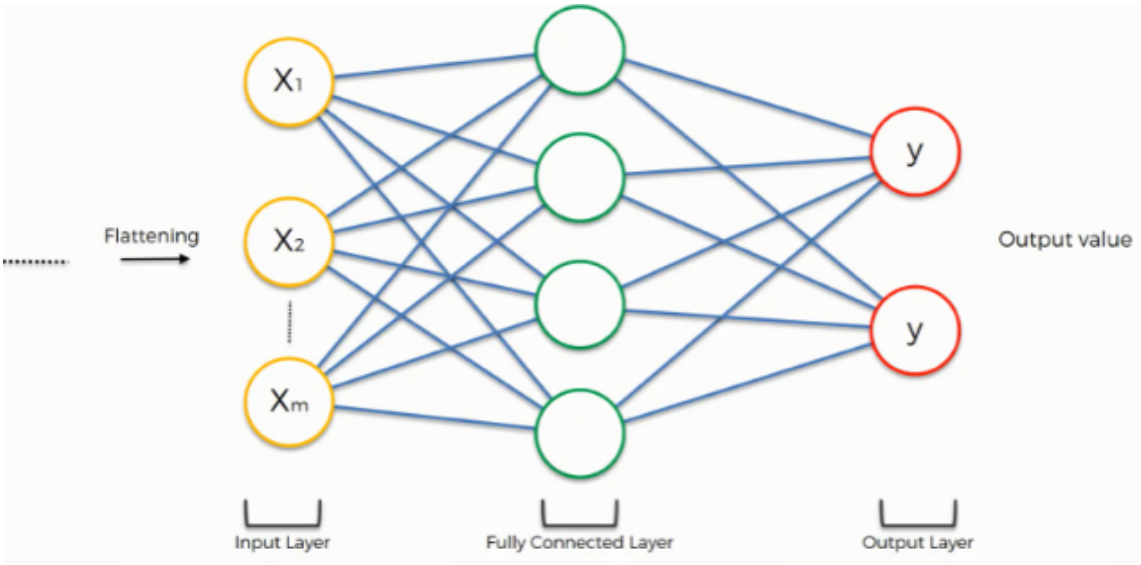


Figure 3.4: Pooling Layer

3.6.4 Classification Layer (CL)

The classification layer specifies how training penalizes the deviation between the predicted and true labels and is normally the final layer. Various loss functions appropriate for different tasks may be used there. SoftMax loss is used for predicting a single class of K mutually exclusive classes. Sigmoid cross-entropy loss is used for predicting K independent probability values in [0,1] A typical CNN architecture is shown below. A simple CNN (Convolutional Neural Net) is created as an initial step. This is then trained with training data.The CNN architecture is shown in figure 3.5.

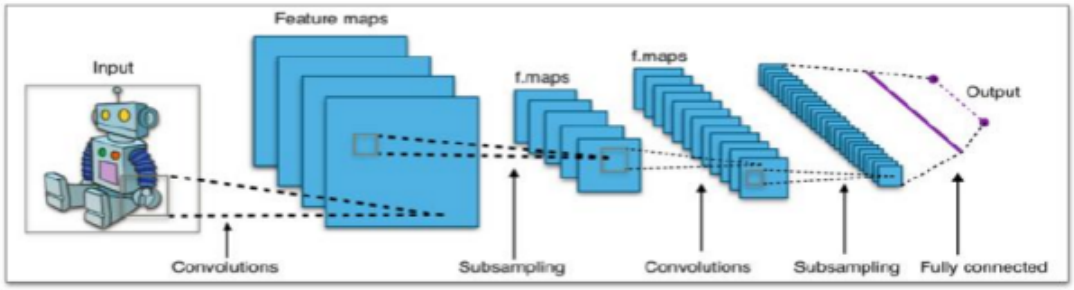


Figure 3.5: CNN Architecture

3.7 Software requirements and specification

• Program logic : Python 3

• IDE : VS Code

• Browser : Chrome/ Edge

• Designing tool : HTML, CSS, Bootstrap, Javascript

• Web framework : Django

• Database : MySQL

• Packages : Tensorflow, Keras

3.7.1 VS CODE – IDE

Visual Studio Code combines the simplicity of a source code editor with powerful developer tooling, like IntelliSense code completion and debugging. First and foremost, it is an editor that gets out of your way. The delightfully frictionless edit-build-debug cycle means less time fiddling with your environment, and more time executing on your ideas. At its heart, Visual Studio Code features a lightning-fast source code editor, perfect for day-to-day use, With support for hundreds of languages, VS Code helps to be instantly productive with syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more. Visual Studio Code includes an interactive debugger, so step through source code, inspect variables, view call stacks, and execute commands in the console. VS Code also integrates with build and scripting tools to perform common tasks making everyday workflows faster.

3.7.2 Python

Python is a high-level, interpreted scripting language developed in the late 1980s by Guido van Rossum at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python is a general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming.

3.7.3 Python Libraries

The Python Standard Library is a collection of exact syntax, token, and semantics of Python. It comes bundled with core Python distribution. We mentioned this when we began with an introduction. It is written in C, and handles functionality like I/O and other core modules. All this functionality together makes Python the language it is. More than 200 core modules sit at the heart of the standard library. This library ships with Python. But in addition to this library, you can also access a growing collection of several thousand components from the Python Package Index (PyPI).

Pandas

Pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with "relational" or "labeled" data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real-world data analysis in Python. Like we've said before, Pandas is a must for data-science. It provides fast, expressive, and flexible data structures to easily (and intuitively) work with structured (tabular, multidimensional, potentially heterogeneous) and time-series data.

NumPy

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It has advanced math functions and a rudimentary scientific computing package

SciPy

SciPy, a scientific library for Python is an open source, BSD-licensed library for mathematics, science and engineering. The SciPy library depends on NumPy, which provides convenient and fast N-dimensional array manipulation. The main reason for building the SciPy library is that, it should work with NumPy arrays. One of the libraries we have been talking so much about. It has a number of user-friendly and efficient numerical routines. These include routines for optimization and numerical integration

Scikit

Scikit-learn is a free machine learning library for Python. It features various algorithms like support vector machine, random forests, and k-neighbours, and it also supports Python numerical and scientific libraries like NumPy and SciPy.

3.7.4 MySQL

MySQL is the world's most popular open source database software. A relational database organizes data into one or more data tables in which data may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups. The best and the most-used database in the world for online applications.

• Available and affordable for all

• Continuously improved while remaining fast, secure and reliable

• Fun to use and improve

• Free from bugs.

3.7.5 HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

3.7.6 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate . css file which reduces complexity and repetition in the structural content as well as enabling the . css file to be cached to improve the page load speed between the pages that share the file and its formatting. CSS also has rules for alternate formatting if the content is accessed on a mobile device. The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

3.7.7 Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components. Bootstrap is among the most starred projects on GitHub, with more than 142,000 stars, behind freeCodeCamp (almost 312,000 stars) and marginally behind Vue.js framework.

3.7.8 JavaScript

JavaScript is a scripting language. A scripting language is easy and fast to learn. A scripting language is interpreted in run-time. It is not compiled like other languages as C++, C, VB.net etc. JavaScript is a client side language and it runs on a client browser. However JavaScript can also be used on the server side. JavaScript can be used on all most known browsers. It can be easily used to interact with HTML elements. You can validate text fields, disable buttons, validate forms, or change the background color of your page. All this is possible with JavaScript. Like each programming language, it contains variables, arrays, functions, operators, objects and much more

which can help you to create better script for your pages. On the server side you can use JavaScript for example to manage database entry. JavaScript code can be inserted directly in the HTML or you can place it in a separate file with the .js extension and link the webpage with the .js file.

3.7.9 Django

Django is a high-level python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

• Ridiculously fast. Django was designed to help developers take applications from concept to completion as quickly as possible.

• Reassuringly secure. Django takes security seriously and helps developers avoid many common security mistakes.

• Exceedingly scalable.

3.7.10 TensorFlow

TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications. TensorFlow is a free and open-source software library for machine learning. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks. Tensorflow is a symbolic math library based on dataflow and differentiable programming. It is used for both research and production at Google. TensorFlow was developed by the Google Brain team for internal Google use. TensorFlow is Google Brain's second-generation system. While the reference implementation runs on single devices, TensorFlow can run on multiple CPUs and GPUs (with optional CUDA and SYCL extensions for general-purpose computing on graphics processing units). dimensional data arrays, which are referred to as tensors.

3.7.11 Keras

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library. Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear actionable error messages. It also has extensive documentation and developer guides. Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code. In addition to standard neural networks, Keras has support for convolutional and recurrent neural networks. It supports other common utility layers like dropout, batch normalization, and pooling. In addition to standard neural networks, Keras has support for convolutional and recurrent neural networks. It supports other common utility layers like dropout, batch normalization, and pooling.

Chapter 4

Result And Discussion

This project involves identification of facial expressions that reveal human emotions can help computers to better assess the human state of mind, so as to provide a more customized interaction. We explore the recognition of human facial expressions through a deep learning approach using a Convolutional Neural Network (CNN) algorithm.

4.1 Confusion matrix

A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. It allows the visualization of the performance of an algorithm. It allows easy identification of confusion between classes e.g. one class is commonly mislabeled as the other. Most performance measures are computed from the confusion matrix.
The important terms included in confusion matrix are as following:

- True Positive (TP) :Observation is positive, and is predicted to be positive.
- False Negative (FN) : Observation is positive, but is predicted negative.
- True Negative (TN) : Observation is negative, and is predicted to be negative.
- False Positive (FP) : Observation is negative, but is predicted positive.

The confusion matrix represent like this

		Predicted 0	Predicted 1
Actual	0	True Negative	False Negative
	1	False Positive	True Positive

Figure 4.1: Basic Representation Of Confusion matrix

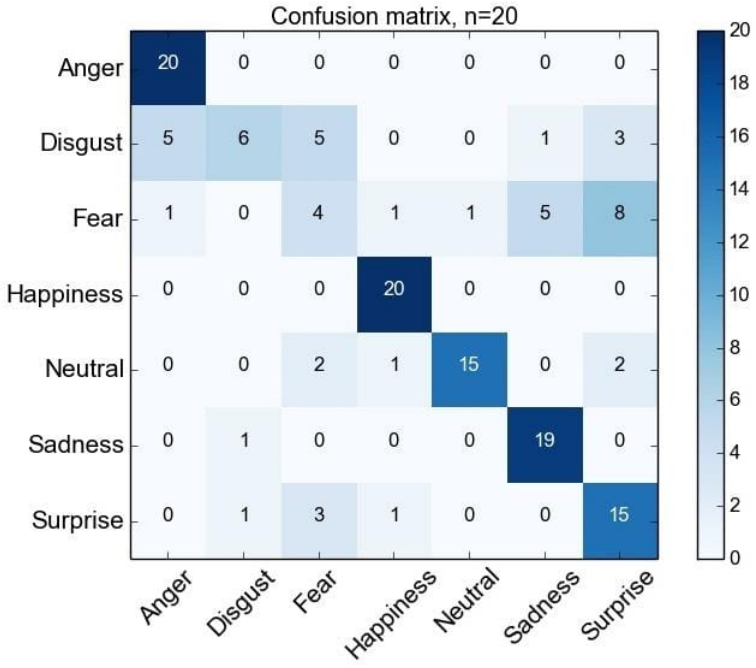


Figure 4.2: Confusion matrix

4.2 Training And Validation Results

The model trained for 100 epochs with an accuracy of 91 percent, which amount to 448 steps per epoch for training. During training there is a validation loss of 1.3259 and validation accuracy of 0.6801. The below figure 4.3 shows the details of training and testing accuracy.

```
Epoch 00097: val_loss did not improve from 0.94382
Epoch 98/100
448/448 - 17s - loss: 0.3218 - accuracy:  0.9028  val_loss: 1.3699 - val_accuracy: 0.6877

Epoch 99/100
448/448 - 17s - loss: 0.3126 - accuracy: 0.9099 - val_loss: 1.3140 - val_accuracy: 0.6773

Epoch 100/100
448/448 - 17s - loss: 0.3112 - accuracy: 0.9199 - val_loss: 1.3259 - val_accuracy: 0.6801
```

Figure 4.3: Training And Validation Result

4.3 Performance Metrics

4.3.1 Precision

Precision is the ability of a classifier not to label an instance positive that is actually negative. For each class it is defined as the ratio of true positives to the sum of true and false positives.

Precision = TP/TP+FP

4.3.2 Recall

Recall is the ability of a classifier to find all positive instances. For each class it is defined as the ratio of true positives to the sum of true positives and false negatives. Said another way,“for all instances that were actually positive, what percent was classified correctly?”

Recall =TP /TP + FN

4.3.3 f1 Score

The F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0. Generally speaking, F1 scores are lower than accuracy measures as they embed precision and recall into their computation. As a rule of thumb, the weighted average of F1 should be used to compare classifier models, not global accuracy.

f1Score =2 (Recall * precision)/ Recall + Precision

4.3.4 Support

Support is the number of actual occurrences of the class in the specified dataset. Imbalanced support in the training data may indicate structural weaknesses in the reported scores of the classifier and could indicate the need for stratified sampling or re-balancing. Support doesn't change between models but instead diagnoses the evaluation process.

The Performance metrics is shown in figure 4.4.

	Precision	Recall	F1score
Anger	1	0.77	0.43
Disgust	0.3	0.75	0.22
Fear	0.2	0.28	0.12
Happy	1	0. 87	0.47
Neutral	0.75	0.94	0.42
Sadness	0.95	0.76	0.42
Surprise	0.75	0.54	0.31

Figure 4.4: Performance Metrics

Chapter 5

Conclusion

This paper proposes a 4-layer convolution network modelfor facial emotion recognition. The model classifies 7 different facial emotions from the image dataset-angry, disgusted, fearful, happy, neutral, sad, and surprised. The model has comparable accuracy which convey that the model is having a best fit and is generalized to the data. The model uses an Adam optimizer to reduce the loss function. This system can be used to find out the changes in emotion using a video sequence which in turn can be used for different real time applications etc.

5.1 Future Enhancement

This model proposes a convolution network model for facial emotion recognition. Future scope of this work can be to do further testing with datasets containing images from various other angles. It will lead to extend this model that can recognize human facial expression from any angle in any lighting conditions and background. Also, this system can be implemented in single board computer like Raspberry Pi. This system can also be integrated with other electronic devices for their effective control.

References

[1] Sign Language Recognition System Using Deep Neural Network[S. Suresh, H. T. P Mithun and M. H. Supriya], 2019 5th International Conference on Advanced Computing Communication Systems (ICACCS)

[2] G. Cao, Y. Ma, X. Meng, Y. Gao and M. Meng, ”Emotion Recognition Based On CNN,” 2019 Chinese Control Conference (CCC), Guangzhou, China, 2019, pp. 8627-8630.doi: 10.23919/ChiCC.2019.8866540

[3] [3] Lu Lingling liu, ”Human Face Expression Recognition Based on Deep Learning-Deep Convolutional Neural Network”, 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA)

[4] [2] [4] S. Suresh, H. T. P Mithun and M. H. Supriya, ”Sign Language Recognition System Using Deep Neural Network,” 2019 5th International Conference on Advanced Computing Communication Systems (ICACCS), Coimbatore, India, 2019, pp. 614-618. doi: 10.1109/ICACCS.2019.8728411

[5] G. Yang, J. S Saumell, J Sannie, “ Emotion Recognition using Deep Neural Network with Vectorized Facial Feature” 2018 IEEE International Conference on Electro/Information Technology (EIT)

[6] [6] K. Bouaziz, T Ramakrishnan, S. Raghavan, K. Grove, A.A.Omari, C Lakshminarayan, “ Character Recognition by Deep Learning: An Enterprise solution.”, 2018 IEEE Conference on Big Data.

APPENDICES

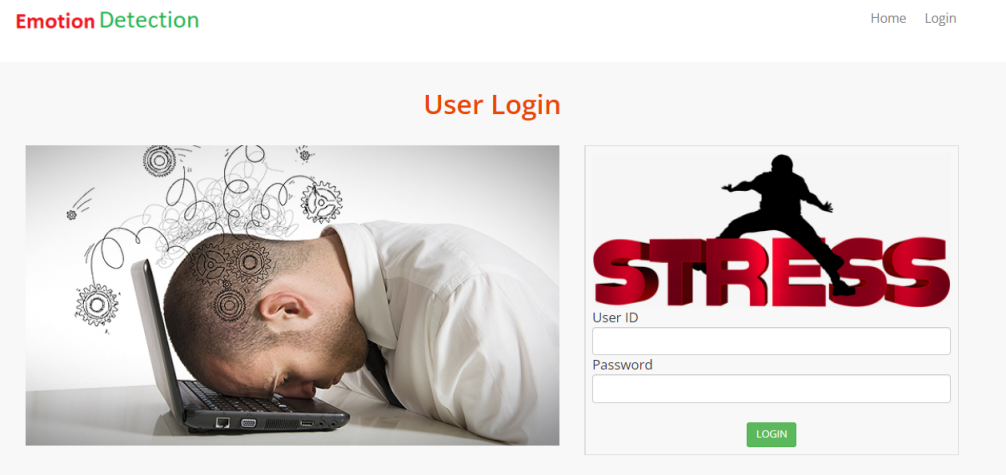


Figure 5.1: Login page

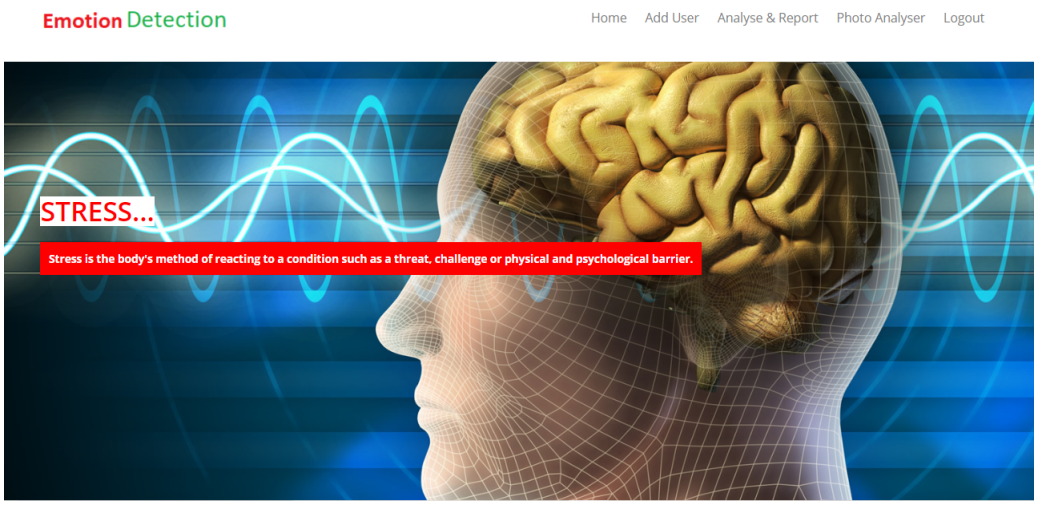


Figure 5.2: Home page

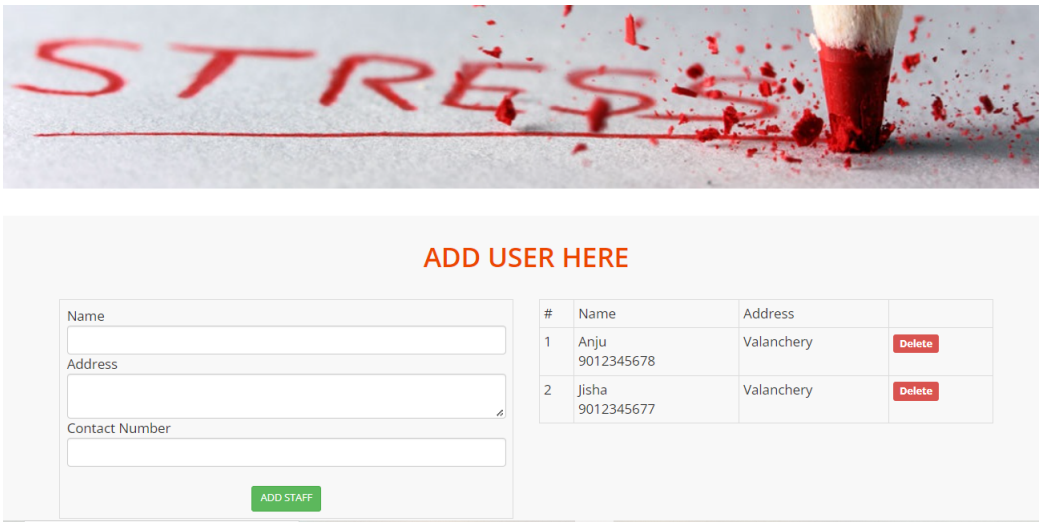


Figure 5.3: Add User

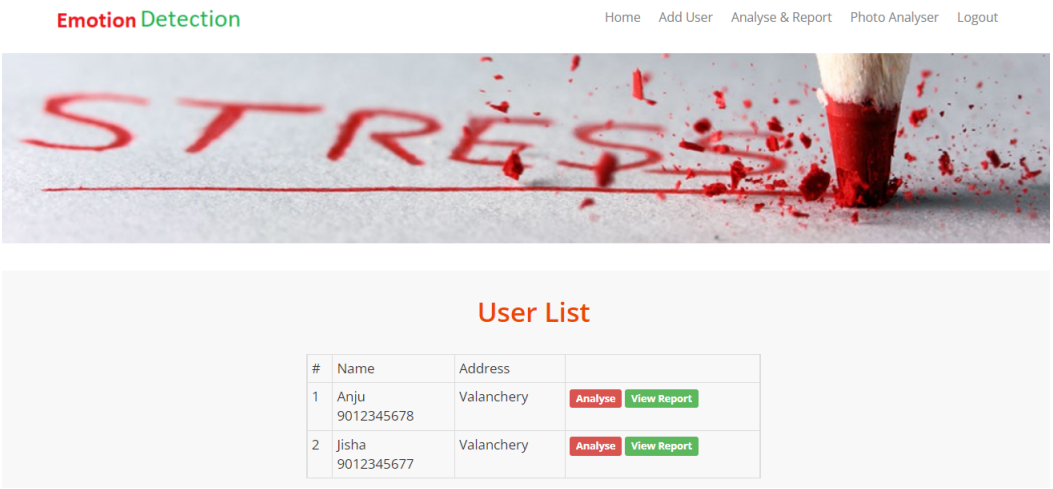


Figure 5.4: User List

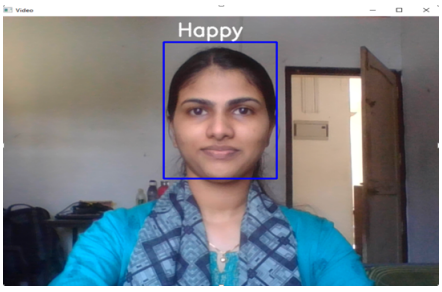
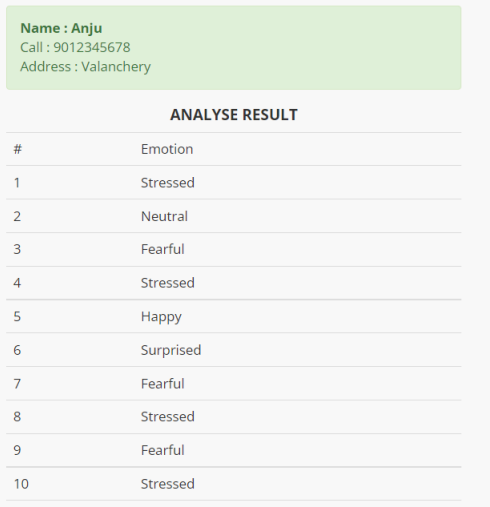


Figure 5.5: Emotion Detection



Name : Anju
Call : 9012345678
Address : Valanchery

ANALYSE RESULT

#	Emotion
1	Stressed
2	Neutral
3	Fearful
4	Stressed
5	Happy
6	Surprised
7	Fearful
8	Stressed
9	Fearful
10	Stressed

Figure 5.6: Report Analysis

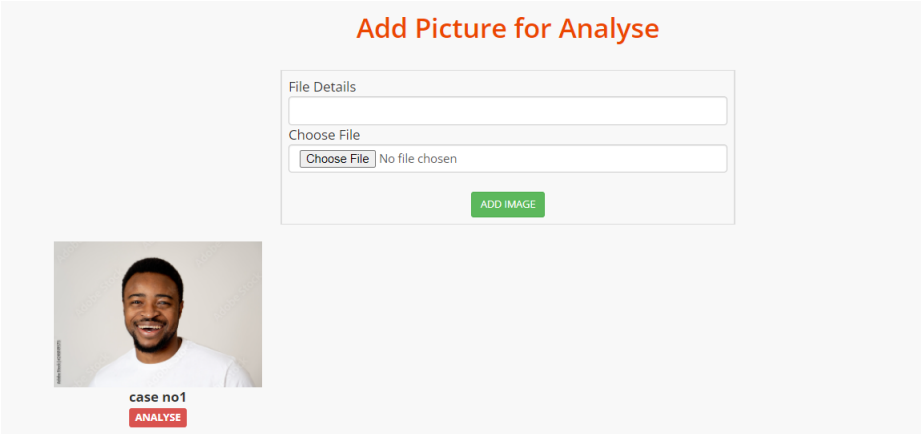


Figure 5.7: Image Analysis