### FACIAL EMOTION RECOGNITION

**A.SRI KALYAN REDDY**

**(19STUCHH010130)**



##### Faculty of Science and Technology

**ICFAI University, Hyderabad-501203, Telangana, India.**

### FACIAL EMOTION RECOGNITION

*Project report submitted in partial fulfilment of the requirements for the degree of*

##### Special project

**In**

##### Data science and artificial intelligence

*by*

**A.SRI KALYAN REDDY**

**(19STUCHH010130)**

Under the guidance of

###### Dr. Shadab Ahmad

Assistant Professor

Department of Data Science & Artificial Intelligence ICFAI University, Hyderabad

##### Faculty of Science and Technology

**ICFAI University, Hyderabad-501203, Telangana, India**

**Faculty of Science and Technology**

ICFAI University, Hyderabad-501203, Telangana, India.

**Declaration**

I hereby certify that the work presented in this thesis entitled “**FACIAL EMOTION RECOGNITION**" in fulfilment of the requirements for the Special project Data science & Artificial intelligence and submitted to ICFAI University, Hyderabad is an authentic record of my own work carried out during the period from August 2022 to November 2022, under the supervision of **Dr. Shadab Ahmad**

The matter embodied in this thesis has not been submitted by me for the award of any other degree of this or any other University/Institute.

This is to certify that the above statement made by the scholar is correct to the best of our knowledge.

**A.SRI KALYAN REDDY**

**(19STUCHH010130)**

The Special project Viva-voce examination ofA.SRI KALYAN REDDY (19STUCHH010130) fourth Year Student, which has been held on . . . . . . . . . . . . .and is recommended for the award of the B. Tech degree.

**Sign of Supervisor**

##### Faculty of Science and Technology

ICFAI University, Hyderabad-501203, Telangana, India.

**Certificate**

This is to certify that the project entitled “FACIAL EMOTION RECOGNITION**”,** A.SRI KALYAN REDDY (19STUCHH010130) submitted by in partial fulfilment of the requirement for the Special project in *Computer Science & Engineering of ICFAI University, Hyderabad, Telangana, India*, is a bonafide record of original research work carried out by him under my guidance and supervision.

This work has not been submitted elsewhere for the same purpose.

Assistant Professor

ICFAI University Hyderabad.

###### Dr. [Shadab Ahmad](mailto:shadab.ahmad@ifheindia.org)

Data science and artificial intelligence

**Faculty of Science & Technology, IFHE University**

**Station: ICFAI tech, India, Hyderabad**

**Centre: Hyderabad**

**Duration: Date of Start: 18/08/2021 Date of Submission: 05/12/2021**

**Title of the Project: EMOTION DETECTION**

**Name of the Student: AKITI SRI KALYAN REDDY**

**ID. No:19STUCHH010130**

**Discipline: DS&AI**

**Name of the faculty mentor: Dr. Shadab Ahma**

**Project Areas: Machine learning, model and CNN**

**Signature of Student Signature of faculty 05-12-2021 05-12-2021**

**TABLE OF CONTENTS**

**ABSTRACT 7**

CHAPTER-1: **INTRODUCTION 8**

CHAPTER-2: **LITERATURE SURVEY**   **9**

CHAPTER-3: **MOTIVATION 11**

CHAPTER-4: **IMPORTING LIBRARIES AND MODULES 12**

CHAPTER-5: **CNN-CONVENTIONAL NEURAL NETWORK 15**

CHAPTER-6: **SEQUENTIAL MODEL AND LAYERS 18**

CHAPTER-7: **DATASET 20**

CHAPTER-8: **IMPLEMENTATION OF EMOTION DETECTION 23**

CHAPTER-9: **RESULT 26**

CHAPTER-10: **CONCLUSION 27**

**REFERENCES 28**

**ABSTRACT**

We all have emotions which are dependent on different situations and people express their emotions on their faces in real life to demonstrate their psychological activities and attitudes in interactions with others. Here our task is processing a human facial expression and categorising it by using an Emotion Detection extraction which helps in facial recognition. We are going to use a method of recognising an emotion by a model for detecting, extracting, and evaluating the facial expressions which will enable automatic recognition of human emotion in images and videos. Because the human face is difficult to interpret, emotion recognition can be further subdivided into basic emotion classification and compound emotion classification.

In this project, we address the problem by employing a Conventional Neural Network (CNN). Here the model typically detects the emotions, and the trained classifier produces the output based on the given features. The algorithm can achieve reasonable accuracy, depending on the testing set and test emotions.

KEY WORDS: Conventional Neural Network (CNN), Emotion Detection, facial Recognition

**CHAPTER 1 - INTRODUCTION**

In this task, our goal is to build a model which predicts the emotion of the person using the images on different highlights like grayscale dimension and other useful models. We will play out every one of the means from importing libraries, modules, Data to Model convention. During Model assessment, we analyse different AI calculations based on accuracy score metric and track down the best one. Then, at that point, we make a model utilizing some of these features which is a python miniature structure.

We should assign ourselves to gather information about models which is a crucial step for getting best results. For this project we are going to rely on different types of sources. For example, to find best information for which model we are going to use. That information would be useful in our next process. We can’t ignore that it should be based on facts for accurate data, here the task is to find information on super resolution and also the information suggested by famous personalities on the same topic. After brief research over the internet, we might find useful information which will be useful in making our project.

Many traditional problems can be framed as image transformation tasks, in which a system receives an input image and transforms it into an output image. Denoising, emotion detection, and colorization are image processing examples where the input is an image (noisy, low-resolution, or grayscale) and the output is capable of detecting the emotion. Semantic segmentation and depth estimation are two computer vision examples where they input images and try to get the output as emotion, encodes semantic or geometric information about the scene. (springer) [1]

**CHAPTER 2 - LITERATURE SURVEY**

**INTRODUCTION**

The project FACIAL EMOTION RECOGNITION allows you to analyse a single person's emotion using this emotion detector. Researchers noted that by developing an effective emotional detector in the medical field, where it may be possible to recognize patients' emotions, in automatic driving systems it can improve the driving experience and in the marketing industry, particularly in super markets where we can observe whether or not a customer is satisfied with a product [2].

Traditional research has shown that areas in the image processing process have a tendency to produce better results. On varied scales as well, to make the most of them. Additional layers can be used to increase the effectiveness of the model, but in the current environment, a deep learning-based technique can also produce good results. In this research, we describe a dilated convolution-based approach to learning about convolutional neural networks. Where we'll build a deep network to identify facial emotions [3]. There will be several dilated convolutional layers for it to learn. We increased the size of the welcoming area of our network to gather more contextual information.

where, we will collect the data that has been broken down into various classes and label it in a sequential manner. There are numerous emotions that a human can experience in facial emotion detection. Therefore, we will concentrate on a few courses and train a model that can detect face emotion [4]. The machine learning model will then attempt to differentiate between various inputs before producing results. It will use facial recognition to find human faces and artificial intelligence techniques like neural networks to analyse human emotions.

**LITERATURE REVIEW**

Prior research has concentrated on getting data from unimodal systems. Machines that can solely read facial expressions [5] or vocal sounds have been used to predict emotion. Multimodal systems that employ many characteristics to predict emotion become more successful and produce more precise findings with time. Consequently, since then, a combination of elements including audio-visual emotions, EEG, and body gestures has been applied. The emotion recognition system is implemented using neural networks and many intelligent machines. [6] Shiqing et al multimodal’ s recognition method has been shown to be more successful than unimodal systems. Deep neural networks can successfully produce discriminative characteristics that roughly replicate the intricate non-linear correlations between features in the original set, according to research. These deep generative models have been used for speech and language processing, as well as emotion recognition. Martin et al [7]. demonstrated that a bidirectional Long Short-Term Memory (BLSTM) network outperforms a conventional SVM approach. Recurrent neural networks in conjunction with Convoluted Neural Networks (CNN) in an underlying CNN-RNN architecture were utilised by Samira et al. [8] to predict emotion in the video.

**CHAPTER 3 - MOTIVATION**

At present we are using AI to recognize the primary images and to make some of the best models while using data [10]. In exploring more about AI, I found enhancers and models in AI. I got more questions like How do they work for a practical image? And what are the needs.

Emotion detection is an especially challenging, unclear problem that aims to predict the emotion of a person as output. Recently, effective deep studying algorithms were implemented to emotion detection and feature completed modern-day performance.

Fascinated by Image classification and Computer vision development which use deep convolutional neural networks to implement the model.

I am very aware of the situation in the AI industry where Computer Vision related projects are extremely in demand.

I also consider this a learning opportunity that helps us learn and understand in demand cutting-edge technology that is currently booming in the Computer Vision Industry.

I was interested in the photoshop industry where an image can be converted white and black into a colorized image. Here these types of models are useful and save a lot of time. This is what amazes us how an AI can understand and develop a clearance interface by itself without any help from humans and use it to enhance and change the world.[11] What is more interesting is AI doing all things by itself and what we need to do is just give an image. By evaluating this I thought this would be the best project to do and I might be able to contribute from my side to get best results in this field.

Through this project I hope to develop a better understanding of the technologies mentioned above.

After learning about the drawbacks of the current system, we will create this project. The technique begins by building a CNN model from an input image (static or dynamic) by adding convolutional, pooling, flattening, and dense layers to the image. Convolution coatings will be added for larger datasets to improve accuracy. The dataset is gathered from a CSV file (in pixels format), translated to pictures, and then categorised according to the appropriate facial expressions. With lots of photographs for the training dataset and enormous images for testing, emotions are categorized as joyful, sad, angry, surprised, neutral, disgust, and fear in this instance. Different facial features, such as raised cheekbones, creases around the nose, wide-open eyelids, and many more, are used to represent each emotion. For improved accuracy and a result that would be the object class for an input image, a huge dataset was trained. It conducts convolution layers and max pooling based on certain characteristics. With the aid of Keras [9] which consists a variety of Models, the suggested system uses deep learning. The classification of human face emotion is included in these models.

**CHAPTER 4 - IMPORTING LIBRARIES AND MODULES**

**VSCODE:**

Visual Studio Code is a lightweight, however incredible source code supervisor which runs on your work area and is accessible for Windows, macOS and Linux. It accompanies worked in help for JavaScript, TypeScript and Node.js and has a rich biological system of expansions for different dialects (like C++, C#, Java, Python, PHP, Go) and runtimes, (for example, .NET and Unity). Start your excursion with VS Code with these initial (“VSCODE”)[12]

**PYTHON:**

Python is a deciphered, object-arranged, undeniable level programming language with dynamic semantics. Its significant level of implicit information structures, joined with dynamic composing and dynamic restricting, make it exceptionally appealing for Rapid Application Development, just as for use as a pre-arranging or paste language to associate existing parts together. Python's straightforward, simple to learn sentence structure underscores intelligibility and accordingly diminishes the expense of program upkeep. Python upholds modules and bundles, which empowers program measured quality and code reuse. The Python mediator and the broad standard library are accessible in source or parallel structure without charge for every single significant stage, and can be openly dispersed. (“python”) [13]

**Importing libraries:**

As for now we are going to import some of the libraries and tools and we'll utilize Python and a portion of its well-known information science related bundles, so that we’ll be able to perform operations to our project in a linear way and learn about machine learning so that we can understand the basic concepts of the machine learning and how to do our project work in a convenient environment

We will be using vs code or collab to carry out the executions and obtain the required output. In the system we will be using the following the libraries:

1. Kearas
2. Keras.modules
3. Keras.optimizers
4. ImageDataGenerator
5. Os.path
6. Glob
7. CV 2
8. Numpy
9. ImageDataGenerator
10. Pandas
11. Seaborn
12. Matplotlib
13. Sklearn
14. Tensorflow

**Libraries Defined:**

**1) Keras:** Google created the high-level Keras deep learning API to implement neural networks. It is used to make the implementation of neural networks simple and is developed in Python. Additionally, various backend neural network computation is supported.[14]

**2) keras.modules:** The fundamental units of Keras' neural networks are layers. A layer is made up of a tensor-in tensor-out calculation function (the layer's call method) and some state (the layer's weights) that is stored in TensorFlow variables.[15]

**3) keras.optimizers:** One of the two inputs needed to compile a Keras model is an optimizer. An optimizer can be passed to model.compile() either already instantiated, like in the example above, or by its string identifier. The optimizer's default parameters will be applied in the latter scenario [16].

**4)** **ImageDataGenerator:** The input of the original data is obtained using the Keras ImageDataGenerator, which then transforms the input data randomly and outputs a result that solely contains the newly altered data. The data is not added. In order to increase the generalisation of the model overall, data augmentation is also carried out using the Keras picture data generator class. With the use of an image data generator, data augmentation operations including rotations, translations, scale adjustments, and horizontal flips are performed at random [17].

**5) Os.path:** This module implements some useful pathname functions. Open() is used to read and write files, and the OS module is used to access the filesystem. Path parameters can be strings, bytes, or any object that implements the OS.PathLike protocol.[18]

**6)** **glob:** The glob module uses the Unix shell's rules to identify all path names that match a particular pattern, though the results are returned in any order. There is no tilde expansion, but character ranges indicated with \*,?, and [] will be accurately matched. This is accomplished by combining the os.scandir() and fnmatch.fnmatch() routines rather than running a subshell. Unlike fnmatch.fnmatch(), glob considers filenames that begin with a dot (.) to be special circumstances. (“os.path — Common pathname manipulations — Python 3.10.4 documentation”)[19]

**7)** **OpenCV** packages for Python that are CPU-only. If you want to compile the bindings from source to enable other modules like CUDA, we can consult the manual build section. (“opencv-python · PyPI”) [20]

**8)** **Numpy:** We will likewise utilize numpy to change our information into an organization reasonable to take care of our grouping model. (“NumPy Tutorial”) [21]

**9)** **OS:** The Python OS module allows for the establishment of interaction between the user and the operating system. It provides many useful OS functions that are used to perform OS-based tasks and obtain operating system-related information.[22]

**10)** **Pandas:** we will import panda's to peruse our information from a CSV record and control it for additional utilization. (“Python Pandas Tutorial”) [23]

**11)** **Seaborn**: Seaborn is a matplotlib-based Python data visualization package. It has a high-level interface for creating visually appealing and instructive statistics visuals. (“An introduction to seaborn — seaborn 0.11.2 documentation”) [24]

**12) Matplotlib:** We'll utilize ocean conceived and Matplotlib for representations (“Matplotlib documentation — Matplotlib 3.5.2 documentation”) [25]

**13) Sklearn:** sklearn This calculation will assist us with building our order model (“scikit-learn Tutorials — scikit-learn 1.0.2 documentation”)[26]

**14)** **Tensorflow**: While Keras is a high-level neural network framework that works on top of TensorFlow, TensorFlow is an open-sourced end-to-end platform, library for many machine learning applications. Both offer high-level APIs that make it simple to design and train models, while Keras is more approachable due to the inclusion of Python.[27]

**CHAPTER 5 - CNN (CONVENTIONAL NEURAL NETWORK)**

The IT industry has seen a surge in demand for a specific skill set known as Deep Learning in recent years. Deep Learning is a subset of Machine Learning that includes algorithms inspired by the operation of the human brain or neural networks [28].

Neural Networks is the name given to these structures. It teaches the computer to do what humans do naturally. Deep learning models include Artificial Neural Networks (ANN), Autoencoders, Recurrent Neural Networks (RNN), and Reinforcement Learning [29]. However, one model, in particular, has made significant contributions to computer vision and image analysis, and that is the Convolutional Neural Networks (CNN) or Conv Nets [30].

CNNs are a type of Deep Neural Network that can recognize and classify specific features in images and are commonly used in image analysis [31]. Image and video recognition, image classification, medical image analysis, computer vision, and natural language processing are among their uses.

The mathematical function of convolution, a particular sort of linear operation in which two functions are multiplied to produce a third function that expresses how the shape of one function is modified by the other, is denoted by the term "convolution" in CNN [32]. Simply, two matrices are multiplied to provide an output used to extract features from the image.

###### CNN Architecture

**A CNN architecture is typically divided into three components:**

1. Convolutional layer
2. Pooling Layer
3. Fully connected Layer

In a process known as Feature Extraction, a convolution tool isolates and identifies the distinct characteristics of an image for analysis [33].

A fully connected layer that uses the output of the convolution [34] process to forecast the image's class using the information acquired in previous stages.

Artificial neurons are arranged in numerous layers to form convolutional neural networks. Artificial neurons are mathematical functions that compute the weighted sum of several inputs and output an activation value, roughly imitating their biological counterparts. Each layer of a ConvNet creates a number of activation functions that are passed on to the following layer when an image is entered.

Typically, the first layer extracts fundamental features like edges that run horizontally or diagonally. The following layer receives this output and detects more complex features like corners or combinational edges. The network may identify increasingly more complex elements, including objects, faces, etc., as we go further into it [35].

The classification layer generates a series of confidence ratings (numbers between 0 and 1) that indicate how likely it is for the picture to be a member of a "class," based on the activation map of the final convolution layer. The output of the last layer, for instance, can be the likelihood that the input image contains any of the cats, dogs, or horses detected by the ConvNet.

It is essential to use visual tools when discussing CNN's architecture in order to communicate about the created models in an effective manner. These tools assist in the creation of CNN diagrams by attractively visualising the model [36].

###### 

###### Non-Linearity Layers

Non-linearity layers are frequently included right after the convolutional layer to add non-linearity to the activation map because convolution is a linear operation and images are not at all linear [36].

Frequently used non-Linearity layers are:

* **Sigmoid:** The mathematical formula for the sigmoid nonlinearity is σ(κ) = 1/(1+e¯κ). It maps a real-valued number into the range between 0 and 1. The gradient of a sigmoid becomes almost zero when the activation is at either tail, which is a very unfavourable sigmoid characteristic. Backpropagation will effectively destroy the gradient if the local gradient gets too small. Additionally, if the input to the neuron is exclusively positive, the output of the sigmoid will either be exclusively positive or exclusively negative, leading to a zigzag dynamic of gradient updates for weight.
* **Tanh:** A real-valued number is mapped by Tanh to the range [-1, 1]. Similar to sigmoid neurons, the activation saturates, but unlike them, its output is zero-centred.
* **ReLU:** The Rectified Linear Unit (ReLU) performs the function ƒ(κ)=max (0, κ) computation i.e., the activation just exists at zero threshold. ReLU speeds up convergence by six times hence it is more dependable than sigmoid and tanh. There is a huge possibility that ReLU could display fragility during training, which is a drawback. It can be updated by a large gradient that prevents the neuron from ever updating further. Nevertheless, by choosing an appropriate learning rate, we can make this work.
* **LeakyReLu:** It is a type of activation function based on a Rectified Linear Unit (ReLU) known as a Leaky Rectified Linear Unit (Leaky ReLU), which has a tiny slope for negative values as opposed to a flat slope. The slope coefficient is not acquired through training rather predetermined before training.[37]

**CHAPTER 6 - SEQUENTIAL MODEL AND LAYERS**

In this project we are going to use a sequential model for CNN and in these there are different types of layers such as Convolutional layers, pooling layers, and fully-connected (FC) these types of layers that make up the CNN. A CNN architecture will be constructed when these layers are stacked. There are two more crucial factors, the dropout layer, and the activation function, that are defined below in addition to these layers.

### ● Sequential model

The Sequential model is a linear stack of layers. The common architecture of ConvNets is a sequential architecture. However, some architectures are not linear stacks [38]. For example, siamese networks are two parallel neural networks with some shared layers.

### ● Convolutional Layer

This is the initial layer that extracts the different features from the input photos. The convolution mathematical process is done between the input image and a filter of a specific size MxM in this layer. The dot product between the filter and the sections of the input image with regard to the size of the filter is taken by sliding the filter across the input image (MxM)[39].

The Feature map is the result, and it contains information about the image such as its corners and edges. This feature map is then supplied to further layers, which learn a variety of other features from the input image.

### ● Pooling Layer

A Pooling Layer is usually applied after a Convolutional Layer. This layer's major goal is to lower the size of the convolved feature map in order to reduce computational expenses. This is accomplished by reducing the connections between layers and operating independently on each feature map [40]. There are numerous sorts of Pooling operations, depending on the mechanism used.

The largest element is obtained from the feature map in Max Pooling. The average of the elements in a predefined sized Image segment is calculated using Average Pooling. Sum Pooling calculates the total sum of the components in the predefined section. The Pooling Layer is typically used to connect the Convolutional Layer with the FC Layer.

### ● Fully Connected Layer

The Fully Connected (FC) layer connects the neurons between two separate layers by combining the weights and biases with the neurons. The last several layers of a CNN Architecture are usually positioned before the output layer.

The previous layers' input images are flattened and supplied to the FC layer in this step [41]. After that, the flattened vector is sent via a few additional FC layers, where the mathematical functional operations are normally performed. The classification procedure gets started at this point.

### ● Dropout

When all of the characteristics are connected to the FC layer, the training dataset is prone to overfitting. Overfitting happens when a model performs so well on training data that it has a negative impact on its performance when applied to new data.

A dropout layer is used to solve this problem, in which a few neurons are removed from the neural network during the training process, resulting in a smaller model [42]. After passing a dropout of 0.3, 30% of the nodes in the neural network are randomly removed.

### ● Activation Functions

Finally, the activation function is one of the most crucial elements in the CNN model. They are used to learn and approximate any continuous or complex relationship between network variables. In simple terms, it determines which model information should fire in the forward direction and which should not at the network's end.

It gives the network non-linearity. The ReLU, SoftMax, tanH, and Sigmoid functions are some of the most often utilized activation functions. Each of these functions has a distinct use. The sigmoid and SoftMax functions are preferred for binary classification CNN models, and SoftMax is frequently employed for multi-class classification [43].

**●** **GlobalAveragePooling2D**

Tensorflow.js is an open-source toolkit created by Google for running deep learning neural networks and machine learning models in the browser or on the node platform [44]. Additionally, it enables programmers to build machine learning models in JavaScript and use them with Node.js or directly in the browser. The global average pooling procedure for geographic data is applied using the tf.layers.globalAveragePooling2d() function.

### ● Flatten layer

The input's spatial dimensions are condensed by a flatten layer into its channel dimension [45]. When your project is complete and you want to make all of your layers into a background image since you no longer want to change them, Flatten Image is advised. In this manner, you can export the final image and utilize it however you like.

### ● MaxPooling2D

2DMaximum pooling for 2D spatial data. The input is down sampled along its spatial dimensions (height and width) by taking the maximum value for each input channel over an input window with a size determined by pool size [46]. Each dimension of the window is moved one step at a time.

**CHAPTER 7 - DATASET**

The information was gathered and made accessible by "**FER - 2013** and correspondence of this fer-2013 team as a component of the image database. Several limitations were put on the determination of these examples from a bigger data set. Specifically, all participants in the team have a place within the collection of data (which is in public domain) [47].

Here the data is described in labels = (Anger, Disgust, Fear, Happy, Neutral, Sad, Surprise)

In this, the first task we should assign ourselves is to gather information which is a crucial step in any data collection. For this project we are going to rely on different types of sources. For example, to find a few information whose topic would be emotion detection. That information would be useful in our next process. We can’t ignore that it should be based on facts for accurate data, here the task is to find information on emotion detection and also the information suggested by famous personalities on the same topic. After brief research over the internet, we might find useful information. We should also provide some links of that information online, which are to be documented [48].

Information assortment is characterized as the method of gathering, estimating and investigating precise bits of knowledge for research utilizing standard approved strategies. A scientist can assess their theory based on gathered information. Much of the time, information assortment is the essential and most significant stage for research, independent of the field of exploration. The methodology of information assortment is diverse for various fields of study, contingent upon the necessary data.

The most basic goal of information assortment is guaranteeing that data rich and dependable information is gathered for measurable investigation so information driven choices can be made for research.

**Angry fig (1) Sad fig (2) Happy fig (3) Surprised fig (4) Neutral fig (5)**

**DATA SET**

For training, we mainly use the FER - 2013 dataset [49], which is a high-quality dataset for image restoration tasks. Beyond the training set of Fer -2013 that contains 36K images, we also seek for other datasets with rich and diverse textures for our training. To this end, we further use the Fir datasets consisting of images collected on the Fir website, and other datasets to enrich our training set. We empirically find that using this large dataset with richer textures helps the generator to produce more natural results, as shown in Fig. 1. We train our models in RGB channels and augment the training dataset with random horizontal flips and 90-degree rotations.

**Data overview**

Newly collected dataset images with a large diversity of contents. The data consists of 48x48 pixel grayscale images of faces. The faces have been automatically registered so that the face is more or less centred and occupies about the same amount of space in each image.

**The FER-2013 dataset is divided into:**

Train data: starting from 3000 images we obtain up to 28k images in this folder.

Validation data: This generated by using ImageDataGenerator [50]

Test data: This dataset contains of 3k images which are of resolution 48 \*48

**Data structure**

FIR -2013 dataset has the following structure:

38k images divided into: 28k images for training, generated images for validation,3k images for testing

For each challenge Track we have:

The images: 0001.png, 0002.png, ..., 28000.png

The task is to categorize each face based on the emotion shown in the facial expression into one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). The training set consists of 28,709 examples and the public test set consists of 3,589 examples.

**Example of data collection:**

Information assortment is a significant part of exploration. We should think about an illustration of a portable maker, organization X, which is dispatching another item variation. To direct research about highlights, value range, target market, contender examination and so on information must be gathered from suitable sources. The advertising group can direct different information assortment exercises, for example, online reviews or centre gatherings.

The overview ought to have the appropriate inquiries concerning highlights and evaluating, for example, "What are the main 3 elements anticipated from an impending item?" or "What amount are your liable to spend on this item?" or "Which contenders give comparable items?" and so on

For leading a centre gathering, the showcasing group ought to choose the members just as the middle person. The subject of conversation and objective behind directing a centre gathering ought to be clarified ahead of time so a convincing conversation can be led.

Information assortment strategies are picked relying upon the accessible assets. For instance, leading polls and reviews would require the least assets while centre gatherings require reasonably high assets.

**CHAPTER 8 - IMPLEMENTATION OF EMOTION DETECTION**

It is a very straightforward task to implement this model.

Firstly, create a file named as emotion detection and make sure that these folders are located in the same directory in which the "test.py" file where images need to be tested then we are going to import libraries such as matplotlib.pyplot, numpy, pandas, seaborn, OS

**IMPORTING LIBRARIES FROM KERAS LAYERS**

We import layers Dense, Input, Dropout, GlobalAveragePooling2D, Flatten, Conv2D, BatchNormalization, Activation, MaxPooling2D, LeakyReLu.

**IMPORTING LIBRARIES FROM KERAS MODELS**

We import model which is Sequential model

**IMPORTING LIBRARIES FROM KERAS OPTIMIZERS**

We import the optimizer Adam [51]

Adam Optimizer: It is a stochastic gradient descent technique called Adam optimization is based on adaptive estimation of first- and second-order moments.

**IMPORTING LIBRARIES AND KERAS PREPROCESSING OF IMAGE**

We import ImageDataGenerator and add images to the folder by using the location of the data set’s by using variables. The next step involves data generation by using ImageDataGenerator. Keras ImageDataGenerator is used to take in the original data and then randomly transform it before returning the output result containing only the newly transformed data. It does not include the information. The Keras image data generator class is also used for data augmentation, with the goal of increasing the model's generalisation. In data augmentation, operations like rotations, translations, shearin, scale changes, and horizontal flips are performed at random using an image data generator.[52]. By using flow\_from\_directory command we will access the data in the train path and test path where, images are located.

Moreover, label should be given to the classes, such as

class labels=['Angry’, ‘Disgust', 'Fear', 'Happy', 'Neutral', 'Sad', 'Surprise']

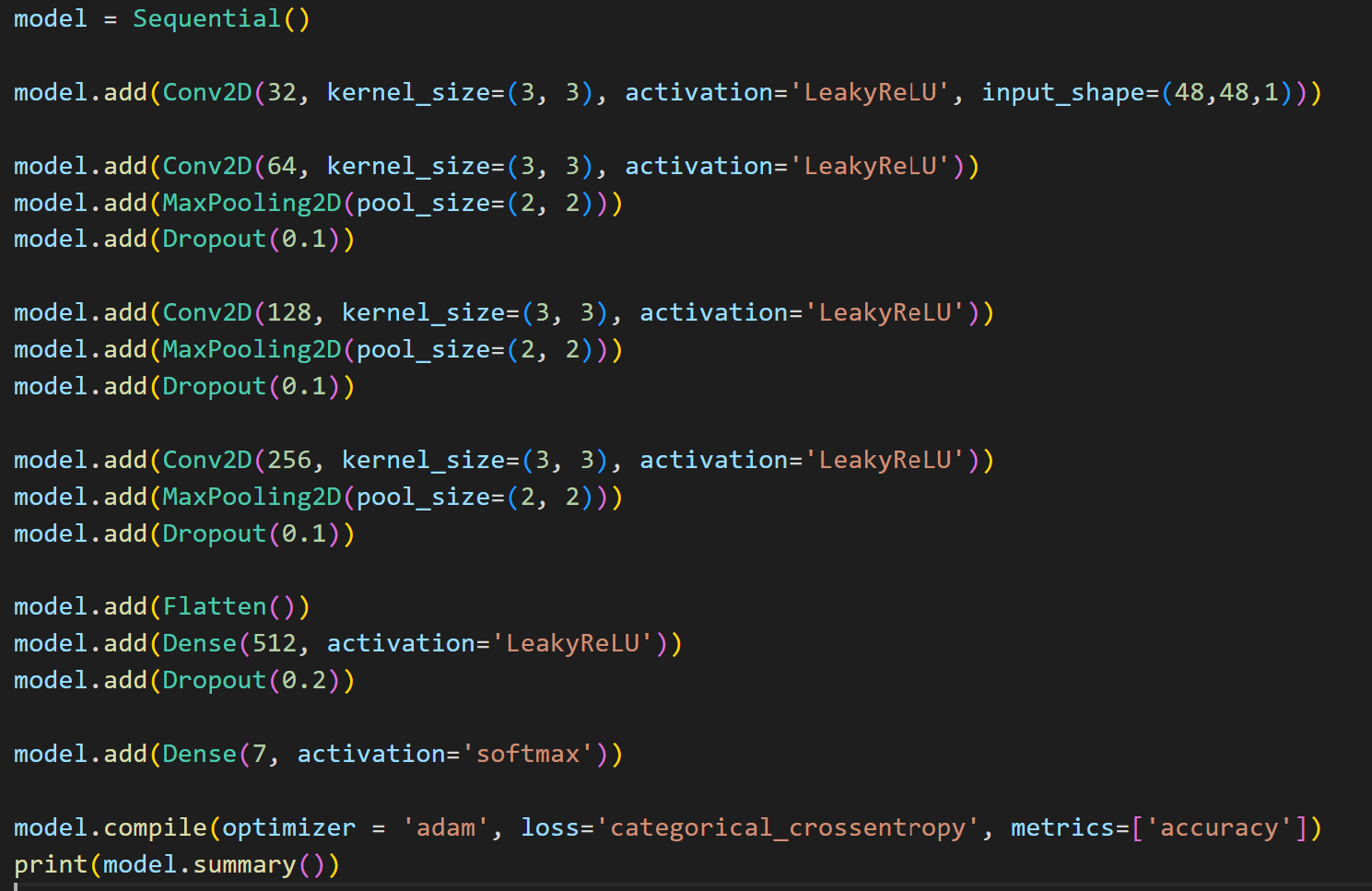
We will be creating a model using different layers and activation functions and optimizers.

Here, the input layer consists of Conv2D with activation function of LeakyReLu with the kernel size of 3 X 3 input size 48 X 48 with 1 channel because it is a grayscale image.

Layer 2 consists of Conv2D with activation function of LeakyReLu with a kernel size of 3 X 3 and MaxPooling layer of pool size 2 X 2 and dropout of (0.1). The filters of 3 and 4 layers have increased from 64 to 128 and 256 respectively and all other aspects are the same.

Layer 5 consists of a flatten layer with Dense layers of filter 512 and dropout of 0.2.

The final layer is a dense layer with 7 classes and activation function of SoftMax. In model compilation the optimizer used is Adam and loss is equal to categorical Crossentropy with metrics as accuracy.



Thirdly, make sure to select the device on which the model will be implemented, either CUDA or CPU.

We will use Early stopping and model checkpoint to increase the capabilities of our code. Early stopping is used when a monitored metric no longer improves, stopping training. Assuming the goal of training is to reduce loss. The metric to be monitored would be 'loss,' and the mode would be in.' A model.fit() training loop will check at the end of each epoch to see if the loss is still decreasing, taking into account the min delta and patience if applicable. When it is discovered that it is no longer decreasing, model.stop training is set to True, and the training is terminated. The quantity to be monitored must be present in the logs dict. To do so, pass the loss or metrics to the model. compile [53].

At some frequency, call back to save the Keras model or model weights by using Modelcheckpoint. It is a call back which is used in conjunction with model.fit() training to save a model or weights (in a checkpoint file) at some interval so that the model or weights can be loaded later to continue training from the saved state[54].

Following this we will fit our model by using the number of epochs required here. An epoch is defined as the total number of iterations of all the training data in one cycle for training the machine learning model when all of the training data is used at once [55].

Fourthly, specify the model path and also specify the path of the folder “model " and "results."

After getting the model we will test it, for this we will use commands and different libraries such as cv2, numpy and haarcascade frontal face default.xml to detect faces from the frontal view. Find haar cascade to draw bounding boxes around face. Sub face image is used to zoom into the particular picture to extract features from it and neglect surrounding objects. Then it will create a rectangular frame around the face of the person after predicting the emotion and display the emotion by using cv2. By using the iamshow attribute in cv2 it will show the photo.

Finally, run the "test.py" after the adjustments mentioned above, and then you will be able to see the emotion detection in the results folder.

**CHAPTER 9 - RESULT**

After completing the whole model, and executing the code it leaves us with result. Here the model was able to predict the emotion accurately but there are curtain limits up to what a model can do. Not every model can be called a good model neither a model which gives 90 percent of accuracy it, wholly depends on the project, and test-based results prove an understanding whole project in different format. So, when coming to our model we are did not got the whole accuracy even though it capable of predicting different emotions from the given input Image. As an example, we are going to give an input of image which is around 225\*255 which has different emotions binder within the picture and we will look at the output.



The above images show us the detailed emotions of a single person which were detected using a model. Comparing the two photos we can say that the model is capable of predicting different emotions in humans and was able to extract features from the data set and embed them in the model.

**CHAPTER 10 - CONCLUSION**

We have introduced an emotion detection model that accomplishes the preferred perceptual quality and utilized an original design containing a few layers. Likewise, important methods, including detection of human face and expressions, are utilized to work with the preparation of the proposed profound model.

The relativistic Model that we implemented as the discriminator differentiates between more realistic images and helps the generator to harvest more details in the image.

Also, we have upgraded the perceptual loss by utilizing the highlights before activation, which offer more significant management and hence re-establish more saturation and realistic surfaces.

A similar model can be built for various other purposes like converting black and white images into colorized images. In much detail and can be highly reliable once it gets high enough accuracy.

In this research, we propose a facial expression detection approach based on a CNN model that effectively extracts face features. In comparison to conventional approaches, the proposed approach can learn pattern features automatically and eliminate the incompleteness caused by artificial design aspects. Deep learning algorithms were used in a study to classify facial emotions over static or a series of image data.

This is a complex subject that has been handled multiple times using various strategies. While this experiment concentrated on keras and CNN yielded positive outcomes. This project achieves the desired output, which is that it detects a face from a given input within a border and detects the emotion of the face, displaying the detected output as happy, sad, neutral, angry, and surprised with a 60.14% accuracy. In conclusion, we were able to build a functional architecture for creating emotion detection with the increase in accuracy of the model and learned about machine learning, Research about data aspects.

**REFERENCES’**

1)springer. “Perceptual Losses for Real-Time Style Transfer and Super-Resolution.” *springer*, Justin Johnson, Alexandre Alahi & Li Fei-Fei, https://link.springer.com/chapter/10.1007/978-3-319-46475-6\_43.

2) https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7037130/

3) https://www.sciencedirect.com/science/article/pii/S1877050920318019

4)https://iq.opengenus.org/different-types-of-cnn-models/

5) Khan, Reeshad & Sharif, Omar. (2017). A Literature Review on Emotion Recognition Using Various Methods.

6) Shiqing Zhang, Xiaohu Wang , Gang Zhang, Xiaoming Zhao; Multimodal Emotion Recognition Integrating Affective Speech with Facial Expression ; Institute of Image Processing and Pattern Recognition Taizhou University Taizhou 318000 CHINA, Hunan Institute of Technology Hengyang 421002 CHINA,Bay Area Compliance Labs. Corp. Shenzhen 518000 CHINA, 2014

7) Martin W¨ollmer, Angeliki Metallinou, Florian Eyben, Bj¨orn Schuller, Shrikanth Narayanan; ContextSensitive Multimodal Emotion Recognition from Speech and Facial Expression using Bidirectional LSTM Modeling; Institute for Human-Machine Communication, Technische Universit¨at M¨unchen, Germany Signal Analysis and Interpretation Lab (SAIL), University of Southern California, Los Angeles, CA, 2010.

8) Samira Ebrahimi, Vincent Michalski, Kishore Konda, Goethe Roland Memisevic, Christopher Pal― Recurrent Neural Networks for Emotion Recognition in Video‖,Kahou École Polytechnique de Montréal, Canada ; Universität Frankfurt, Germany; Université de Montréal, Montréal, Canada; 2015

9) Chalciya Elizabeth Rani P, H.P. Mohan Kumar (2021). Facial Emotion Detection Using Keras

10)https://www.analyticsinsight.net/what-is-ai-image-recognition-how-does-it-work-in-the-digital-world/

11)https://www.brookings.edu/research/how-artificial-intelligence-is-transforming-the-world/

12)“python.” *Welcome to Python.org*, https://www.python.org/. Accessed 9 May 2022.

13)“vscode.” *Visual Studio Code - Code Editing. Redefined*, https://code.visualstudio.com/. Accessed 9 May 2022.

14)https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-keras#:~:text=Keras%20is%20used%20for%20creating,%2C%20Yelp%2C%20Uber%2C%20etc.

15)https://keras.io/api/layers/

16)https://keras.io/api/optimizers/

17)https://www.educba.com/keras-imagedatagenerator/

18)Solegaonkar, Vikas. “Introduction to PyTorch. PyTorch is an open source machine… | by Vikas Solegaonkar.”*TowardsData Science*, https://towardsdatascience.com/introduction-to-py-torch-13189fb30cb3. Accessed 9 May 2022.

19) torch.nn.functional — PyTorch 1.11.0 documentation.” *PyTorch*, https://pytorch.org/docs/stable/nn.functional.html. Accessed 9 May 2022.

20) torch.nn — PyTorch 1.11.0 documentation.” *PyTorch*, https://pytorch.org/docs/stable/nn.html. Accessed 9 May 2022.S

21) os.path — Common pathname manipulations — Python 3.10.4 documentation.” *Python Docs*, https://docs.python.org/3/library/os.path.html. Accessed 9 May 2022.

22) https://www.geeksforgeeks.org/os-module-python-examples/

23) “Python Pandas Tutorial.” *Tutorialspoint*, https://www.tutorialspoint.com/python\_pandas/index.htm. Accessed 9 May 2022.

24)“An introduction to seaborn — seaborn 0.11.2 documentation.” *Seaborn*, https://seaborn.pydata.org/introduction.html. Accessed 9 May 2022.

25)“Matplotlib documentation — Matplotlib 3.5.2 documentation.” *Matplotlib*, https://matplotlib.org/stable/index.html. Accessed 9 May 2022.

26)“scikit-learn Tutorials — scikit-learn 1.0.2 documentation.” *Scikit-learn*, https://scikit-learn.org/stable/tutorial/index.html. Accessed 9 May 2022.

27)https://www.simplilearn.com/keras-vs-tensorflow-vs-pytorch-article#:~:text=TensorFlow%20is%20an%20open%2Dsourced,because%20it's%20built%2Din%20Python.

28)LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521:436–444

29)Ehteshami Bejnordi B, Veta M, Johannes van Diest P et al (2017) Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. JAMA 318:2199–2210

30)Lakhani P, Sundaram B (2017) Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. Radiology 284:574–582

31)Esteva A, Kuprel B, Novoa RA et al (2017) Dermatologist-level classification of skin cancer with deep neural networks. Nature 542:115–118

32)Gulshan V, Peng L, Coram M et al (2016) Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. JAMA 316:2402–2410

33)https://www.upgrad.com/blog/basic-cnn-architecture/#:~:text=other%20advanced%20tasks.-,What%20is%20the%20architecture%20of%20CNN%3F,the%20main%20responsibility%20for%20computation.

34)Yasaka K, Akai H, Abe O, Kiryu S (2018) Deep learning with convolutional neural network for differentiation of liver masses at dynamic contrast-enhanced CT: a preliminary study. Radiology 286:887–896

35) https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/

36)https://www.upgrad.com/blog/basic-cnn-architecture/#:~:text=other%20advanced%20tasks.-,What%20is%20the%20architecture%20of%20CNN%3F,the%20main%20responsibility%20for%20computation.

37)https://paperswithcode.com/method/leaky-relu

38)https://stackoverflow.com/questions/57751417/what-is-meant-by-sequential-model-in-keras#:~:text=on%20this%20post.-,The%20Sequential%20model%20is%20a%20linear%20stack%20of%20layers.,networks%20with%20some%20shared%20layers.

39)https://www.sciencedirect.com/topics/engineering/convolutional-layer

40)https://www.sciencedirect.com/topics/mathematics/pooling-layer#:~:text=The%20purpose%20of%20the%20pooling,Correction%20(Second%20Edition)%2C%202021

41)https://towardsdatascience.com/convolutional-layers-vs-fully-connected-layers-364f05ab460b?gi=ddcc70cfd2a

42)https://www.tensorflow.org/api\_docs/python/tf/keras/layers/Dropout#:~:text=The%20Dropout%20layer%20randomly%20sets,over%20all%20inputs%20is%20unchanged.

43)https://www.v7labs.com/blog/neural-networks-activation-functions#:~:text=An%20Activation%20Function%20decides%20whether,prediction%20using%20simpler%20mathematical%20operations.

44)https://www.geeksforgeeks.org/tensorflow-js-tf-layers-globalaveragepooling2d-function/

45)https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-3-flattening

46)https://keras.io/api/layers/pooling\_layers/max\_pooling2d/

47)https://www.kaggle.com/datasets/msambare/fer2013

48)https://www.scribbr.com/methodology/data-collection/#:~:text=about%20data%20collection-,Step%201%3A%20Define%20the%20aim%20of%20your%20research,and%20why%20does%20it%20matter%3F

49)https://www.researchgate.net/figure/Sample-images-from-the-Fer-2013-dataset-Images-in-the-same-column-show-identical\_fig1\_329422139

50)https://www.tensorflow.org/api\_docs/python/tf/keras/preprocessing/image/ImageDataGenerator

51)https://www.tensorflow.org/api\_docs/python/tf/keras/optimizers/Adam#:~:text=Used%20in%20the%20notebooks,-Used%20in%20the&text=Adam%20optimization%20is%20a%20stochastic,order%20and%20second%2Dorder%20moments.

52) https://www.educba.com/keras-imagedatagenerator/

53)https://keras.io/api/callbacks/early\_stopping/

54)https://keras.io/api/callbacks/model\_checkpoint/

55)https://www.simplilearn.com/tutorials/machine-learning-tutorial/what-is-epoch-in-machine-learning#:~:text=An%20epoch%20is%20when%20all,dataset%20takes%20around%20an%20algorithm.

result.

56)https://web.stanford.edu/class/ee368/Project\_Autumn\_1617/Reports/report\_pao.pdf-research

57)https://docs.lib.purdue.edu/cgi/viewcontent.cgi?article=1849&context=purc – CNN