A PROJECT REPORT

On

**Facial Expression Recognition**

Using

**Convolutional Neural Networks**

(Keras and Tensorflow)

Submitted in partial fulfillment for the requirement of the award of

TRAINING

IN

Data Analytics, Machine Learning and AI using Python

****

*Submitted By*

***Dhanush Sridhar***

***(College of Engineering Guindy, Anna University Chennai-25)***

*Under the guidance of*

Mr. Bipul Shahi

**Acknowledgement**

I am highly indebted to Mr. Bipul Shahi for his guidance and for providing necessary information regarding the project. This project is submitted as a part of one-month internship programme by Diginique labs (July 2020-Aug 2020) along with 15 days of training in Machine learning, Data Analytics and Artificial Intelligence using Python. I would also like to convey thanks to Diginique labs for this wonderful intern programme.

**Abstract:**

The field of computer science is rapidly advancing and is ever-growing. The concept of machine learning and AI has been evolving from 1950’s. From a Chess game that you play in computer to Automations in Industries, everything is Artificial intelligence i.e. the ability of computer to perform some tasks on its own by monitoring and sensing the surrounding environment. Artificial Intelligence is the theory and development of computer systems being able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages.

Computer Vision, a part of Artificial Intelligence, works on enabling computers to see, identify and process images in the same way that human vision does, and then provide appropriate output. It is like imparting human intelligence and instincts to a computer.

The main goal of my project is to make a machine learn to identify human facial expression. I have use CNN (Convolutional Neural Networks) model to achieve this. It was implemented using Python with the packages Keras and Tensorflow. The machine will take in an image as an input. and provide the facial expression of the person in image as Output.

Also, a Python Flask web app is providing as an interface for anyone to upload an image and predict their facial expression in that image.

**Technology and Concepts**

**Machine Learning**

Machine learning is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence.

Machine learning sounds modern, but it’s one of the oldest ideas in computer science. In 1959, a room-filling computer [called the Perceptron](https://www.wired.com/story/an-old-technique-could-put-artificial-intelligence-in-your-hearing-aid/) set a milestone in artificial intelligence when it learned to distinguish shapes such as triangles and squares.

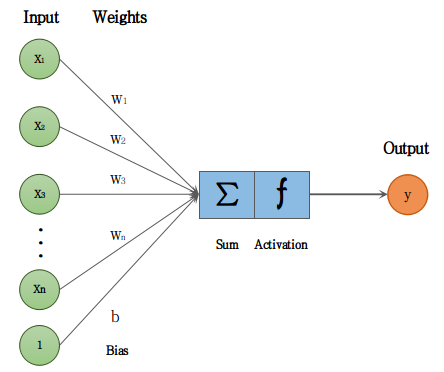
There are different approaches to getting machines to learn, from using basic decision trees to clustering to layers of artificial neural networks (the latter of which has given way to deep learning), depending on what task you’re trying to accomplish and the type and amount of data that you have available. This dynamic sees itself played out in applications as varying as medical diagnostics or self-driving cars.

It can be Classified as:

* *Supervised Learning – Exposing data from past to machine to predict the future results.*
* *Unsupervised Learning – Understanding underlying pattern and learn from it.*

**Neural networks**

Another algorithmic approach from the early machine-learning crowd, artificial neural networks, came and mostly went over the decades. [Neural networks](https://developer.nvidia.com/discover/artificial-neural-network) are inspired by our understanding of the biology of our brains – all those interconnections between the neurons. But, unlike a biological brain where any neuron can connect to any other neuron within a certain physical distance, these artificial neural networks have discrete layers, connections, and directions of data propagation.



Neuron

You might, for example, take an image, chop it up into a bunch of tiles that are inputted into the first layer of the neural network. In the first layer individual neurons, then passes the data to a second layer. The second layer of neurons does its task, and so on, until the final layer and the final output is produced.

Each neuron assigns a weighting to its input — how correct or incorrect it is relative to the task being performed. The final output is then determined by the total of those weightings.

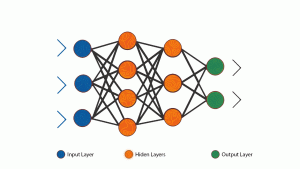
So, think of our stop sign example. Attributes of a stop sign image are chopped up and “examined” by the neurons — its octagonal shape, its fire-engine red colour, its distinctive letters, its traffic-sign size, and its motion or lack thereof.



The neural network’s task is to conclude whether this is a stop sign or not. It comes up with a “probability vector,” really a highly educated guess, based on the weighting. In our example the system might be 86% confident the image is a stop sign, 7% confident it’s a speed limit sign, and 5% it’s a kite stuck in a tree, and so on — and the network architecture then tells the neural network whether it is right or not.

Types of Neural Networks:

1. **Artificial Neural Network**, or ANN, is a group of multiple perceptron’s/ neurons at each layer. ANN is also known as a **Feed-Forward Neural network** because inputs are processed only in the forward direction:

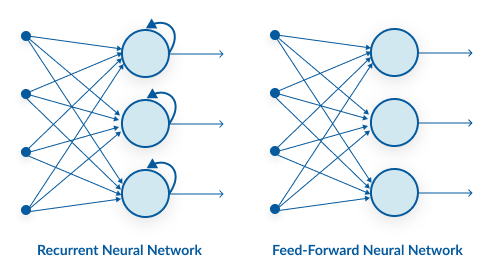
[](https://cdn.analyticsvidhya.com/wp-content/uploads/2020/02/ANN-Graph.gif)

ANN consists of 3 layers – Input, Hidden and Output. The input layer accepts the inputs, the hidden layer processes the inputs, and the output layer produces the result. Essentially, each layer tries to learn certain weights.

1. **Convolutional neural networks** were [inspired](https://en.wikipedia.org/wiki/Mathematical_biology) by [biological](https://en.wikipedia.org/wiki/Biological) processes in that the connectivity pattern between [neurons](https://en.wikipedia.org/wiki/Artificial_neuron) resembles the organization of the animal [visual cortex](https://en.wikipedia.org/wiki/Visual_cortex). Individual [cortical neurons](https://en.wikipedia.org/wiki/Cortical_neuron) respond to stimuli only in a restricted region of the [visual field](https://en.wikipedia.org/wiki/Visual_field) known as the [receptive field](https://en.wikipedia.org/wiki/Receptive_field). The receptive fields of different neurons partially overlap such that they cover the entire visual field.



1. **Recurrent Neural Network -** A feedback ANN

RNN captures the sequential information present in the input data. 

RNN

ANN

**Programing Language and libraries used:**

One of the aspects that makes [Python such a popular choice in general](https://www.netguru.com/blog/why-python-is-growing-so-quickly-future-trends), is its abundance of libraries and frameworks that facilitate coding and save development time. Machine learning and deep learning are exceptionally well catered for.

NumPy, used for scientific computation, SciPy for advanced computation, and scikit-learn for data mining and data analysis, are among the most popular libraries, working alongside such heavy-hitting frameworks as TensorFlow, CNTK, and Apache Spark, Keras+-. In terms of machine learning and deep learning, these libraries and frameworks are in essence Python-first, while some, like PyTorch, are written specifically for Python. Open-cv is used for Computer vision.

This project uses –

* Numpy, Pandas (For Advanced Computations and Data handling),
* Tensorflow, Keras (For Creating CNN models),
* cv2(for working with images) \

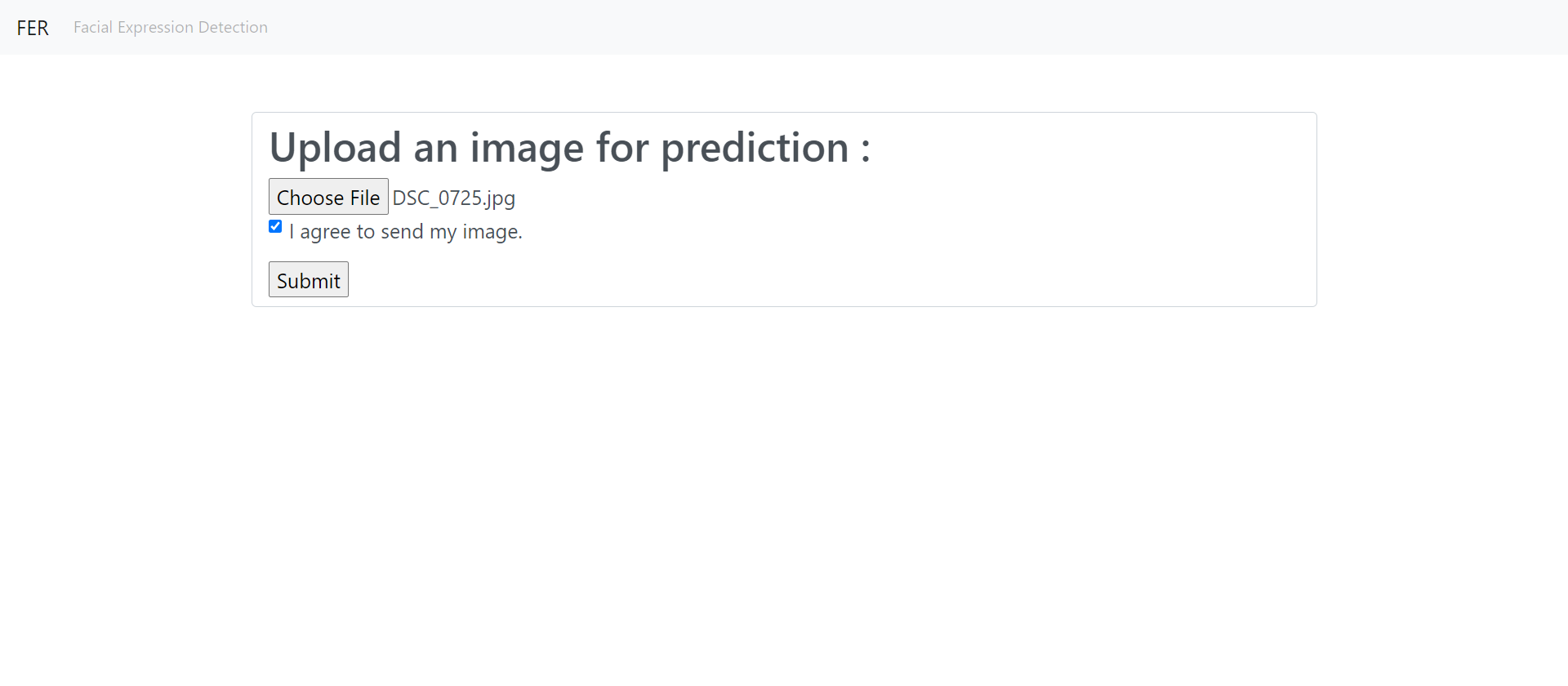
as major libraries.

Also, Python Flask framework is used to create a web interface for user to interact with the model. i.e. Users can upload image and predict the facial expression of the person in the image.

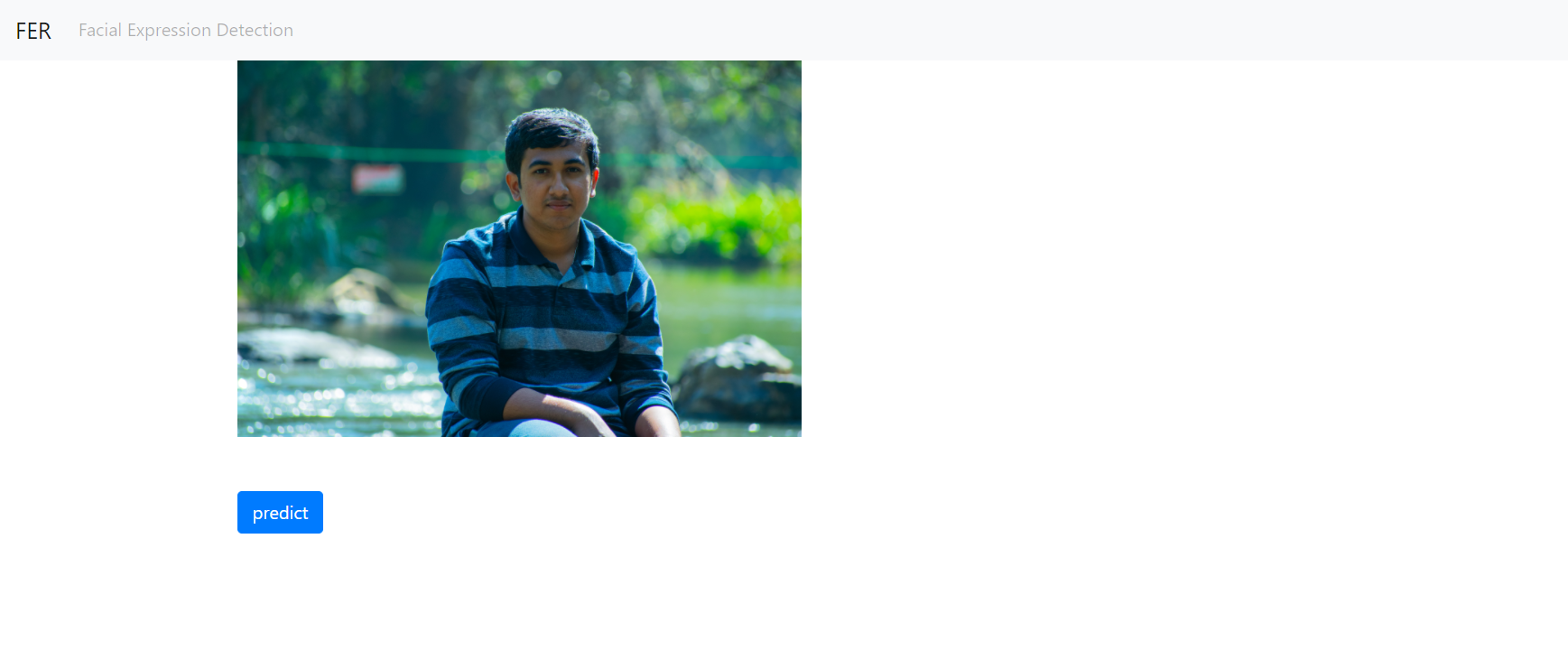
**A sample prediction:**

I will use the web interface and show a demo.

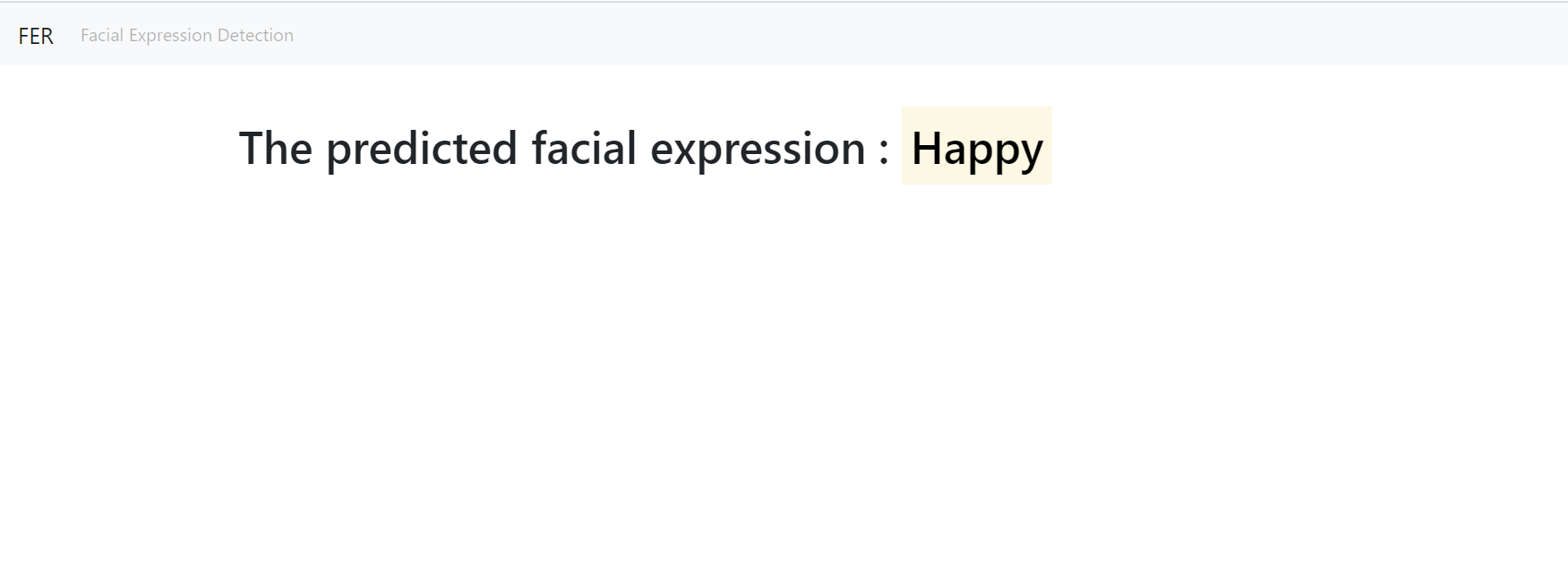
* Upload the image by clicking the “Choose File” option.
* Check the check box.
* Click Submit.



* Image uploaded will be displayed.
* Click predict button to predict the image.



* Your Facial expression will be predicted.



Conclusion:

We have created a CNN model which would predict facial expression of a human in an image.

Thus a model with 75% accuracy has been created and is ready for deploying. Further faces of all the people can be detected and the expression can be predicted.

This is available in github:

<https://github.com/DhanushSridhar26/FacialExpressionDetection>

The ipynb file is here:

<https://github.com/DhanushSridhar26/FER-python-code/>