DETECTION OF STRESS IN IT EMPLOYEES USING DEEP LEARNING

A Thesis Submitted
in Partial Fulfilment of the Requirements
for the Degree Of

MASTER OF TECHNOLOGY
in
Information Systems
by

Aditya Kanwar
(Roll No. 2K22/ISY/03)

Under the Supervision of
Dr. Varsha Sisaudia
Assistant Professor, Delhi Technological University

Department of Information Technology

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi – 110042, India

May, 2024

ACKNOWLEDGEMENTS

I am thankful to my supervisor Dr Varsha Sisaudia Assistant Professor in the Department of Information Technology at Delhi Technological University, Delhi and all the department's faculty members. They all assisted me whenever I needed any help. This work would not be possible without their direction and assistance. I am also grateful to the IT department for providing the various resources required to complete this work.

ADITYA KANWAR

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

CANDIDATE'S DECLARATION

I Aditya Kanwar hereby certify that the work which is being presented in the thesis entitled "Detection of Stress in IT Employees using Deep Learning" in partial fulfilment of the requirements for the award of the Degree of Master of Technology, submitted in the Department of Information Technology, Delhi Technological University is an authentic record of my own work carried out during the period from 2022 to 2024 under the supervision of Dr. Varsha Sisaudia.

The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other institute.

Candidate's Signature

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

CERTIFICATE BY THE SUPERVISOR

Certified that **Aditya Kanwar** (2K22/ISY/03) has carried out their search work presented in this thesis entitled "**Detection of Stress in IT Employees using Deep Learning**" for the award of **Master of Technology** from Department of Information Technology, Delhi Technological University, Delhi, under my supervision. The thesis embodies results of original work, and studies are carried out by the student himself and the contents of the thesis do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Signature

(Dr. Varsha Sisaudia)

(Assistant Professor)

(Address)

Date:

Detection of Stress in IT Employees using Deep Learning

Aditya Kanwar

ABSTRACT

In the contemporary era, characterized by advanced technological devices, a pervasive sense of stress is increasingly affecting individuals. Despite the abundance of material wealth, people often find themselves discontented. Stress, defined as a feeling of pressure, can manifest in mental, emotional, or physical forms. It is crucial to establish effective stress management systems to gauge and address stress levels that disrupt our socio-economic lifestyles. According to the World Health Organization (WHO), one out of every four individuals grapples with the mental health challenge of stress.The stress experienced by individuals can lead to both mental and socioeconomic challenges, including decreased concentration at work, strained relationships with colleagues, feelings of despair, and, in extreme cases, even suicide. To address this, it is essential to offer counseling support to individuals facing stress, aiding them in effectively managing their emotional burden. While it's unrealistic to eliminate stress entirely, adopting preventive measures can play a crucial role in its effective management.Only individuals with medical and physiological expertise are currently able to evaluate if someone is experiencing depression or stress. One well-established method for detecting stress involves using questionnaires. The main objective of our approach is to employ advanced deep learning and image processing techniques to recognize indicators of stress in IT professionals.This technology represents an enhanced iteration of prior stress detection technologies, distinguishing itself by incorporating considerations for employee emotions and real-time detection. In contrast to its predecessors, this system integrates both periodic and immediate detection of employee emotions.The identification of stress through automated means reduces the likelihood of health problems and enhances the well-being of both the IT employee and the organization. Understanding the emotional state of IT employees enables the company to offer appropriate support, leading to improved performance. The suggested system model, constructed with CNN Model Architecture demonstrates an accuracy of 98.45% and with Inception-V3 demonstrates an accuracy of 70%.

TABLE OF CONTENTS

Catalog

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 General

In the fast-paced realm of Information Technology, the mental health of employees is crucial for sustained productivity and organizational success. This report investigates a novel approach—utilizing machine learning, specifically the Convolutional Neural Network (CNN) Model Architecture, for real-time stress detection in IT professionals.

We'll delve into the significance of addressing stress in the IT workforce and explore the benefits of leveraging machine learning to identify and mitigate stressors.Furthermore, We'll assess how well the suggested system model performs highlighting its correctness during the verification and training stages.This dissertation serves as a concise exploration of an innovative solution to workplace stress, offering insights for organizations seeking to prioritize the well-being of their IT employees through the integration of machine learning technologies.

1.2 Objective

The objectives of this dissertation are as follows:

- Assess the significance of addressing stress in the Information Technology industry and its impact on individual well-being and organizational performance.
- Explore the application of advanced machine learning techniques for real-time stress detection, emphasizing the proactive and tailored support it can offer to IT professionals.
- Analyze the performance metrics of the proposed CNN-based stress detection system, including accuracy during both training and validation phase.
- Provide insights and recommendations for organizations seeking innovative solutions to prioritize the mental health.

1.3 Deep Learning

Computational models consisting of several processing layers can acquire representations of data with various degrees of abstraction through deep learning. In several fields, including discovering medicines and biology, these techniques have significantly raised the bar for state-of-the-art performance, including audio recognition, optical object identification, and object.

By applying the back propagation technique to suggest changes to a machine's own variables which are employed to determine the representation in each layer based on the depiction in the layer preceding it, deep learning uncovers intricate patterns inside massive data sets.While recurrent networks have brought insight into linear information such as literature and audio, deep convolutional nets have made significant advances in the processing of pictures, video, speech, and audio.

Many elements of today's society are fueled by machine learning technology, including web searches, Facebook and Twitter filtering of posts, e-commerce website suggestions, and the growing common use of machine learning in everyday products like smartphones and cameras. Machine learning algorithms are used to recognize subjects in photos, interpret voice into written language, connect goods, postings, and headlines to consumer preferences, and select outcomes of searches that are pertinent to their queries. These applications are using a class of methods known as deep learning more and more. The raw natural data could only be processed to a limited extent by traditional machine-learning methods.For many years, building a system for pattern identification or machine learning required meticulous engineering and deep expertise in the field to create a feature extractor that converted the original information into an appropriate internal representation or feature vector (such as the pixel value subsystem, which is frequently a classifier and could detect or classify patterns in the inputs of an image).

Deep learning techniques are model-learning techniques that incorporate many layers of representation. They achieve this by building straightforward but dynamic modules, each of which converts the raw input's representation at a particular level into a more advanced, somewhat broader representation.Learning of highly complicated equations is possible if adequate of these transformations are coupled. Greater levels of representations emphasize characteristics of the input that are vital to excluding and suppress modifications that are unimportant for classification tasks.The acquired characteristics in the initial phase of representation, for instance, generally reflect the existence or lack of borders at specific perspectives and locations within an image, which can be expressed as a collection of pixel counts.In general, the subsequent tier recognizes patterns according to certain boundary structures, even when there are slight differences in the border placements.Later layers would determine things alongside one another of these pieces after the third layer assembled patterns into bigger combinations that correlated to parts of recognizable objects.The essential component of deep learning is because the feature levels are not created by mortal programmers; rather, a universal algorithm for learning is used to learn them from data. Significant progress is being made by deep learning in addressing issues that have eluded the artificial intelligence finest efforts for an extended period.This appears to be highly effective in identifying complex structures in data with multiple dimensions, making it useful in a variety of scientific, commercial, and federal fields.Apart from surpassing previous milestones in photo and detection of speech,it has also performed better than alternative machine learning techniques.

in analyzing particle accelerator data, reenacting neural networks, forecasting the impact of non-coding DNA mutations on gene expression and illness, and predicting the activity of possible drug molecules.

1.4 Convolutional Neural Networks

Convolutional neural networks are structured to handle input in the form of several arrays. For instance, they may analyze a color picture made up of 3 2D arrays with pixel values in each of the three color bands.Several arrays are used to represent many different information paradigms. 1D signals and sequences of events, such as speech; 2D pictures or audio spectroscopy images; and 3D video or spatial images.Convolutional neural networks use the characteristics of natural signals through 4 main concepts: multiple layer usage, pooled weights, pooling, and local links.

A typical Convolutional neural network has a topology that is divided into many phases. Layers of convolution and layers of pooling make up the initial phases of the system.CNN layer components are arranged in feature maps, where individual component is linked to local patches in the preceding layer's feature maps by a filter bank, which is a collection of weights.

Put otherwise, the logic behind units at various places having the identical weights and recognizing the identical design in various regions of the array is that if a motif is present in one portion of the picture, it may appear anywhere.The term comes from the fact that a feature map's sorting function is equivalent to a discrete convolution in mathematics. While the CNN layer is responsible for identifying local connections between features from the preceding layer, the pooling layer unifies semantically related elements into a single layer.Coarse-graining individual feature's spot can help recognize motifs more reliably since the spatial arrangement of the features that make up a motif might fluctuate.

In a single feature map (or several feature maps), a standard pooling module calculates the highest value of a local patch of units. By accepting feedback from patches that have been moved by many rows or columns, neighboring pooling units lower the representation's dimensionality and produce an invariance to slight deformities and changes.More convolutional and fully-connected layers come after two or three stacked levels of convolution, non-linearity, and pooling.It is as easy to backpropagate gradients via a CNN as it is using a conventional deep network, enabling training of all the weightings throughout the filter banks. The notion that numerous spontaneous signals are compositional hierarchies—in which higher-level features are produced by assembling smaller-scale ones—is something that deep neural networks take use of. Local boundary permutations create motifs in pictures, which then put together to make components, which then form things. Equivalent hierarchies between sounds, phones, phonetic sounds, syllables, words, and sentences may be seen in both speech and writing.When items in the preceding layer change in location and style, the pooling permits relatively minimal variation in representations. CNN's convolutional and pooling layers are strongly influenced by the traditional ideas of straightforward and intricate cells in optical neurology, and the whole structure is similar to the ventral routes of the visual cortex's LGN–V1–V2–V4–IT hierarchy. Beginning via time-delay neural networks for detecting speech and text processing in the early nineties, CNN have found many uses in several fields. The ConvNet that was trained in tandem with a probabilistic framework to impose linguistic restrictions was utilized by the document reading system.Over ten percent of all paychecks written in American banks were being read by this technology by the end of the nineties. Microsoft later implemented many ConvNet-based optical-character detection and handwriting detection systems.Earlier in the nineties, ConvNets were also tested for recognizing facial features and finding objects in natural pictures, including limbs and faces.

1.5 Image understanding with deep convolutional networks

ConvNets have been successfully used for objects and area recognition, segmentation, and detection in pictures since the beginning of the twenty-first century. These included objectives like identifying roadway markings, splitting organic pictures, especially for connectomics, and identifying faces, text, pedestrians, and humanoids in natural photographs—all of which required a fair amount of labeled data.

Recognition of facial features is one of ConvNets' most notable current applications.It is important to have the capacity to categorize images down to the pixel level because it will be useful in the development of autonomous mobility robots and autonomous vehicles.Companies such as Tesla is using such ConvNet-based methods in their upcoming vision systems for cars. Speech recognition and natural language understanding are two further applications that are becoming more and more popular. Even with these achievements, ConvNets were not given much attention by the mainstream machine learning and computer vision fields until the 2012 ImageNet contest.Deep convolutional networks produced incredible outcomes, including nearly half the rate of error of the best competitive solution, when used on a data set consisting of approximately million of internet pictures with 1,000 different categories.

This was made possible by maximizing the potential of GPUs, ReLUs, a novel normalization method termed dropout, and methods for producing additional training instances by altering the ones that already existed. This achievement has revolutionized computer vision ConvNets have become the standard method for nearly all tasks involving recognition and detection, and they even rival employee productivity in many cases.

ConvNets and recurrent net modules are used in a latest amazing presentation to generate picture descriptions. Current ConvNet topologies have billions of connections between units, countless millions of weights, and ten to twenty layers of ReLUs.A few years prior, training such massive networks may have taken days, but thanks to advancements in hardware, software, and algorithm parallelization, training periods are now a matter of hours. High-performance GPUs and TPUs, alongside optimized frameworks like TensorFlow and PyTorch, have significantly accelerated the training process. Moreover, innovative techniques in distributed computing and improved model architectures contribute to reducing the computational load, making rapid development and iteration feasible. An increasing amount of new companies to launch R&D initiatives and roll out ConvNet-based image interpretation services and goods. ConvNets can be efficiently implemented on chips or gates that are field-programmable with ease. This allows for the customization of neural network architectures to suit specific tasks, optimizing performance and energy consumption. Furthermore, field-programmable gate arrays (FPGAs) offer the flexibility to reconfigure the hardware as the algorithms evolve, ensuring adaptability and longevity in rapidly changing technological landscapes.

According to deep-learning theory, deep nets outperform traditional learning algorithms that do not make use of distributed representations in 2 distinct ways.Each of these benefits stem from composition's power and require that the distribution that generates the data has the proper componential structure.In the beginning standardization of previously unknown combinations of acquired feature values beyond those seen during training is made feasible by learning distributed representations (for instance, two n combinations are conceivable with n binary features).subsequently, creating layers of representation within a deep net may yield an additional exponential benefit (explicit in the depth).A multilayer neural network's hidden layers learn to represent the inputs of the network in a way that makes target outputs simpler to anticipate.An excellent method to demonstrate this is to train a multilayer neural network to predict the next word in a series depending on the local context of preceding words. The network is shown with every phrase in its proper context as a 1-of-N vector, meaning one particular element gets an intrinsic value of 1 and the other components have values of 0.Every word generates a distinctive sequence of activation events or word vectors, in the first layer. The remaining layers of a language model's network learn how to translate the supplied vectors of words into the resultant vector of words for the anticipated next word, which may be utilized to forecast the likelihood that any vocabulary word will occur as the following word.The word vectors with several active components, each of which may be interpreted as a distinct word feature, are taught to the network.This was initially shown to function when acquiring distributed representations for symbols.

CHAPTER 2
LITERATURE SURVEY

Widenti et. al. [1] proposed that tension manifests as a blend of physiological and psychological strain, impacting various aspects of an individual's daily life. This amalgamation of stressors can give rise to adverse emotions or sentiments that are in stark contrast to one's genuine desires, thereby jeopardizing emotional well-being. Moreover, stress has the potential to distort an individual's perception of reality, hinder problem-solving capabilities, and impair logical thinking processes. Stressors, the catalysts of stress, are broadly categorized into internal and external stressors. Internal stressors stem from within the individual, such as health conditions or hormonal fluctuations, while external stressors originate from external factors or surrounding conditions, such as the loss of a loved one or financial difficulties.

Building upon the aforementioned issue, they have conceived a novel concept aimed at [1] assessing an individual's stress level through the evaluation of parameters like hypertension, pulse rate, and galvanic skin response (GSR), [2] analyzing the data obtained from these parameters, and subsequently displaying the corresponding stress level on an LCD screen. These stress levels are categorized into four stages, commencing from the lowest, which represents relaxation, and advancing through serene, anxious, and ultimately agitated states. [3] To mitigate stress and foster a more tranquil mood, they propose the utilization of infrared radiation emitted from a device worn around the user's neck. This proactive approach not only enhances self-awareness but also empowers individuals to take control of their emotional well-being.

Moreover, this concept envisions the integration of wearable devices equipped with biofeedback sensors, allowing users to monitor their stress levels throughout the day and make informed decisions to mitigate stressors. By fostering a holistic approach to stress management, this solution aims to empower individuals to lead healthier, more balanced lives. Additionally, these wearable devices can be synchronized with mobile applications, providing users with real-time feedback and personalized recommendations to manage stress more effectively. By incorporating relaxation techniques, such as guided breathing exercises and mindfulness practices, the system supports users in developing long-term coping strategies.

Ravinder et. al. [2] were one of the first to introduce that psychological strain is a big problem, particularly in young people. The age that was formerly thought to be the most easygoing is currently under a lot of stress. These days, a rise in stress causes a number of issues, including feelings of hopelessness, cardiac arrest, and seizures. Within the following paper, they compute students' psychological anxiety levels both during their internet use and a week before to the test. The goal is to examine stress in undergraduates at various stages of their lives[4].The often-unnoticed impact that stress from exams or recruiting has on students.They plan to examine the impact of various factors on students' mental well-being and explore the connection between stress levels and internet usage. This study utilizes a dataset from Jaypee Institute of Information Technology, encompassing information from 206 students.They used four classifying methods: SVM, Random Forest, Naïve Bayes, and Linear Regression.They evaluated each algorithm's performance using three parameters: sensitivity, specificity, and accuracy. To increase the precision and dependability of our results, 10-Fold Cross-Validation was used. With an accuracy of 85.71%, the Support Vector Machine stood out as the most accurate model.

Sharma et.al. [3] proposed that the World Health Organization highlights stress as a major contemporary issue impacting the well-being of individuals, affecting both their physical and mental health. Numerous conventional approaches exist for detecting stress, including various techniques for monitoring the human brain to analyze behavior[7]. It is worth noting that current research primarily focuses on stress detection methods rather than technological interventions for stress reduction[10].This study suggests an innovative approach to identify stress through EEG signals and alleviate it by incorporating interventions into the system. Employing the k-means clustering technique, the research gauges perceived stress, categorizing subjects and estimating stress levels. The proposed method holds potential for creating products aimed at reducing human stress. Successfully implementing and advancing this research is anticipated to streamline the process and minimize human effort in finding optimal stress management solutions.

G.Giannakakis et. al. [4] Using video recordings of people's facial expressions, this study establishes a comprehensive framework for identifying and analyzing emotional states associated with stress and anxiety. By subjecting participants to a diverse range of internal and external stress-inducing stimuli, a meticulously designed experimental protocol was developed to systematically elicit consistent fluctuations in affective states, encompassing neutral, calm, and stressed/anxious conditions[8]. The focal point of the analysis revolves around involuntary and partially voluntary facial cues, aiming to provide a more objective assessment of emotional representation.

In this investigation, various aspects of facial expressions were scrutinized, including ocular movement, oral motions, movement of the head parameters, and cardiovascular rate, which were quantitatively measured using a camera-based photoplethysmograph. Through the application of feature selection techniques, the most discerning characteristics were identified, and classification methodologies were employed to distinguish between neutral and stressed/anxious states, juxtaposed with a calm condition during each experimental phase.

Furthermore, a novel rating translation approach was proposed, leveraging self-reported data to elucidate the intricate relationship between facial attributes and an individual's reported levels of anxiety or stress. The findings of this study unveiled that specific visual indicators, such as movement of the head, mouth, and gaze, along with camera-based cardiac rhythms, exhibited remarkable precision and efficacy in discriminating between stress and anxiety.

The exploration of the intricate interplay between the knowledge conveyed through a person's facial expressions and their underlying emotional state remains a topic of significant interest. As per Darwin's observations, regardless of ethnicity or cultural background, most emotions manifest similarly on the face[12]. Recent studies have further underscored the importance of facial expressions and cues in the identification and categorization of distress. Notable areas of interest include ocular features (e.g., gaze distribution, blinking pace, pupil diameter fluctuation), mouth movements (e.g., mouth movement, lip distortions), as well as broader head behaviors (e.g., head motions, head velocity), which collectively serve as primary indicators of anxiety on the face. Additionally, eyelid trembling, cosmetic pallor, and a pinched expression are among the other facial manifestations associated with anxiousness, further enriching our understanding of facial cues related to emotional states

Nisha et. al. [5] have examinedDistress is an undesirable emotional condition that people encounter in everyday settings, such as spending a lot of time in front of a computer. Because they spend so much time on computers[16], they have become a part of our daily lives, and as a result, we are more susceptible to the positive and negative aspects they bring.Even if it is impossible to totally avoid using laptops for work, one may at least limit their use when they see signs of stress[7]. Taking regular breaks and practicing good posture can also help alleviate the strain caused by prolonged laptop use. It is vital for an individual's safety to keep an eye on their mental state while they are spending extended amounts of time in front of a computer[11]. This technique uses real-time, non-intrusive video recording to analyze a subject's facial expressions to determine their emotional state.Every video frame has a distinct emotion that we are able to identify, and they determine the stress level throughout the course of the film in successive hours[18].They use a method that lets us train a model and examine variations in feature prediction. Theano is a Python framework designed to enhance the speed at which the linear regression model, utilized as a deep learning method in this case, executes and develops.Based on data from the generalized model of every age group, the experimental findings demonstrate that the designed system performs effectively.

CHAPTER 3
DATASET

The Kaggle website is the source of the dataset.The link of the dataset is given below.

Kaggle Link: https://www.kaggle.com/datasets/jayaprakashpondy/emotion-dataset

This dataset is widely used resource for training and evaluating facial expression recognition models. Some details about the dataset:

1. Emotion Categories:

The dataset contains images categorized into seven different emotions:

0: Angry-6802 images (14.04%)

1: Disgust-872 images(1.80%)

2: Fear-6412 images(13.24%)

3: Happy-11876 images(24.52%)

4: Sad-8764 images(18.1%)

5: Surprise-8274 images(17.08%)

6: Neutral-5418 images(11.19%)

2. Image Characteristics:

- Images have a resolution of 48x48 pixels.
- The images have undergone automatic registration to ensure that each face is approximately centered and occupies a similar amount of space.

3. Dataset Size:

- The training set consists of 48,418 examples, and the public test set comprises 9,908 examples.

4. Challenges:

- Recognizing facial expressions is a challenging task due to variations in lighting conditions, facial poses, and individual differences in expressions.

Figure 3.1: Dataset Sample Images

CHAPTER 4
PRELIMINARY

4.1 Convolutional Neural Network

One kind of artificial neural network that's frequently utilized for image and video processing applications is the convolutional neural network (CNN). CNNs use a sequence of convolutional layers to automatically extract and learn significant features from raw input data, such pixels in an image.

A CNN processes an input picture via many convolutional layers, each of which extracts and filters the characteristics that are most pertinent to the input image. After that, a number of pooling layers are applied to the collected features, reducing the spatial dimensions of the feature maps, all the while maintaining the salient characteristics.

The final output is passed via one or more fully connected layers, which use the retrieved features to perform classification or regression tasks, after a number of convolutional and pooling layers. Using gradient-based optimization techniques like backpropagation to minimize a loss function, the CNN learns the ideal weights and biases for each layer during training.

On a variety of computer vision tasks, such as picture classification, object identification, segmentation, and image production, CNNs have demonstrated state-of-the-art performance. Additionally, they are employed in fields like natural language processing and speech recognition.

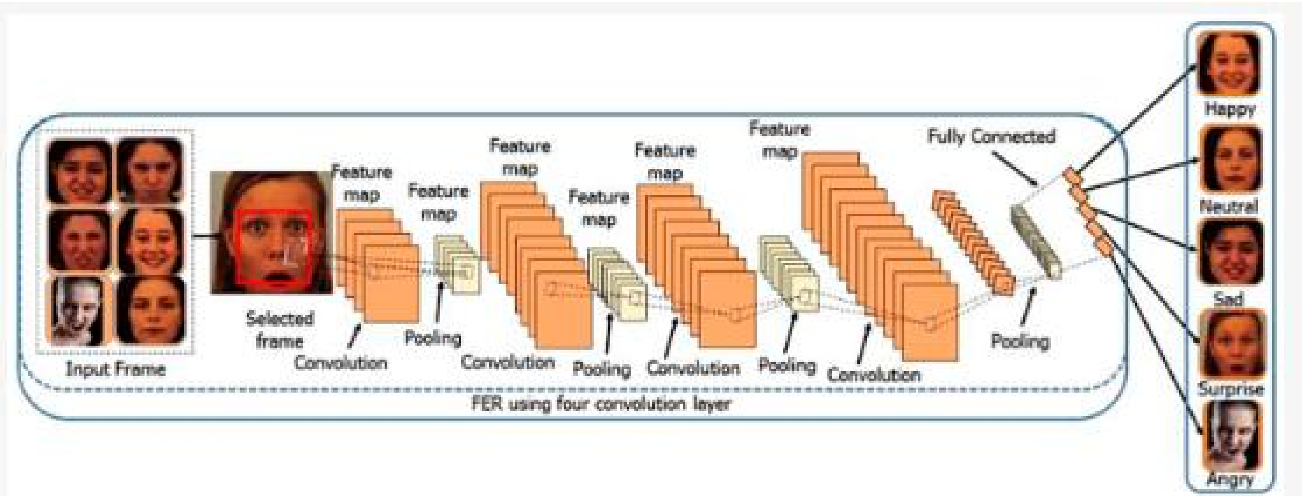

Figure 4.1: CNN Architecture

4.2 InceptionNet

Google unveiled InceptionNet, sometimes referred to as Inception, a convolutional neural network architecture, in 2014. By employing a number of smaller convolutional filters in simultaneously, it was created to solve the issue of classic convolutional neural networks' wasteful use of processing resources.

A module known as a "Inception module"—consisting of many simultaneous convolutional filters of varying sizes—is used in the InceptionNet architecture. As a result, the network may concurrently record characteristics at various sizes and orientations. Subsequently, the filter outputs are combined and sent into the subsequent layer. This procedure is iterated numerous times in order to produce a deep neural network that has various Inception modules

InceptionNet has demonstrated state-of-the-art performance on many benchmark datasets and has been used to a variety of computer vision applications, such as semantic segmentation, object identification, and picture classification. Inception-V2, Inception-V3, and Inception-ResNet are some of the later iterations of InceptionNet that have been expanded upon and enhanced, each adding new features to boost efficiency.

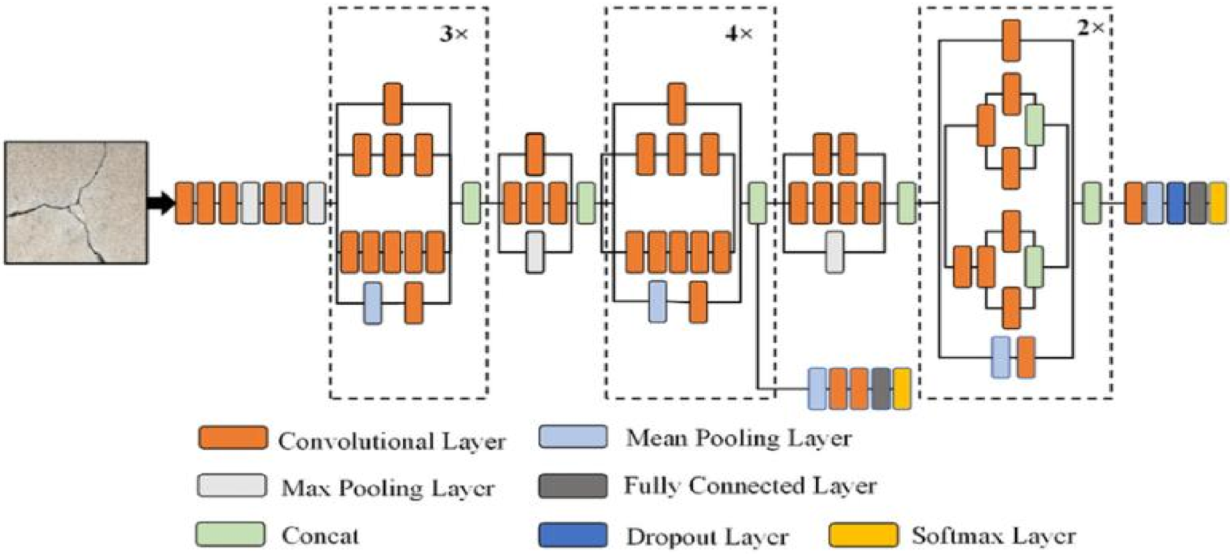

Figure 4.2: InceptionNet Architecture

.

Convolutional Neural Networks

- Recalling the terms (padding, stride, filter, etc.) used in convolutional neural networks

Building convolutional neural networks for the classification of multi-class images

Computer Vision

Few of the computer vision issues we'll be addressing are:

1. Object Identification
2. Transfer of neural style
3. Object Identification

One of the main problems with computer vision problems is the abundance of data available. Assume that a picture measures 68 by 68 by 3. At that point, the input feature dimension is 12,288. If we have larger photos (say, 720 X 720 X 3), this will be even larger. Now, a neural network's number of parameters will increase dramatically if we feed it an input this size.This will mean that more memory and computing power—which the majority of individuals unable to handle—will be required.

.Edge Detection Example

As it was observed in the prior piece, the initial layer within a neural network can identify edges in a picture. Higher levels are likely be able to identify the source of an individual object, while deeper layers could possibly be able to identify the source of an entire thing, like a person's face..The method of recognizing edges from an image will be the main focus of this section.Suppose we are given the below image:

Figure 4.3:Sample Image for edge detection

The picture includes multiple boundaries, both vertical and horizontal, as every observer may perceive.The first thing to do is to identify these limits.

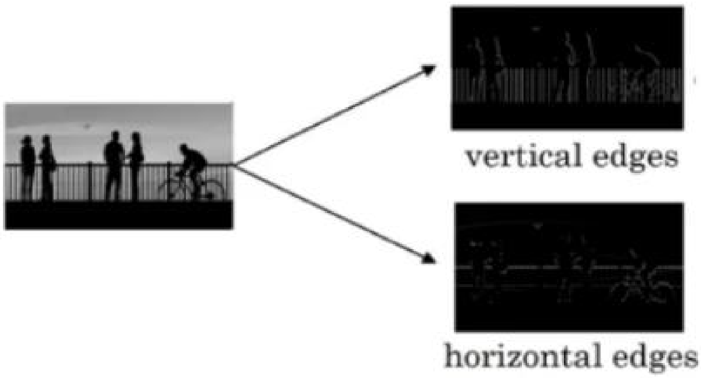

Figure 4.4: Vertical and Horizontal edges

However, how can we find these edges? To illustrate this, we'll utilize a six by six monochrome image:

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

Figure 4.5: 6X6 grayscale image

Next, we convolve this six by six matrix with a filter of 3 by 3 :

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

1	0	-1
1	0	-1
1	0	-1

Figure 4.6: Convolution of Image with Filter

Following the convolution, a four by four photograph will be obtained. The four by four matrix's first component will be identified as:

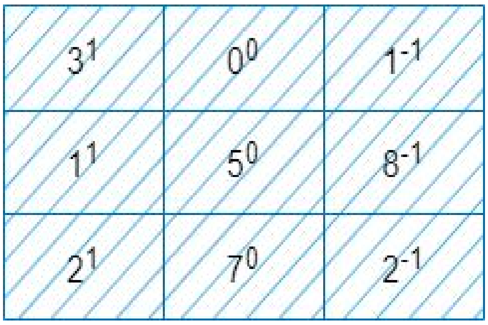

Figure 4.7: First element of Convoluted Image

Consequently, we multiply the first three by three matrix from the six by six picture by the filter. The aggregation of these element-wise product, or 3*1 + 0 + 1*-1 + 1*1 + 5*0 + 8*-1 + 2*1 + 7*0 + 2*-1 = -5, will now be the first element of the four by four result.To compute the following element of the four by four output and get the component-wise product total once more, we will shift our filter by one position to the right.

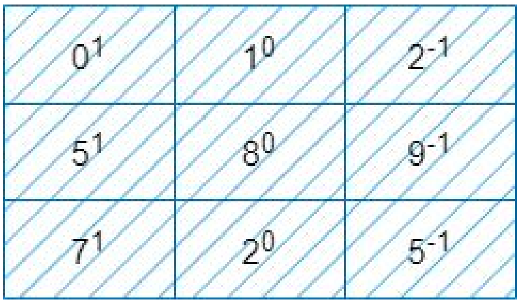

Figure 4.8: Second element of convoluted image

For instance, the next calculation will involve the matrix formed by the elements immediately to the right of the initial three-by-three segment. We repeat this process, sliding the filter over the entire image, ensuring each position is covered systematically. Likewise, we will convolve across the whole image to get an output that is 4 x 4, capturing localized features across the entire original image. This sliding and multiplying procedure is the fundamental operation in convolutional layers of neural networks, effectively reducing the dimensionality and extracting essential features from the image for further processing.

Lets take another example:

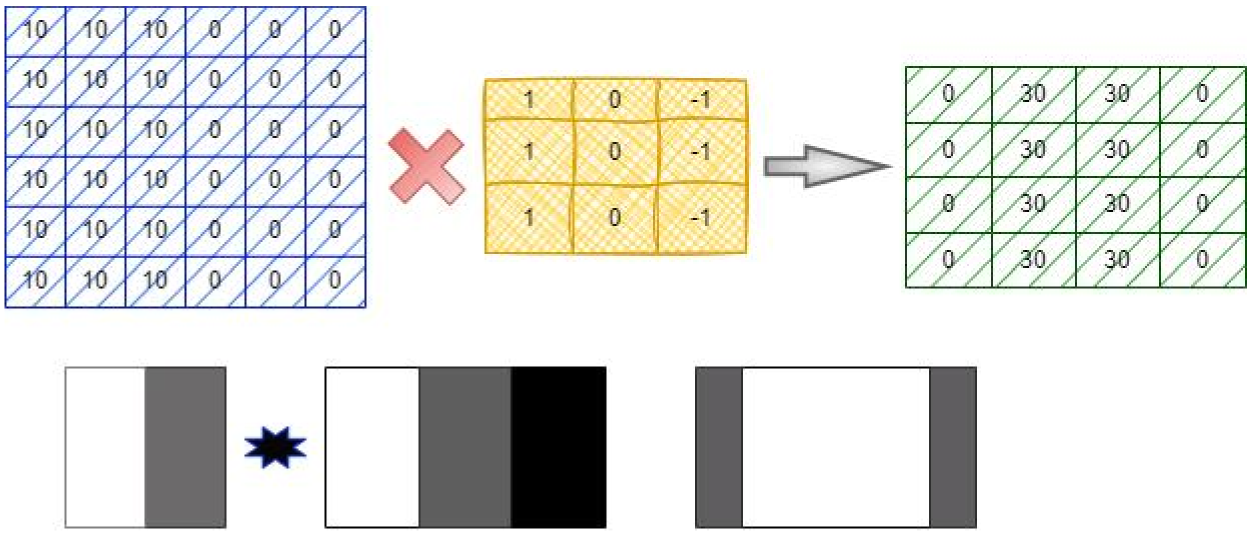

Figure 4.9: Convoluted Image Example

Greater values for pixels indicate the image's brighter regions, while lower pixel values indicate its darker regions. The vertical edge of a picture may be found in this way.

Convolutions Over Volume:

Assume we have a 3-D input image of the dimensions six by six by three in place of a 2-D image. How will convolution be applied to this picture? Rather of using a three by three, we will utilize a three by three by three. Let's examine an illustration:

- **Input:** 6 X 6 X 3
- **Filter:** 3 X 3 X 3

The dimensions above reflect the height, width, and channels in the input and filter. Recall that the number of channels in the input and filter should match. This produces an output of four by four. Let's look at it visually: because the input has three channels, the filtering will also have three channels. The final form after convolution is a four by four matrix. Consequently, the first element of the output is the total element-wise product of the first twenty-seven entries from the input and the first twenty-seven values from the filter, or nine values from each channel. After that, we convolve throughout the entire image.

We have the option to employ many filters in place of simply one. How are we going to do that? Assume that the image's vertical and horizontal borders are identified by the first and second filters, respectively. The output dimension will change when several filters are used. As a result, if we had used two filters, we would have obtained a four by four by two output rather than a four by four output like in the prior example. This means that each filter produces its own four by four output matrix, and these matrices are stacked along a new dimension, representing the depth of the output. This approach allows the model to extract different types of features from the input image simultaneously. For instance, while one filter might detect edges, another might capture textures or other patterns, enabling the model to learn more complex and abstract representations of the data.

Let's try to visualise this.

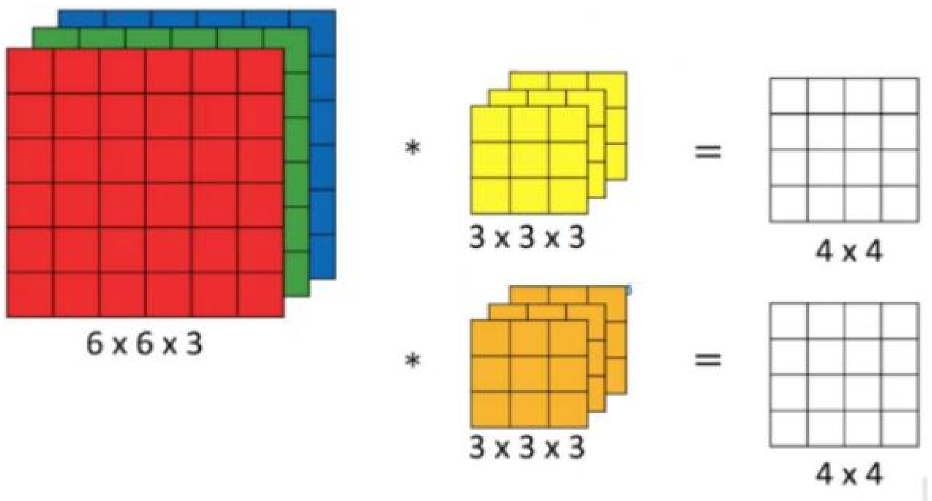

Figure 4.10: Convolution over a 3D Image

Generalised dimensions can be given as:

- **Input:** n X n X n_c
- **Output:** [(n+2p-f)/s+1] X [(n+2p-f)/s+1] X n_c'
- **Filter:** f X f X n_c
- **Padding:** p
- **Stride:** s

One Layer of Convolutional Network :

A convolutional network's top layer looks like this: we use a filter to convolve across the entire image, get an outcome, add a term that represents bias to the corresponding outputs, and then use an activation function to build activations.

$$z^{[1]} = w^{[1]} * a^{[0]} + b^{[1]}$$

$$a^{[1]} = g(z^{[1]})$$

In this instance, the weights w[1] are the filters (3 X 3 X 3) and the input (6 X 6 X 3) is a[0]. Layer 2 receives these activations from Layer 1 and so on as input.It is evident that the amount of variables in a convolutional neural network is independent of the picture's size. Essentially, the filter size determines it. Suppose we have 10 filters, arranged in a three-by-three configuration. What many of parameters is that layer going to have?

Because a total of 10 filters, the total parameters for that layer = 28*10 = 280

The characteristics are only dependent on the filter size, regardless of the size of the picture. Fantastic, isn't that right? Let's examine the convolution layer's breakdown of notations.

- f[l] = filter size
- p[l] = padding
- s[l] = stride
- n[c][l] = number of filters

Now, let's use what we've learned and look at an illustration of a convolutional network

Example of a Simple Convolutional Network

Let's examine the operation of a convolution neural network that includes a pooling and convolutional layer.Assume there is an input which is of size thirty two by thirty two by thirty two.

It is then convolved with ten filters of size three by three with an input picture, setting the stride to 1 and removing any padding. Our final product will be 37 X 37 X 10. After further convolving this result, we obtain the outcome that is seen above: 7 X 7 X 40. When all these numbers add up to 1960 (7 X 7 X 40), we finally unfurl them into a sizable vector before sending them to a classifier for prediction.

This is a convolutional network in miniature.

While creating a convolutional network, we may adjust a lot of hyperparameters. These consist of the quantity and size of filters, the chosen stride, padding, etc. Later in this post, we will examine each of these in further depth. Just remember that the picture size decreases and the number of channels often rises as we proceed further into the network.

In a convolutional network (ConvNet), there are basically three types of layer :

1. Convolution Layer
2. Pooling Layer
3. Fully Connected Layer

Pooling Layer :

Pooling layers are commonly used to reduce the number of inputs and speed up calculation.

A 4 X 4 matrix is depicted below:

1	3	2	1
2	9	1	1
1	3	2	3
5	6	1	2

Figure 4.11: A 4X4 matrix

The following matrix will yield a 2 X 2 output when max pooling is applied:

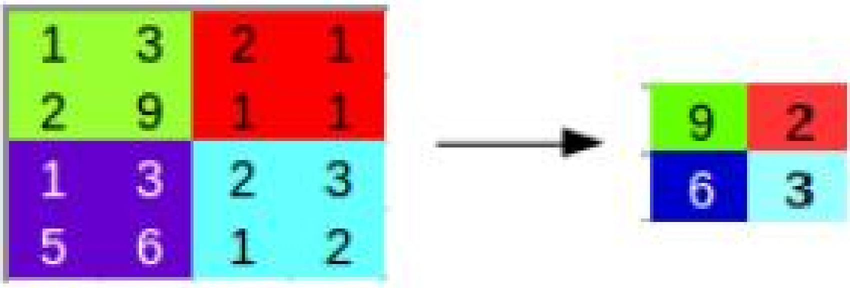

Figure 4.12: Max Pooled Output

In each subsequent two by two block, we take the largest number. This particular scenario, a filter having a size of two and a stride of two have been used. The pooling layer's parameters are as follows. In addition to max pooling, average pooling may be used in situations where, We take the average of the numbers rather than their maximum. In nutshell pooling layer mainly has these parameters:

1. Filter size
2. Stride
3. Max or average pooling

CHAPTER 5
IMPLEMENTATION

5.1 CNN Model

5.1.1 Importing the necessary libraries:

We plan to utilize the Python programming language for this task. Initially, we'll import essential libraries, including Keras for constructing the primary model, scikit-learn for dividing the data into training and testing sets, PIL for transforming images into numerical arrays, and additional libraries like pandas, numpy, matplotlib, and tensorflow.

5.1.2 Retrieving the images:

We'll collect both the images and their corresponding labels, followed by resizing the images to a uniform size of (48,48) to ensure consistent recognition. Finally, we'll transform the images into a numpy array.

5.1.3 Splitting the dataset:

Divide the dataset into test and train sets. There are 20% test and 80% train data.

5.1.4 CNN Architecture

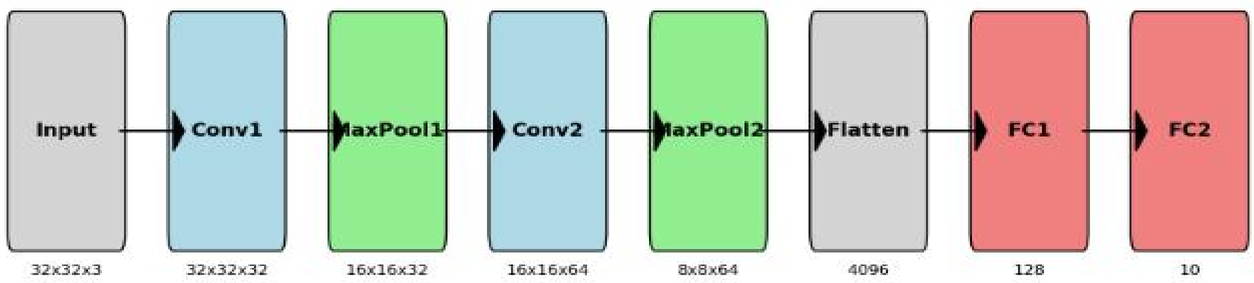

Figure 5.1: CNN Architecture

5.1.5 CNN Model Architecture

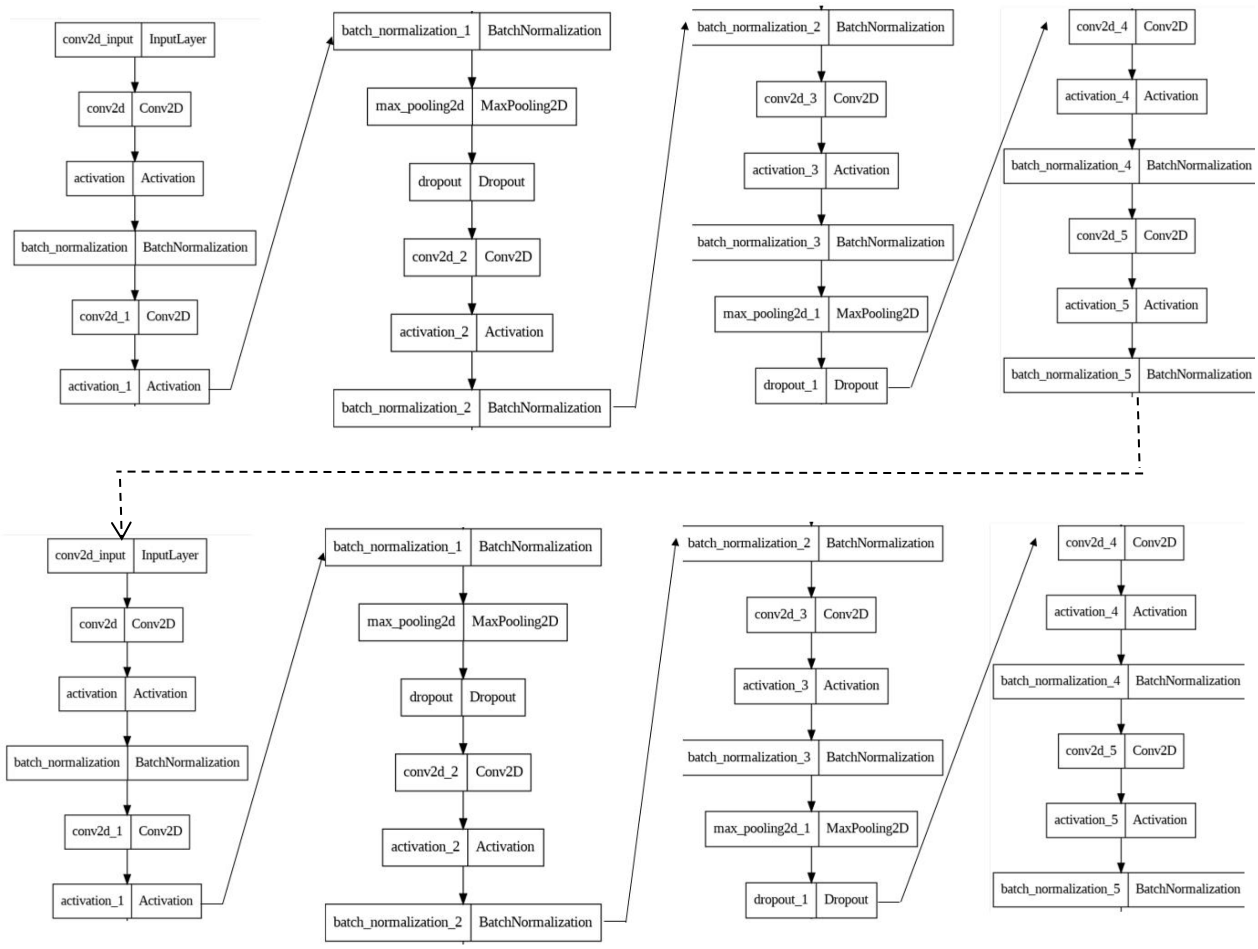

Figure 5.2: CNN Model Architecture

5.1.6 Building the model

We will utilize the sequential model offered by the popular deep learning framework, Keras, to build the model. After that, we'll add a number of layers to create a convolutional neural network (CNN). To be more precise, we start by adding the first two Conv2D layers, each with five and five kernel size having thirty two filters. The different characteristics in the input photos are detected by these layers. After that, we include a MaxPool2D layer with a pool size of (2,2). With this option, the image's spatial dimensions are essentially reduced by a factor of two since the layer will determine the highest value inside each 2 x 2 sector.

We incorporate a dropout layer with a dropout rate of 0.25, which means that randomly, 25% of the neurons will be silenced after each training iteration in order to prevent overfitting. By using this method, it is possible to prevent the model from being unduly dependent on any specific neurons. To further improve the feature extraction procedure, we repeat the Conv2D, MaxPool2D, and dropout layers after these layers, but with modified values.

Upon extracting characteristics from the preceding layers, the first dense layer—which comes after the flatten layer—is intended to teach itself complicated representations. Another dropout layer is added in order to further reduce overfitting. Lastly, we include an additional dense layer with seven nodes that functions as the output layer. The seven categories of the Facial Expression Detection challenge are represented by these nodes.

The softmax activation function is used in this last dense layer to translate the output into probability values. The model can predict the expression with the highest probability out of the seven alternatives by using these probabilities, which show the likelihood of each facial emotion. Our convolutional neural network is built to efficiently learn and recognize facial expressions from photos using this arrangement of layers.

5.2 Inception-V3 Model

5.2.1 Importing the necessary libraries

With pre-trained models like ResNet50 for transfer learning and specialized modules like `ImageDataGenerator} for image preprocessing and augmentation, TensorFlow and Keras offer tools for building, training, and assessing neural network models. File management and numerical operations are supported by standard libraries like `os` and `numpy`, while `scikit-learn` helps with dataset splitting for training and testing. `matplotlib` is also used for data visualization, and `argparse` makes command-line parameter parsing easier. Smooth functioning across multiple Python versions is ensured using compatibility imports.

5.2.2 Retrieving the images:

The images are transformed into NumPy arrays that can be fed into the model after being resized to 224 by 224 pixels.

5.2.3 Splitting the dataset

The dataset is divided into test and train in a ratio of 20% and 80% respectively.

5.2.4 High Level Architecture of Inception-V3 model

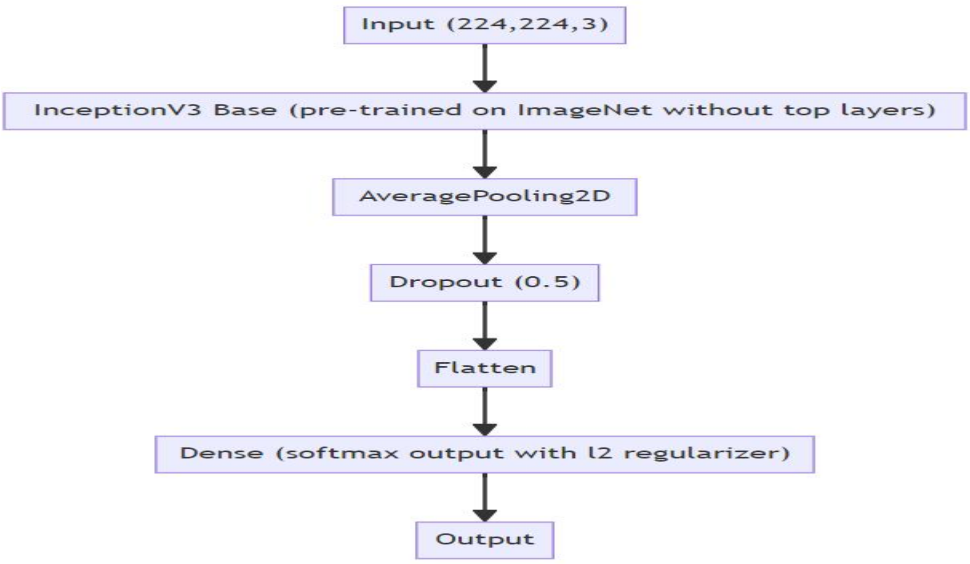

Figure 5.3: High level Architecture of Inception-V3 model

5.2.5 Building the model

The process of building the model begins with importing essential libraries from Keras and TensorFlow, which facilitate constructing and training deep learning models. At the core of this model is the InceptionV3 architecture, a highly efficient convolutional neural network pre-trained on the ImageNet dataset. By initializing InceptionV3 without its top layers and specifying the input shape as (224, 224, 3), we leverage its powerful feature extraction capabilities. The output from the InceptionV3 base model undergoes further processing through an AveragePooling2D layer, which reduces the spatial dimensions of the feature maps. This is followed by a Dropout layer with a dropout rate of 0.5 to mitigate overfitting by randomly setting a fraction of input units to zero during training. The subsequent Flatten layer converts the 2D feature maps into a 1D vector, preparing it for the final classification stage. The model's output layer is a Dense layer with a number of units equal to the number of classes in the dataset, utilizing the 'glorot_uniform' initializer and L2 regularization to prevent overfitting, and employing a softmax activation function for multi-class classification.

The model is then compiled using the SGD optimizer with a learning rate of 0.1 and momentum of 0.9, optimizing it for categorical cross-entropy loss and tracking accuracy as the evaluation metric. Additionally, several callbacks are configured to enhance the training process: ModelCheckpoint saves the best-performing model based on validation accuracy, CSVLogger records the training process, and LearningRateScheduler adjusts the learning rate according to a predefined schedule to ensure optimal training dynamics. The learning rate schedule starts at 0.001 and decreases progressively based on the epoch count. If the specified model file does not exist, the newly built model is saved. This comprehensive approach ensures that the model is not only well-structured and capable of leveraging pre-trained weights but also robustly trained and evaluated for performance.

CHAPTER 5
RESULTS

Table 5.1: Accuracy Comparison

Model	Accuracy
CNN Model	98.45%
InceptionV3 Model	70%

Table 5.2: Evaluation Metric Comparison

Metric	CNN Model	InceptionV3 Model
Recall	0.987	0.67
Precision	0.980	0.67
F1 Score	0.987	0.66

CHAPTER 6
CONCLUSION AND FUTURE WORK

The Stress Detection System is crafted to evaluate employee stress by analyzing images submitted by verified users, ensuring the reliability of the framework. Upon successful registration and login, users can upload images and utilize the live camera feature. Following image upload, the system generates a stress level output displayed atop a bounded box, indicating emotions such as anger, sadness, happiness, disgust, and neutrality. The model is constructed using the CNN Model Architecture, allowing for accurate predictions. We assess the model's performance by calculating accuracy, recall, F1 score, and generating a confusion matrix. Our system offers effective solutions for stress management, fostering conducive working conditions for employees, and optimizing productivity throughout work hours.

FUTURE SCOPE

The suggested approach integrates image processing and deep learning to detect stress. The procedure involves collecting and analyzing images to extract relevant features. In addition to utilizing the Live Cam, the video feature can prove advantageous for future work employing diverse algorithms. The outputs from algorithm processing were employed to train and test the model using a test dataset. Although the results obtained are preliminary, attributed to the limited number of participants and technical details, the significant contribution of this paper lies in enabling end-users to accurately identify ongoing stress, thereby mitigating potential health risks in the future. A more extensive population study will be a focus of our upcoming endeavors.

REFERENCES

[1] Widanti, N., Sumanto, B., Rosa, P., Miftahudin, M.F.. Stress level detection using heart rate, blood pressure, and gsr and stress therapy by utilizing infrared. In: Industrial Instrumentation and Control (ICIC), 2017 International Conference on. IEEE; 2017.

[2] Ravinder Ahujaa, Alisha Bangab “Mental Stress Detection in University Students using Machine Learning Algorithms”, International Conference on Pervasive Computing Advances and Applications,2019.

[3] Sharma, L.D., Bohat , V.K. , Habib, M. “ Stress Detection Using EEG Signals”, Journal of Advanced Research in Dynamical and Control Systems,2017.

[4] G. Giannakakis, D. Manousos, F. Chiarugi, “Stress and anxiety detection using facial expressions from videos,” Biomedical Signal processing and Control”, vol. 31, pp. 89 - 101, January 2017.

[5] Nisha Raichur, Nidhi Lonakadi, Priyanka Mural, “Detection of Stress Using Image Processing and Machine Learning Techniques”, vol.9, no. 3S, July 2017.

[6] U. S. Reddy, A. V. Thota and A. Dharun, "Machine Learning Techniques for Stress Predict ion in Working Employees," 2018 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Madurai, India, 2018.

[7] Reshmi Gopalakrishna Pillai, Mike Thelwall, Constant in Orasan, “ Detection of Stress and Relaxation Magnitudes for Tweet s”, International World Wide Web Conference Committee ACM, 2018.

[8] Huijie Lin, Jia Jia, JiezhonQiu, “Detecting stress based on social interactions in social networks”, IEEE Transactions on Knowledge and Data Engineering, 2017.

[9] Enrique Garcia-Ceja, Venet Osmani and Oscar Mayora, “ Automatic Stress Detection in working environment s from smart phones’ accelerometer data: A First Step”, arXiv:1510.04221v1 [cs.HC] 14 Oct 2015.

[10] Saskia Koldijk , Mark A. Neerincx, and Wessel Kraaij, “ Detecting Work Stress in offices by combining unobtrusive sensors”, IEEE Transactions on affective computing 2018.

[11]Virginia Sandulescu, Sally Andrews, David Ellis, “Stress Detection using wearable physiological sensors”, Springer International Publishing Switzerland 2015.

[12] Ankita Patil, Rucha Mangalekar, Nikita Kupawdekar, Viraj Chavan, Sanket Patil, and Ajinkya Yadav, “ Stress Detection in IT Professionals By Image Processing And Machine Learning,” International Journal of Research in Engineering, Science and Management , Vol. 3, No.1, January-2020.

[13] B. Padmaja, V. V. Rama Prasad and K. V. N. Sunitha, “ A Machine Learning Approach for Stress Detection using a wireless Physical Activity Tracker,” International Journal of Machine Learning and Computing, February, Vol. 8, No. 1, February 2018.

[14] Zenonos, A., Khan, A., Kalogridis, G., Vatsikas, S., Lewis, T., Sooriyabandara, “Healthy Office: Mood recognition at work using smartphones and wearable sensors”, IEEE International Conference on Pervasive Computing

and Communication Workshops 2016.

[15] Tanev, G., Saadi, D.B., Hoppe, K., Sorensen, H.B "Classification of acute stress using linear and non-linear heart rate variability analysis derived from sternal ECG" 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society.

[16] Deng, Y., Wu, Z., Chu, C.H., Zhang, Q., Hsu, D.F.. Sensor feature select ion and combination for stress identification using combinatorial fusion. International Journal of Advanced Robotic Systems 2013.

[17] Ghaderi, A., Frounchi, J., Farnam, A.. Machine learning based signal processing using physiological signals for stress detection. In: 2015 22nd Iranian Conference on Biomedical Engineering (ICBME). 2015.

[18] Mrs. Megha V Gupta Dr. Shubhangi Vaikole "Recognition of Human Mental Stress Using Machine Learning Paradigms",2020.

[19] Liu, D., Ulrich, M.. Listen to your heart: Stress prediction using consumer heart rate sensors 2015.

[20] Nakashima, Y., Kim, J., Flutura, S., Seiderer, A., Andre, E.. Stress recognition in daily work. In: ´ International Symposium on Pervasive Computing Paradigms for Mental Health. Springer; 2015.

[21] Xu, Q., Nwe, T.L., Guan, C.. Cluster-based analysis for personalized stress evaluation using physiological signals. IEEE journal of biomedical and health informatics 2015.

[22] Gjoreski, M., Gjoreski, H., Lustrek, M., Gams, M.. Continuous stress detection using a wrist device: in laboratory and real life. In: Proceedings of the 2016 ACM International Joint Conference on www.jespublicat ion.com Pervasive and Ubiquitous Computing: Adjunct . ACM; 2016.

[23] Palanisamy, K., Murugappan, M., Yaacob, S.. Multiple physiological signal-based human stress identification using non-linear classifiers. Elektronika irelekt rotechnika 2015.

[24] Villarejo, M.V., Zapirain, B.G., Zorrilla, A.M.. A stress sensor based on galvanic skin response (gsr) controlled by zigbee. Sensors,2012.

[25] Selvaraj, N.. Psychological acute stress measurement using a wireless adhesive biosensor. In: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2015.

[26] Sriramprakash.S ∗, Prasanna Vadana. D, O. V. Ramana Murthy "Stress Detection in Working People" 7th International Conference On Advances In Computing & Communications, ICACC 2017, August 22-24, 2017.

[27] Koldijk, S., Neerincx, M.A., Kraaij, W.. Detecting work stress in offices by combining unobtrusive sensors. IEEE Transactions on Affective Computing 2016.

PAPER NAME

INCEPTION_MERGED_PLAG.pdf

WORD COUNT	CHARACTER COUNT
6973 Words	**39155 Characters**
PAGE COUNT	FILE SIZE
29 Pages	**693.2KB**
SUBMISSION DATE	REPORT DATE
May 31, 2024 6:16 AM GMT+5:30	**May 31, 2024 6:17 AM GMT+5:30**

15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 6% Internet database
- 6% Publications database
- Crossref database
- Crossref Posted Content database
- 13% Submitted Works database

Excluded from Similarity Report

- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less then 8 words)

15% Overall Similarity

Top sources found in the following databases:

- 6% Internet database
- 6% Publications database
- Crossref database
- Crossref Posted Content database
- 13% Submitted Works database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	analyticsvidhya.com Internet	1%
2	Hellenic Open University on 2023-01-19 Submitted works	1%
3	University of North Texas on 2024-02-05 Submitted works	<1%
4	Gitam University on 2024-03-27 Submitted works	<1%
5	University of North Texas on 2024-02-11 Submitted works	<1%
6	LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning", Nat... Crossref	<1%
7	abhishekiiit.github.io Internet	<1%
8	Suresh Kumar Kanaparthi, Surekha P, Lakshmi Priya Bellamkonda, Bha... Crossref	<1%

9	"Machine Learning for Cyber Physical System: Advances and Challeng... Crossref	<1%
10	Griffith College Dublin on 2023-12-24 Submitted works	<1%
11	American University of Culture and Education on 2023-04-20 Submitted works	<1%
12	University of Leicester on 2023-09-11 Submitted works	<1%
13	University of East London on 2023-09-11 Submitted works	<1%
14	University of Leicester on 2023-05-31 Submitted works	<1%
15	as-proceeding.com Internet	<1%
16	Gitam University on 2024-03-27 Submitted works	<1%
17	National Institute of Technology, Rourkela on 2023-05-24 Submitted works	<1%
18	Universiti Teknologi Malaysia on 2021-11-14 Submitted works	<1%
19	University of Salford on 2023-05-11 Submitted works	<1%
20	Western Illinois University on 2020-12-08 Submitted works	<1%

21	**Western Illinois University on 2021-12-11** Submitted works	<1%
22	**Singh, Shekhar. "Facial Expression Recognition Using Convolutional Ne...** Publication	<1%
23	**UOW Malaysia KDU University College Sdn. Bhd on 2023-11-29** Submitted works	<1%
24	**Heriot-Watt University on 2023-04-24** Submitted works	<1%
25	**Queen Mary and Westfield College on 2023-08-24** Submitted works	<1%
26	**South Bank University on 2024-05-23** Submitted works	<1%
27	**frontiersin.org** Internet	<1%
28	**Geetha C, Prabhu V, Bandla Venkata Akash, Bhuvanendra Chowdary V, ...** Crossref	<1%
29	**ebin.pub** Internet	<1%
30	**Amity University on 2020-06-02** Submitted works	<1%
31	**University of East London on 2021-05-11** Submitted works	<1%
32	**upgrad.com** Internet	<1%

33	UC, San Diego on 2021-06-04 Submitted works	<1%
34	Yeditepe University on 2023-06-07 Submitted works	<1%
35	fenix.tecnico.ulisboa.pt Internet	<1%
36	9pdf.net Internet	<1%
37	Liverpool John Moores University on 2022-04-10 Submitted works	<1%
38	University College London on 2018-09-10 Submitted works	<1%
39	"Advances in Multimedia Information Processing - PCM 2016", Springe... Crossref	<1%
40	Harpreet Singh Bedi, Praveen Kumar Malik, Rajesh Singh, Ramendra Si... Crossref	<1%
41	Study Group Australia on 2018-10-07 Submitted works	<1%
42	University of North Texas on 2024-02-12 Submitted works	<1%
43	cemetech.net Internet	<1%
44	doc.ic.ac.uk Internet	<1%

45	Erasmus University of Rotterdam on 2020-01-24 Submitted works	<1%
46	Higher Education Commission Pakistan on 2024-04-26 Submitted works	<1%
47	National College of Ireland on 2020-09-27 Submitted works	<1%
48	Rochester Institute of Technology on 2020-04-13 Submitted works	<1%
49	University of Newcastle upon Tyne on 2017-11-06 Submitted works	<1%

ANNEXURE-IV

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

PLAGIARISM VERIFICATION

Title of the Thesis DETECTION OF STRESS IN IT EMPLOYEES USING DEEP LEARNING Total Pages ____29____ Name of the Scholar Aditya Kanwar Supervisor (s)

(1) Dr.Varsha Sisaudia______________________________

(2)____

(3)____

Department INFORMATION TECHNOLOGY, DELHI TECHNOLOGICAL UNIVERSITY

This is to report that the above thesis was scanned for similarity detection. Process and outcome is given below:

Software used:______Turnitin________ Similarity Index:___15 %___ , Total Word Count: 6973

Date:_31/05/2024

Candidate's Signature **Signature of Supervisor(s)**

REGISTRAR, DTU (RECEIPT A/C)

BAWANA ROAD, SHAHABAD DAULATPUR, , DELHI-110042
Date: 30-May-2024

SBCollect Reference Number :	DUM6891416
Category :	Miscellaneous Fees from students
Amount :	₹2000
University Roll No :	2K22/ISY/03
Name of the student :	Aditya Kanwar
Academic Year :	2023-24
Branch Course :	MTech ISY
Type/Name of fee :	Others if any
Remarks if any :	Examination fee Thesis submission fee
Mobile No. of the student :	9991427366
Fee Amount :	2000
Transaction charge :	0.00
Total Amount (In Figures) :	2,000.00
Total Amount (In words) :	Rupees Two Thousand Only
Remarks :	Examination Fee Thesis submission fee
Notification 1:	Late Registration fee Rs.50 per day, Hostel Room Rent for internship Rs.1000 per month, Hostel Cooler Rent Rs.1000 per year, I card Rs.200, Character certificate Rs.200, Migration certificate Rs.200, Bonafide certificate Rs.200, Special certificate Rs.500, Provisional certificate Rs.200, Duplicate Mark sheet Rs.500, Training Diary Rs.70
Notification 2:	Fee Structure Rs.200, Admit Card Rs.50. Transcript fee and other fee rates has to be confirmed from the Academic Cell prior to remit the fees online by the student.