STRESS DETECTION AND REDUCTION FOR IT PROFESSIONAL

A PROJECT REPORT

Submitted by

SREEKRISHNA PA 181001093
SRIMANJEY R 181001095
SUGUMARAN B 181001102

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

RAJALAKSHMI ENGINEERING COLLEGE, THANDALAM

ANNA UNIVERSITY: CHENNAI 600 025

APRIL 2022

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"STRESS DETECTION AND REDUCTION FOR IT PROFESSIONAL"** is the bonafide work of **SREEKRISHNA P A (181001093), SRIMANJEY R (181001095), SUGUMARAN B (181001102)** who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred earlier occasion on this or any other candidate.

SIGNATURE
Dr. SAKTHIVELAN R G
HEAD OF THE DEPARTMENT
Professor
Information technology,
Rajalakshmi Engineering College,
Thandalam,
Chennai – 602105.

SIGNATURE
DR. KAYALVIZHI S
SUPERVISOR
Assistant Professor [SG]
Information technology,
Rajalakshmi Engineering College,
Thandalam,
Chennai – 602105.

EXTERNAL EXAMINER

INTERNAL EXAMINER

ACKNOWLEDGEMENT

I would like to thank our Hon. Chairperson **Dr. Thangam Mega Nathan** and our Hon. Vice-Chairman M**r. M. Abhay Shankar for** giving us the opportunity to showcase our skill. Then I would also like to extend my heartfelt sincere gratitude to our Principal **Dr. S. N. Murugesan** for providing us this platform to expose our practical knowledge.

I would also like to extend my heartfelt thanks t**o Dr. R. G. Sakthivelan** (Head of Information Technology Department) for giving us his able support and encouragement.

I must also emphasise the point that this project would not have been possible without the highly informative and guidance of **Dr. Kayalvizhi S** (Assistant Professor, SG) whose vast knowledge and experience has greatly helped me in this project. Next, I would like to thank project coordinator **Dr. Kayalvizhi S** (Assistant Professor, SG) for her support throughout the project.

I express my sense of gratitude to all of our staff members and lab assistants of the Information Technology Department who helped me during my project. I am highly grateful to my family and friends for extending their help whenever necessity arose and their continuous help to complete my project successfully.

TABLE OF CONTENTS

ABSTRACT

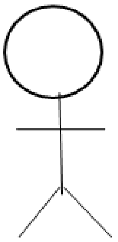
Nowadays, IT industries are developing at a rapid pace in the market by introducing new technologies and deadlines and it requires its employees to cope up with its pace which leads to stress. This develops stress within its employees making them less effective. Hence this study aims to develop a system to detect the stress level of IT professionals and provide them with remedies to reduce stress. The system monitors the facial emotions of the employees at real-time through webcam and it uses the mini-Xception model of the CNN algorithm to identify the emotions of the employee and classifies them. When an employee is classified as “stressed” the system calculates the stress level of the employees by providing questionnaire and displacement of eyebrow from its mean position. Based on the computed stress level of the employee the system provides remedies such as playing classic music, offering to meditate, play stress reduction games, meditate for a few minutes, provide food coupons to the employees, and participation in sports activities. The dataset used is FER-2013 dataset that has approximately 30000 images to train the system. The achieved accuracy of the system is 95.6%. This stress detection and reduction system can also be applied to other industries such as banking sector, mechanical industries. This system helps in reducing stress in employees and thereby promoting a healthy work environment and work culture making the employees to stay motivated, focused and efficient.

KEYWORDS:

Convolutional Neural Network, Facial emotion, mini-Xception.

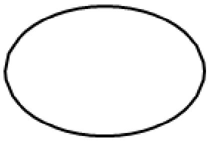
LIST OF FIGURES

LIST OF SYMBOLS



Actor

Standard UML icon for actor is “stick man” icon the name of the actor above or below the icon



Use case

Every use case must have a name. Use case is shown as an ellipse containing the name of the use case.

Arrow

An association between an actor and use case indicates that actor and the use case communicate with each other. An actor could be associated with one or several use cases.

Block

It represents the action that is performed in the sequence.

LIST OF ABBREVIATIONS

I. CNN - Convolutional Neural Network

II. FER Model – Facial Emotion Recognition

III. FFNN – Feed Forward Neural Network

IV. RGB image – Red Green Blue image

V. SFM – Structure from Motion

VI. KDEF – Karolinska Directed Emotional Faces

VII. KNN – K-Nearest Neighbor

VIII. AU – Action Units

IX. SVM – Support Vector Machine

X. SVR – Support Vector Regression

XI. HOG – Histogram of Oriented Gradients

XII. HIGO - Histogram of Image Gradient Orientation

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Nowadays, IT industries are developing at a rapid pace in the market by introducing new technologies and deadlines and it requires its employees to cope up with its pace which leads to stress. This in turn develops enormous stress in employees leading to depletion in focus and work motivation with in turn reduces the company's profit and reputation. Therefore, companies are spending huge amount of money for ensuring the well-being of the employees. An automatic stress detection and reduction system in real-time can help the company in having control over its employee's stress and hence creating a healthy work environment. We have used mini-Xception model of the Convolutional Neural Network for identifying the emotions of the employee to detect stress. The proposed system has proved to be efficient than the other existing systems. To recognize the stress detection and reduction on computer using webcam. Therefore, we are designing a system that can daily record a person's stress level and time and help the user with regulated breathing as a way of reducing their momentary stress. We can detect the stress and reduce through playing a music, play games, providing food coupons and take a rest of the day because of stress reduction.

Stress management systems play a major role to notice the stress levels that disrupts our socio-economic mode. As World Health Organization (WHO) says, Stress may be a psychological state drawback moving the lifetime of one in four voters. Human stress results in mental furthermore as socio-fiscal issues, lack of transparency in work, poor operating relationship, depression and eventually commitment of suicide in severe cases.

1.2 OBJECTIVE

The objective of the system is to monitor the employees and detect employee's emotion for stress and provide remedies to reduce it. As the IT sector is prone to stress, and due to increasing accidents due to stress, we companies are in need for systems to detect and reduce stress. This has resulted in development of stress detection and reduction system.

This system uses the mini-Xception model of the Convolutional Neural Network to monitor the emotions of the employee and identify stress in real-time by using webcam. Each frame extracted from the video will be passed through a mini-Xception model, to extract the information present in that current frame. This in turn compares the emotions with the emotions in the dataset to identify whether an employee is stressed or not. Hence the proposed system is fast in comparing the emotion in each frame with the dataset to identify stress.

This system has helped organizations to identify stress in real-time through webcam and provide the necessary measures to reduce stress. Though there are existing systems to detect stress, they are inefficient as they use sensors with is not a suitable solution for the IT sector. It is therefore necessary to develop an efficient and possible solution for detecting stress. This system promises to provide such a solution.

1.3 ORGANIZATION OF THE PROJECT REPORT:

The project report is organized into six sections:

- Introduction
- Literature survey
- Proposed System design

- System Implementation
- Performance analysis
- Conclusion and Future work

1.4 CONCLUSION

There are many stress level detection projects that are implemented and published to detect the stress level using sensors, smart watch, EEE signals. But our system is completely software oriented for both stress detection and reduction. Stress detection is done by using webcam by which the employee can monitored. The captured video is then viewed frame by frame to determine whether an employee is stressed or not. When they are classified as stressed their stress level can be calculated by measuring the displacement of the eyebrow at real-time and provide the remedies according to the level of stress. The video is captured automatically when the authenticate user is logged in. The test results prove that the developed system can predict emotions and classify if an employee is stressed. Automatic stress detection and reduction model help organizations in creating a healthy work environment and boost the productivity of its employees.

CHAPTER 2

LITRATURE SURVEY

2.1 INTRODUCTION

Daily life stress is an important problem of our modern society. It is a growing issue and it has become an unavoidable part of our daily lives. Psychological stress types can be listed as acute and chronic. Acute stress is more prevalent than chronic stress. American Psychological Association noted that the causes of acute stress are pressure from recent past and near future. Athletic challenges, test taking, or anxiety when meeting new people can induce acute stress. On the other hand, long-standing pressures and demands as a result of socioeconomic conditions, difficulties in interpersonal relationships, or an unsatisfying career can trigger chronic stress. If chronic stress is not handled properly, it could result in serious health issues. Since symptoms of acute stress are more apparent than chronic stress symptoms, acute stress is more widely investigated. After musculoskeletal illnesses, which also could be stress-related in some cases, stress is one of the most significant health problems in the world. The effect of stress on human health depends on the stress type. Emotional distress, muscular ache and tension, back pain, headache, heartburn, digestive tract issues, and overarousal can be named as the effects of acute stress. Overarousal can cause heart attacks, arrhythmias, and even sudden death for people with heart conditions. Effects of the chronic stress on human health are akin to those of acute stress however it can damage physical conditions more. Possible causes of the chronic stress can be listed as hypertension and coronary disease, irritable bowel syndrome, gastroesophageal reflux disease, generalized anxiety disorder, and depression. The above-mentioned stress-related diseases also affect the economy by increasing absenteeism, staff turnover, presenteeism, and tardiness. These problems decrease the production and increase the work-related costs. Public surveys unveiled that at least half of the European workers are subjected to stress at work. Furthermore, at least half of the lost working days in the business sector are assumed to be caused by work-related stress and psycho-social risks.

Researchers found out that stress should be handled when the symptoms first come out to avoid the long-term consequences. In other words, stress must be discovered in early stages to refrain from more damages and impede it from being chronic. The above-mentioned damages of stress on human health and detriments to social life and economy have forced researchers to come up with an automatic stress monitoring scheme which exploits smart wearable devices and advanced affective computing algorithms. This scheme can be applied in automobiles, airplanes, factories, and offices, at job interviews and daily life environments. This scheme can further compute social stress stages during meetings or mutual intercommunication. The ideal scheme should be applicable to daily life, i.e., it should use unobtrusive sensors and devices which users can wear easily in their daily routines. This work involves the development of a stress level detection system that uses face recognition from co-ordinates of eyebrow. Our system can also be applied to daily life of IT professionals. In real-life settings, movements of individuals are unrestricted and artifacts occur because of that. In order for our system to be applicable in these settings, we applied several novel artifact detection and removal strategies. These artifact detection algorithms are developed for specific algorithm and their performances are scientifically proven. Our work has further extracted features from the remedies of stress reduction with our system. From these features, we classified the stress level of an individual by employing machine learning algorithms. To test our system in real-life settings, we collected physiological signals of participants in an algorithmic programming summer camp via co-ordinates of the eye of an individual.

2.2 RELATED WORK

[1] This study focuses on developing a system for predicting depression using feed – forward neural network model and uses depression scale to measure the depression level. Calculating the improvement level is based on the depression levels identified and is visually represented in a dashboard to monitor depression level improvements for the therapist and the patient. The main aim is to provide a solution for Youths who require therapy for depression. The proposed Solution consists of a mobile app which will assist the clients. Then the therapist engages with the online and monitor them. The improvement level of depression of client is recorded on each therapy Session. The FFNN is trained through back propagation by comparing the output Predicted and the expected output with correcting the weights to minimize the errors. The training is done with the videos from training dataset. These videos are sent as input frame by frame to detect depression. The system uses the data of the patient to identify any improvement in the depression level. In case a patient is depressed the system previous data to see if the patient is better than the previous session. But the prediction of the therapy period or the time required for identifying the type of therapy for the depressed patient is high which leads to serious effects.

[2] The authors of this paper propose a novel method for human emotion recognition from a single RGB image. This system is done by the progress of 3D facial reconstruction from images and rich dynamic information accompanying videos of facial performances. The data is collected from a large-scale dataset of facial videos. Here the camera parameters are estimated using rigid Structure from Motion (SFM). The system does not seek to estimate the full degrees of freedom of the 3D facial but it reduces the allowed degrees of freedom by imposing the constraint that it is synthesised using the 3D face model. The system keeps track of the face throughout the process to remove false detections arising due to a failure in the face detector or out of context detections. But the system needs 3D restructuring of the faces at real-time

where 3D rendering needs to be customized and is expensive. Also reconstructing the faces at real-time takes more time.

[3] This study employs the Convolutional Neural Network and Deep Neural Network to develop a facial emotion recognition model that categorizes a facial expression into seven different emotions categorized as Afraid, Angry, Disgusted, Happy, Neutral, Sad and Surprised. The dataset used to train and test the different architecture of hidden layer in the deep neural network of the convolutional neural network is the Karolinska Directed Emotional Faces (KDEF). The dataset consists a total of 4900 images of 562*762. The dataset is divided into test and train dataset in 80% - 20% split. The train dataset consists of 3920 images divided into seven categories and the test dataset consist of 980 images divided into seven categories. The images are scaled down from 562*762 pixels to 256*256 pixels before feeding it into the CNN model. The models were trained for 25 epochs with 3920 steps per epochs for training set and 980 steps for validation set. The images present in the dataset were pre-processed by using the Image Data Generator class which generates batches of tensor images. In this method the images were rescaled by a factor up to 1/255. The images were randomly flipped in horizontal direction in order to generate randomness in the input image while training the model. Images were sheared in counter clockwise direction up to 0.2 degrees and the zoom range for the images were set to be about 0.2 to provide random zoom. Venturi Architecture is the proposed architecture for the hidden layer of the deep neural network in the convolutional neural network. The architecture consists of 6 layers in the hidden layer with one output layer consisting of 7 nodes based on the 7 different categories in which the facial emotions are classified. The venture architecture proved to have more training and test accuracy. Though the model is having more accuracy, the model is trained with limited dataset and has low accuracy for disgust emotion.

[4] This study aims to design a real driving task to extract data and proposes a driver's driving stress monitoring model based on driving behaviour, driving environment, and route familiarity. Based on the psychological data and driver stress inventory (DSI) results, the study used a K-means 3D cluster analysis to obtain the evaluation method of driving stress and constructed an extreme gradient boosting (XGBoost) model to monitor driving stress. However, psychological data collection sensors have not been widely used in conventional vehicles, which make it impossible to apply the results of that research to actual driving tasks on a daily basis, even if the accuracy is high. This study designs a real driving task to extract data and proposes a driver’s driving stress monitoring model based on driving behaviour, driving environment, and route familiarity. The driving behaviour is described by the speed and acceleration of the vehicle, and the driving environment is quantified by a dilated residual networks (DRN) model that divides the video image from the full region into sub regions according to the distribution of the driver’s attention. Based on the psychological data and driver stress inventory (DSI) results, the study used a K-means 3D cluster analysis to obtain the evaluation method of driving stress and constructed an extreme gradient boosting (XGBoost) model to monitor driving stress. The model’s performance indicators, accuracy, sensitivity, and precision, reached 91.18% – 93.25%, 84.13% – 89.37%, and 90.25% – 91.34%, The study also summarises the ranking of effects of different scene elements on driving stress for each visual field. The system lacks diversity as the system is limited because the drivers face more difficulty when introduced to new routes.

[5] This study uses micro-expression to detect facial emotions as they are capable of revealing the genuine emotions that people attempt to hide. Micro expression is a subtle and indiscernible motion of human facial muscles. Micro-Expression is effective in spotting and recognition of expressions and relies heavily on discriminative features which are extracted from image sequences. This approach uses a set of three-dimensional blocks. Based on these, more robust and reliable histograms can be

generated, which are suitable for different types facial analysis tasks. The study uses a proposed spatiotemporal texture map (called STTM) to convolve an input video sequence with A 3D Gaussian kernel function in a linear space representation. By calculating the second-order moment matrix, the spatiotemporal texture and the histogram of the micro-expression sequence are obtained. This proposed a spatiotemporal LBP with Integral Projection (STLBP-IP); according to this approach, LBP is evaluated on the integral projections of image pixels obtained in the Horizontal and vertical directions. These authors also proposed a histogram of image Gradient orientation (HIGO) which uses a simple vote Rather than a weighted vote. HIGO maintains good Invariance of the geometric and optical deformation of the Image. The system lacks proper dataset for training which impacts the emotion recognition as the number of micro-expression samples is too limited to facilitate proper training.

[6] This paper focuses on the automated identification of facial Action Units (AU) as Quantitative indices in order to discriminate between neutral and stress/anxiety state. Thus, a model for automatic Recognition of facial action units is proposed being trained in two available annotated facial datasets, the UNBC and The BOSPHORUS datasets. Facial features, both geometric (non-rigid deformations of 3D shape of AAM landmarks) and Appearance (Histograms of Oriented Gradients) are extracted. The intensity of each AU was regressed using Support Vector Regression (SVR). A combined model was applied to the experimental dataset (SRD"15) containing neutral States and inducing stressful states related to types of stress. The results indicate that there is specific AU relevant to stress and the AU intensity significantly increased during stress leading to a more expressive human face. This study focuses on automatic stress identification from the intensity of facial AU which are estimated from trained SVM models. The procedure has 3 phases: pre-processing (including face Detection, AAM facial landmark estimation, face alignment/normalization, face warping), feature extraction (shape and appearance features), AU classification

(including PCA On appearance features, Support Vector Regression (SVR) Training and AU intensity estimation). It systematically categorizes human facial muscle movements and expressions based on anatomic functions. Additionally, it encodes actions related to eye gaze, head Pose and other actions. The most relevant/important AU features are investigated and selected in order to state their relevance with stress and to improve the performance of the stress model.

[7] The main aim of this paper is automatic depression recognition of facial Expressions associated with depressive behaviour. Algorithms to recognize depression typically explore Spatial and temporal information individually, by using 2D convolutional neural networks (CNNs) to analyse appearance information and then by either mapping facial feature variations or averaging the depression level over video frames. Extensive experiments are conducted on two datasets namely Audio-Visual Emotion Challenge 2013 and 2014 (AVEC2013 and AVEC2014) depression sub-Challenge datasets. The dataset Is organized into three distinct partitions: training, development and test sets. Each partition contains 50 videos which have a label corresponding to the depression level of a subject. The longest video reaches 50 minutes in duration and the shortest lasts 20 minutes. During acquisition of the videos, the subjects perform two tasks: Freeform and Northwind tasks. In the first, the subjects respond to questions like discuss a sad childhood memory. In the second one, subjects read audibly a short note. In both activities, the recordings are segmented into three partitions: training, development and test set. Each partition contains 50 videos. In total, there are 300 videos ranged in duration between 6 and 248 are provided with remedies appropriate to their obtained stress level.

2.3 CONCLUSION

In this paper, we explored the importance of spatial and temporal information for automatic depression assessment. They conducted this study by introducing a novel framework to represent the facial expression alterations called Multiscale Spatiotemporal Network (MSN). The architecture has the potential to encode rich spatiotemporal information of modifications in facial expressions using 3D convolutional layers with various kernel sizes, which allow the method to capture appearance and dynamics in different ranges. Such ability is important for modeling depressive behavior from facial expression variations. In the experiments carried out with benchmark AVEC2013 and AVEC2014 depression datasets, the proposed MSN demonstrated to be more effective than I3D and C3D architectures in exploring spatiotemporal information. Moreover, MSN achieved good results and outperformed state-of-the-art methods, showing its effectiveness for depression detection. We believe that the results of this work can contribute to the progress of automatic medical diagnosis based on face analysis. The basic building block of MSN has the potential to capture rich spatiotemporal features and can be explored for detecting other abnormalities reflective of diseases in person's facial expressions. As a future work, they intend to employ MSN model in another health care application based on facial information. But the proposed MSN does not pay high attention to this corner. Instead of that, the MSN focuses mainly on facial area that involves roughly eyes and mouth. For the patient with severe level of depression, the proposed MSN pays high attention to an area encompassing eyes and mouth which is slightly smaller than the one explored to the model with single kernel. Based on these observations, we can claim that the proposed MSN explores more efficiently the spatiotemporal information when compared with the model with single kernel.

CHAPTER 3

PROPOSED SYSTEM DESIGN

3.1 INTRODUCTION

The purpose of this study is the development of system that takes video as input, process the input, convert the video into images, extract the images, train the dataset, recognize the emotion of the employee and compare with the dataset and if the employee is stressed recognizing stress level through the coordinates of the eyebrow and through a survey form and finally provide remedies to reduce stress based on the calculated stress level.

Stress in IT industry is skyrocketing as 43% of employees are reported to be stressed. The existing systems identify stress in users but provide no mitigation to reduce the stress. The existing systems also use sensors which makes employees uncomfortable which is not feasible for the IT industry.

The system calculates the stress level of the employees only when they are identified to be stressed. The calculation of the stress level and providing the remedies is done only when the employee is classified as stress. This makes it possible for the non-stressed employee to continue working.

3.2 PROPOSED SYSTEM

The main aim of this system is to provide an appropriate solution for IT organizations that require means to reduce their employee's stress. The proposed solution involves a system which will assist the IT employees and the organization by providing remedies to the stressed employees.

The stress detection and reduction system are based on the analysis of the facial expression of the IT employees. The system analyses stress in two ways:

- The system analyses the stress of the employee through a pre-recorded video.

- When an IT professional sits in front of camera the system will be able to detect the facial expression in real-time.

The system processes the captured video and extracts images from it for emotion identification. The system uses the shape predictor package for detecting the face in the frame. The dataset used to train the system is FER2013. The dataset consists of approximately 30,000 images where the image size restricted to 48*48. The emotions are classified into the following types: 0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral. The system predicts stress by comparing the facial expression of the employees with the trained mini-Xception (FER MODEL) model. The system computes seven different types of emotions such as: "angry", "disgust", "scared", "happy", "sad", "surprised", and "neutral". An employee is classified as stressed if their emotions fall under the following category: "sad", "scared". If the employee is found to be stressed in the successive sections of time intervals the system classifies the employees as stressed and starts recognizes the shape of the employees face and calculates the coordinates of the eyebrow. The stress level of the employee is then calculated by the displacement of eyebrow from its mean position. Then the employee is also provided with survey form to assess their stress level. Based on the obtained stress level from real-time recording and through the questionnaires, the final stress level is calculated. The final stress level is analyzed and the employees are provided with remedies appropriate to their obtained stress level.

The remedies that the system provides include:

- Suggesting the employee to take a break
- Playing them a short meditation video to make them understand the benefits of meditation and start practicing them.
- Providing access to YouTube.
- Providing access to a gaming platform.
- Proving them with Food Coupons
- Granting a paid leave.

All the remedies are provided to the employee based on the computed level of stress level. The system also restricts access to YouTube and the gaming platform. The system provides access to such websites for only 10 minutes. By providing remedies to the employees, the system reduces the stress level of the employee and makes them more productive.

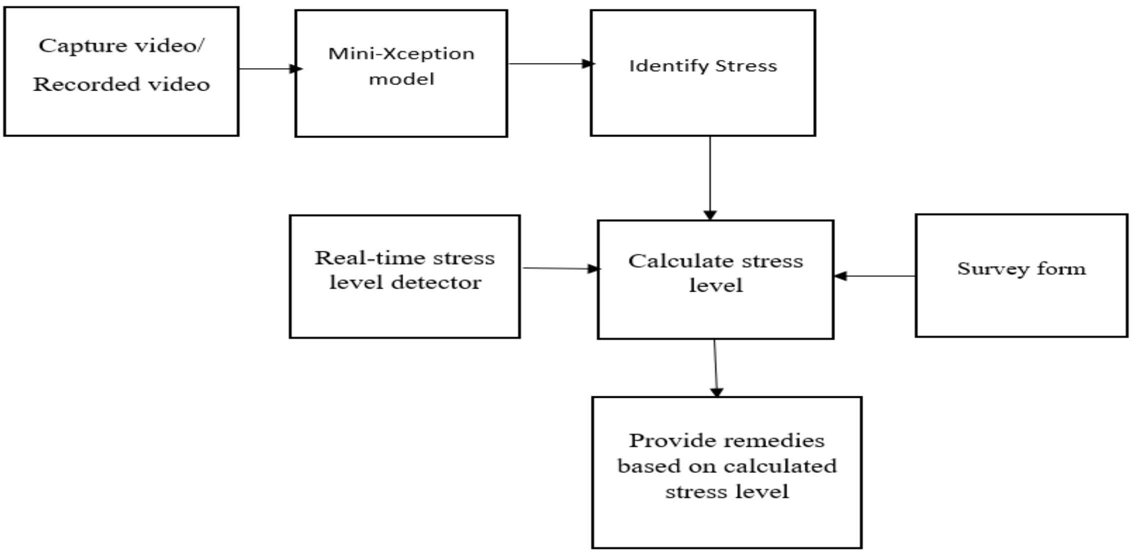


Figure: 3.1 System Architecture

3.2.1 ADVANTAGES OF THE PROPOSED SYSTEM:

- The detection of stress through webcam can be employed without any difficulty and without making the employees uncomfortable.
- Stress Detection and Reduction System enables employees to cope up with their issues caused by stress and stay motivated and happy.

3.3 SYSTEM ARCHITECTURE

3.3.1 BLOCK DIAGRAM

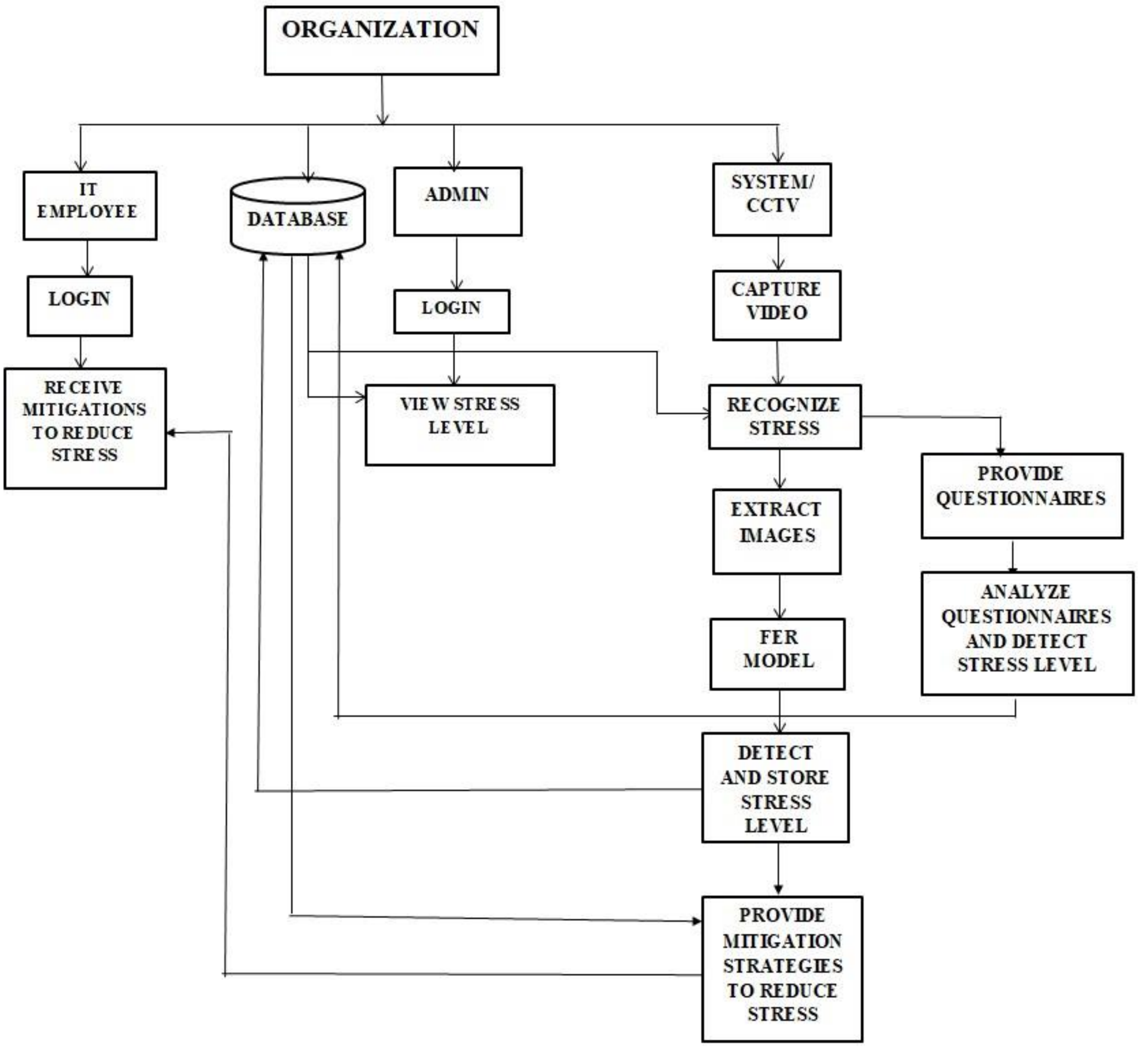


Figure. 3.2 Functional Architecture

3.3.2 USECASE DIAGRAM

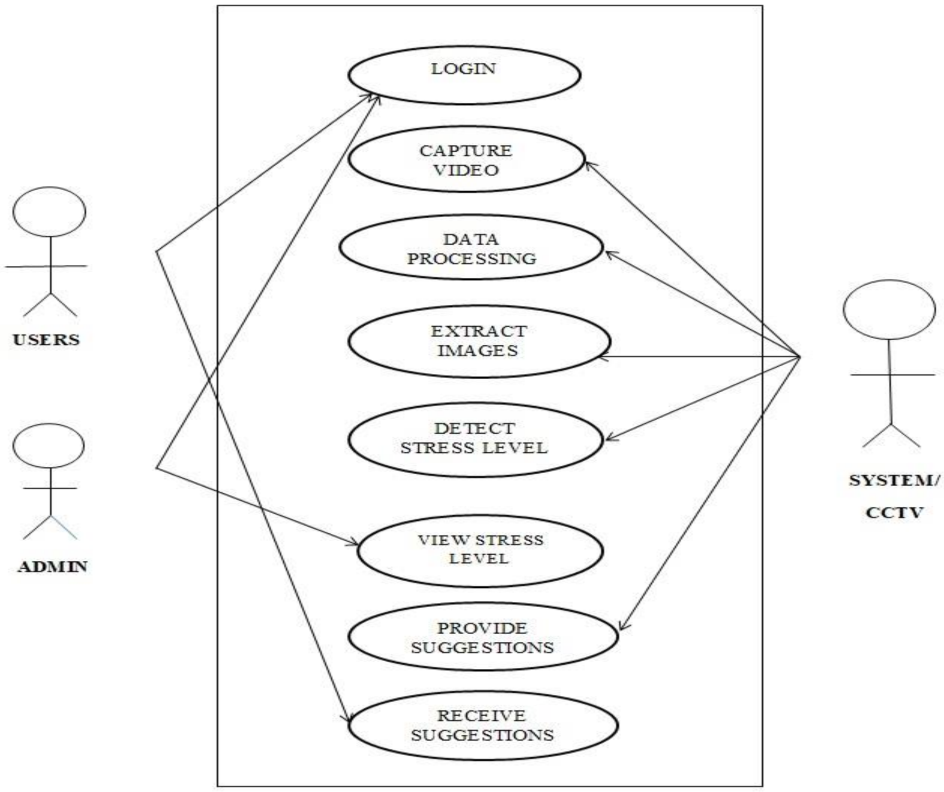


Figure. 3.3 Use case Diagram

The use case diagram represents the employees and the admin’s interaction with the system. The diagram shows the relationship between the employee and admins different use cases in which they are involved. In the above use case diagram, the actors are employee and admin. The employee can either upload the video or provide input at real-time. The admin can view the recorded stress levels of the employees from the dataset.

3.4 CONCLUSION

This proposed system presented an efficient model Stress detection and reduction. The system has proved to be more suitable for It industry and provide genuine outcome. The system can be easily deployed and benefits both the organization and its employees.

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 INTRODUCTION:

Using mini-Xception (FER Model) in CNN, the system identifies the emotion of the employees at real-time efficiently. This enables the system to identify whether an employee is stressed or not. The system then calculates the stress level and provides remedies to the employees only when they are classified as stressed.

4.2 HARDWARE AND SOFTWARE REQUIREMENTS

4.2.1 HARDWARE REQUIREMENTS

Hardware: Dual core

RAM: minimum 2 GB

Hard Disk: 500 GB

Processor:64-bit,1.80GHz minimum

Input: Any kind of video file with human face.

4.2.2 SOFTWARE REQUIREMENTS

Operating system: windows 10

Technology: Anaconda, flask, sqlite, web designing languages(HTML, css)

IDE: Jupyter notebook, Pyhton

4.3 TECHNOLOGIES USED

4.3.1 Convolutional Neural Network (CNN):

Convolutional neural network is basically a consist of large number of interconnected cells. The arrangement of cells is such that each cell receives an input and drives an output for subsequent cells. Each cell has a pre-defined.

The diagram below is a block diagram that depicts the structure and the work flow of a **Convolutional Neural Network.** The neurons are interconnected with each other in a serial manner. The network consists of a number of hidden layers depending upon the resolution of comparison of inputs with the dataset.

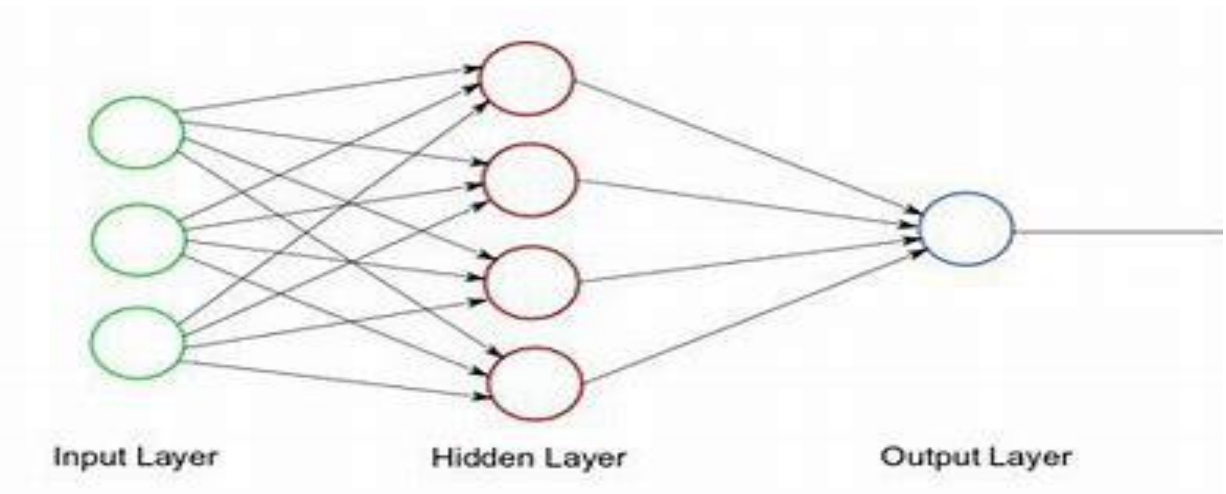


Figure 4.1 Convolutional Neural Network

Convolutional neural networks (CNN) are the one of the major components of neural networks. It contains neurons with learning weights and prejudices. Every neuron receives multiple inputs and takes a weighted sum over them where it transmits an activation function and responds with an output again. The CNN usually depends on three layers, Convolutional Layer, Pooling Layer and Dense Layer (Fully Connected Neural Network). The neuron of the human brain and every neuron does some work. In the same way, the CNN layer has performed its contribution when it comes to classifying an image. The image directly in the CNN instead of converting it into the 2D array or any standard dimension. The data we use for training, such as if we utilize some pictures of employee as training data, then CNN creates a filter based on the features (e.g., eyes, nose, ear, and so on) of each image that helps to detect an image. It is fast and simple to understand. It is the most accurate of all image predicting algorithms. Since CNN has three layers (Convolutional Layer, Pooling Layer and Dense Layer) so let's see how the Convolutional Layer works. Convolutional Layer: In convolutional neural networks, the main building blocks are convolutional layers. Convolutional Layer is the basic process of applying a filter to an input to produce an activation. As a consequence, extremely unique features appear on input images that can be identified anywhere. The CNN first create a filter based on all the features in the input images that allows for image detection. The pooling layer goal is to gradually shrink the spatial size of the representation in order to reduce the number of parameters and computation in the network. Each function map is treated individually by the pooling layer. There are two forms of pooling levels: average maximum pooling and maximum pooling, but maximum pooling is the most common. The pooling layer diminishes the parameters for which the overfitting of the model is decreased.

Each neuron in a layer receives information from all the neurons in the previous layer, making them densely connected. In other words, the dense layer is a completely connected layer, which means that all the neurons in a layer are connected to the next parts and the input image is classified through this layer. Next, we have to define activation functions like Sigmoid etc. The Activation function depends on data such as SoftMax can be used for the case of multiclass classification and Sigmoid for binary classification.

4.3.2 MINI-XCEPTION

Mini-Xception in CNN is one the most mainstream technique used for image classification. This model is used for emotion classification in the stress detection and reduction system. This is done by capturing the video using webcam and then converting it into images. Then work with images to identify the emotion is done with the help of mini-Xception. The system provides remedies to those employees based on the recorded stress level of the employees.

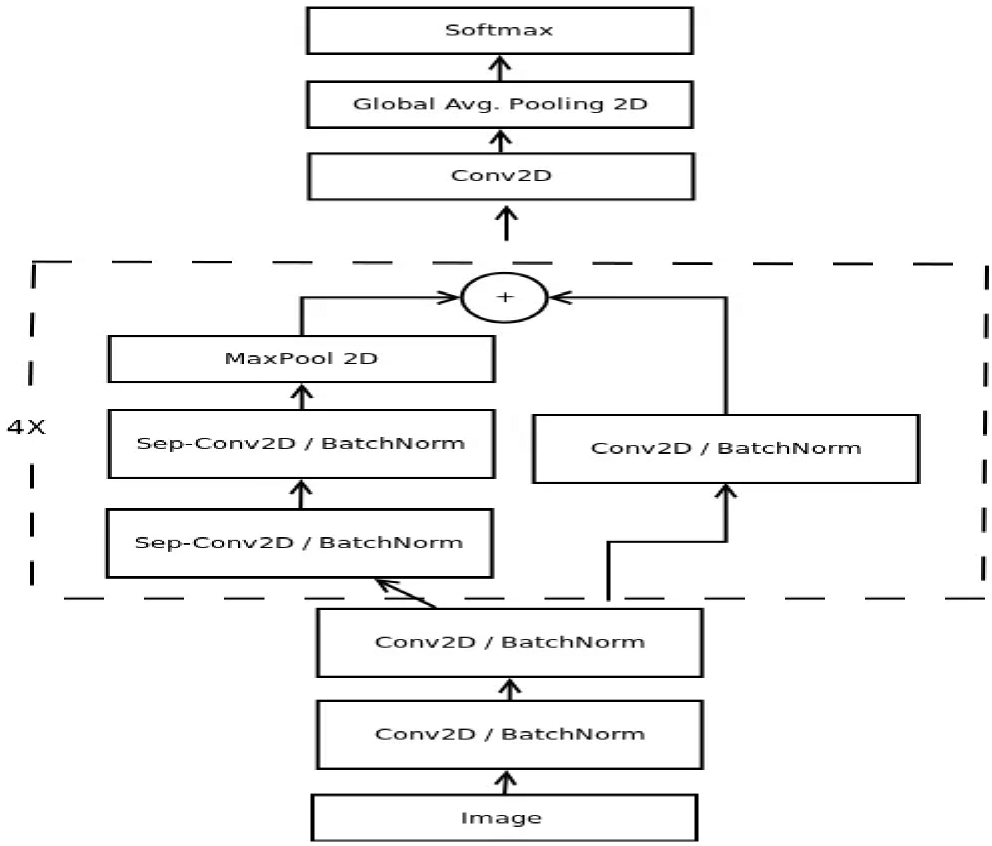


Figure:4.2 mini-Xception architecture for emotion classification

4.4 ALGORITHMS OF OVERALL PROCESS

The overall system is divided into four modules:

MODULE 1: LOGIN MODULE

- The system allows both the employee and admin to login.

- The admin is provided with the provision of viewing the stress data of the employees from the database.

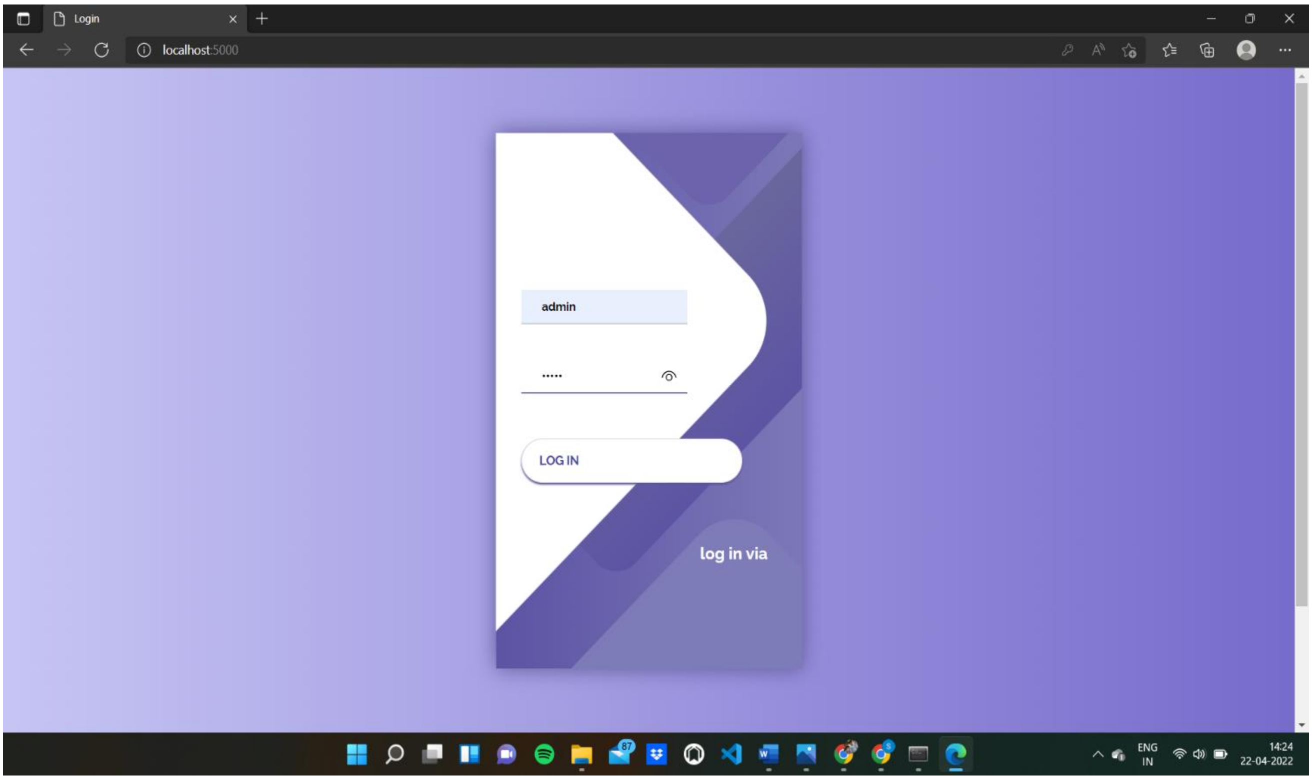


Figure: 4.3 Admin Login Page

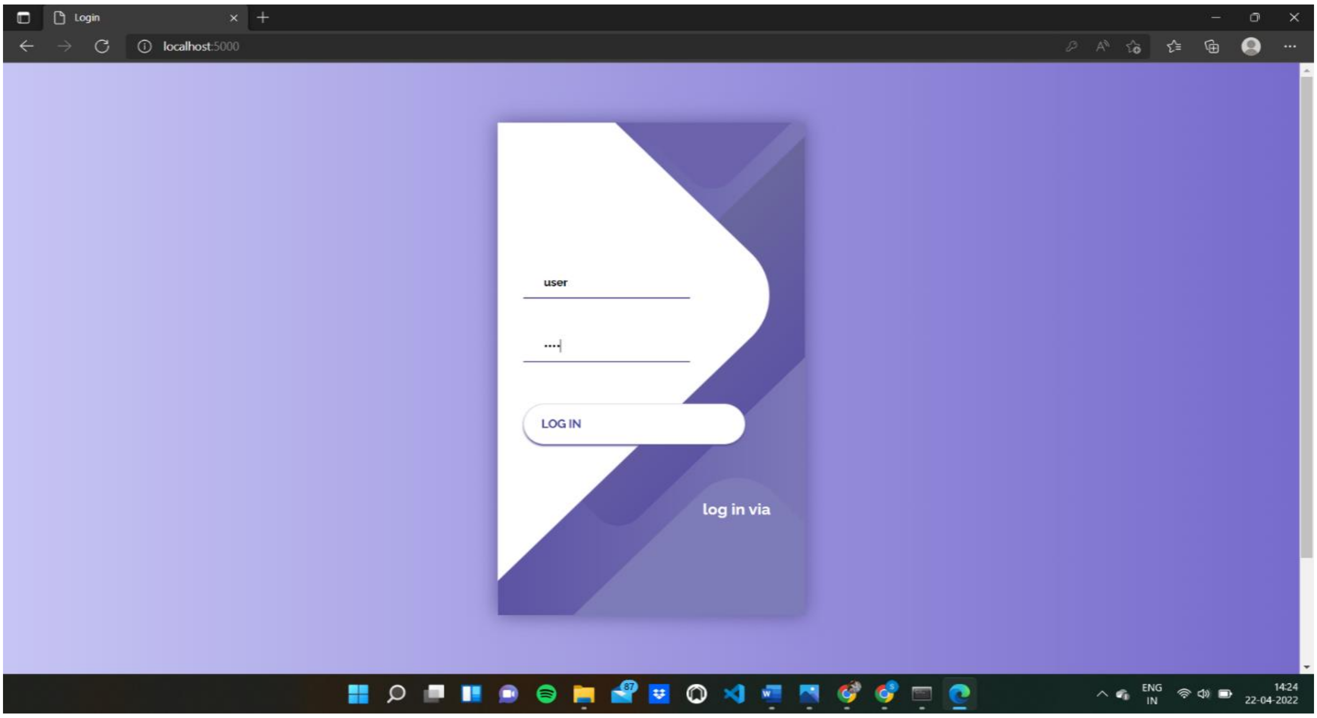


Figure: 4.4 User Login Page

MODULE 2: STRESS DETECTION MODULE

- This module accepts input through real-time video or through pre-recorded video.

- The input is then analyzed to classify whether an employee is stressed or not.

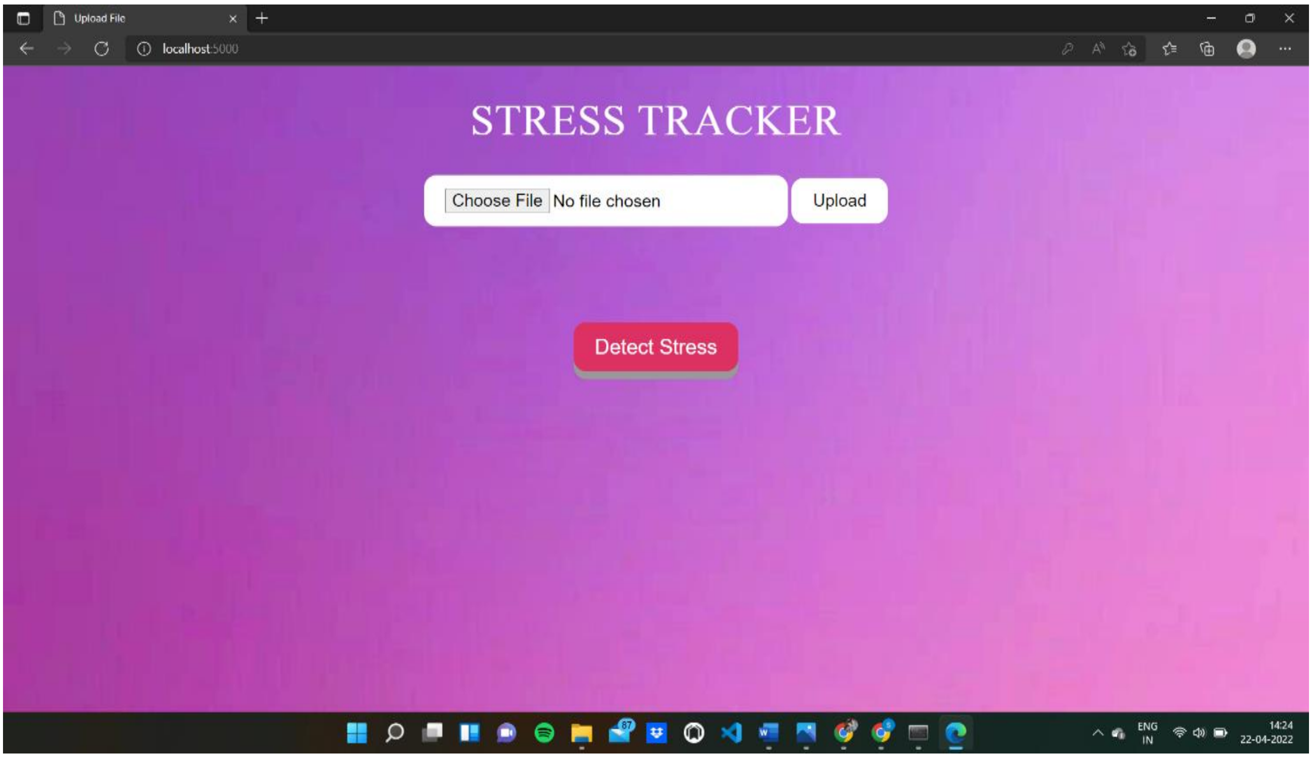


Figure. 4.5 Stress Detection Page

MODULE 3: STRESS EVALUVATION MODULE

- This module is responsible for calculating the stress level of the employee in case they are stressed.

- The stress level is calculated through displacement of eyebrow from its mean position and by providing them with survey form.

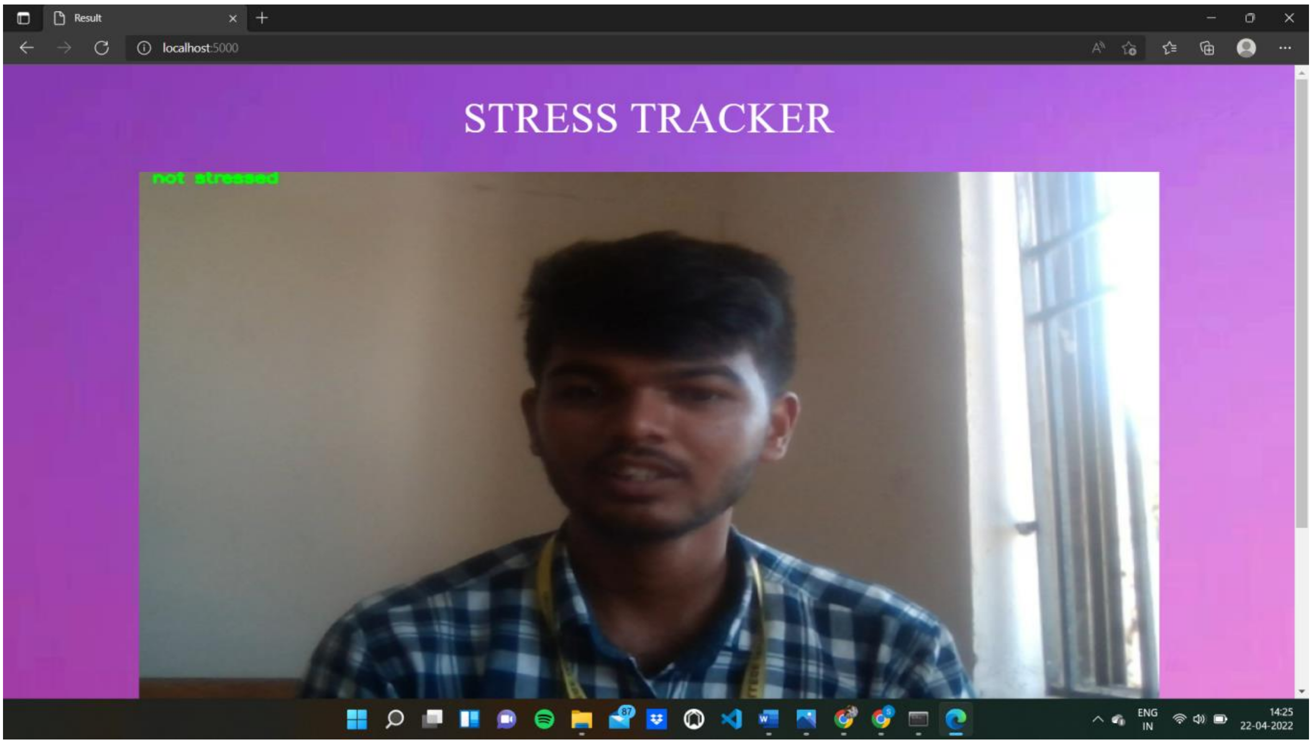
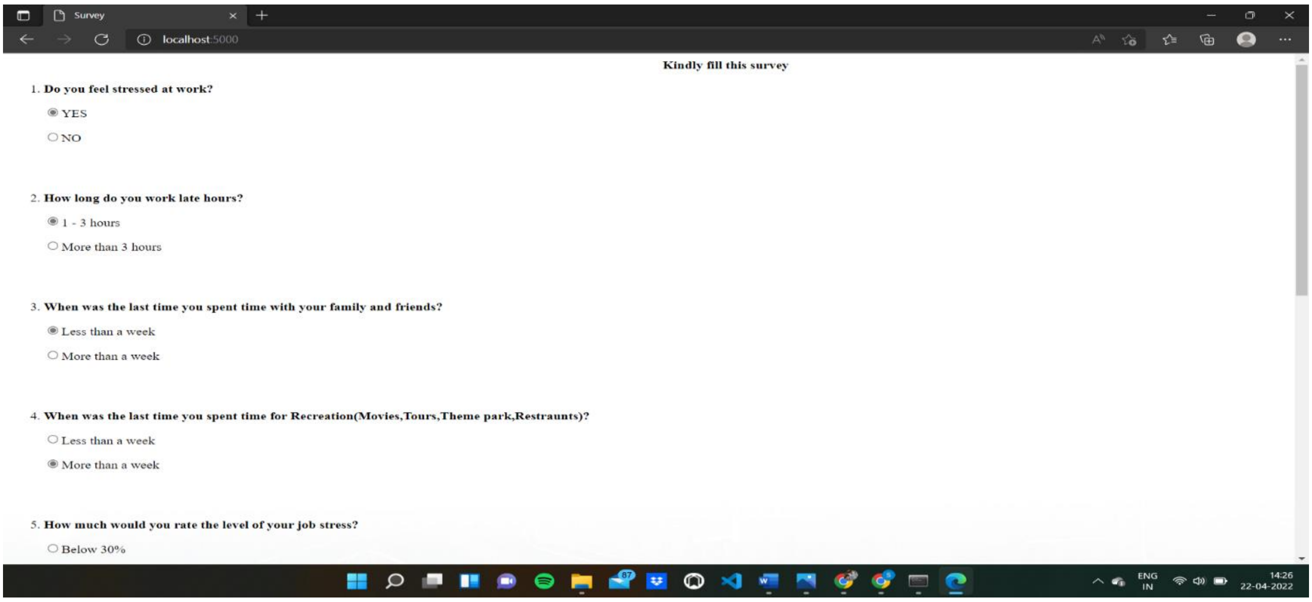


Figure. 4.6 Stress Detection

MODULE 4: STRESS MITIGATION MODULE

- Based on the obtained stress level, the system provides remedies to the employees.
- The remedy appropriate to the stress level is provided.



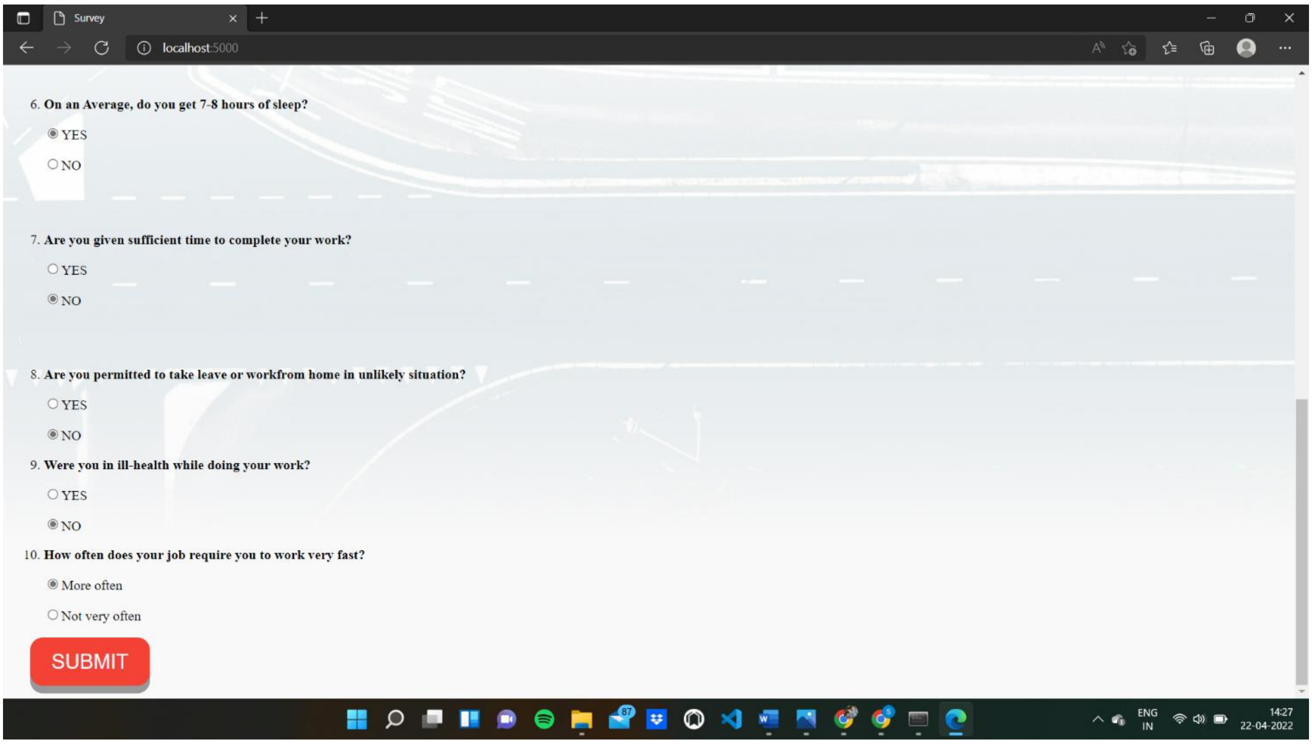


Figure. 4.7 Survey Form

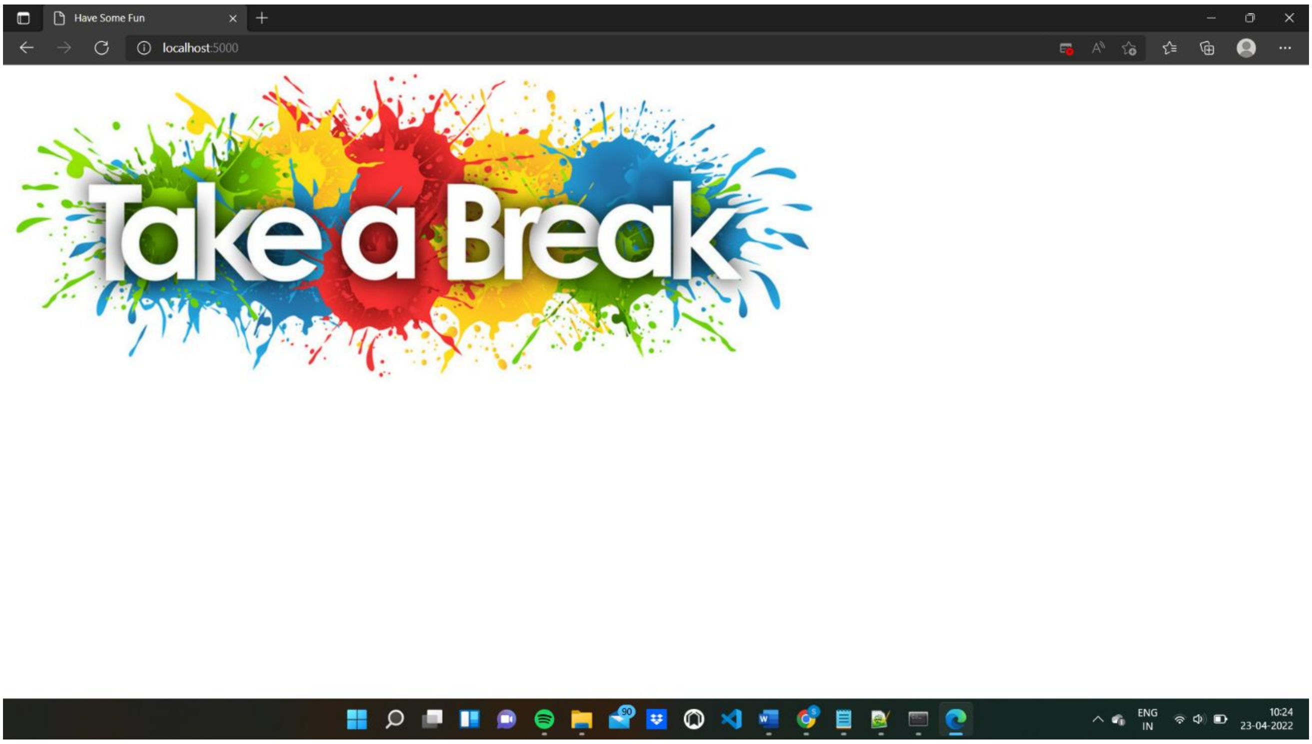


Figure. 4.8 Take a break remedy

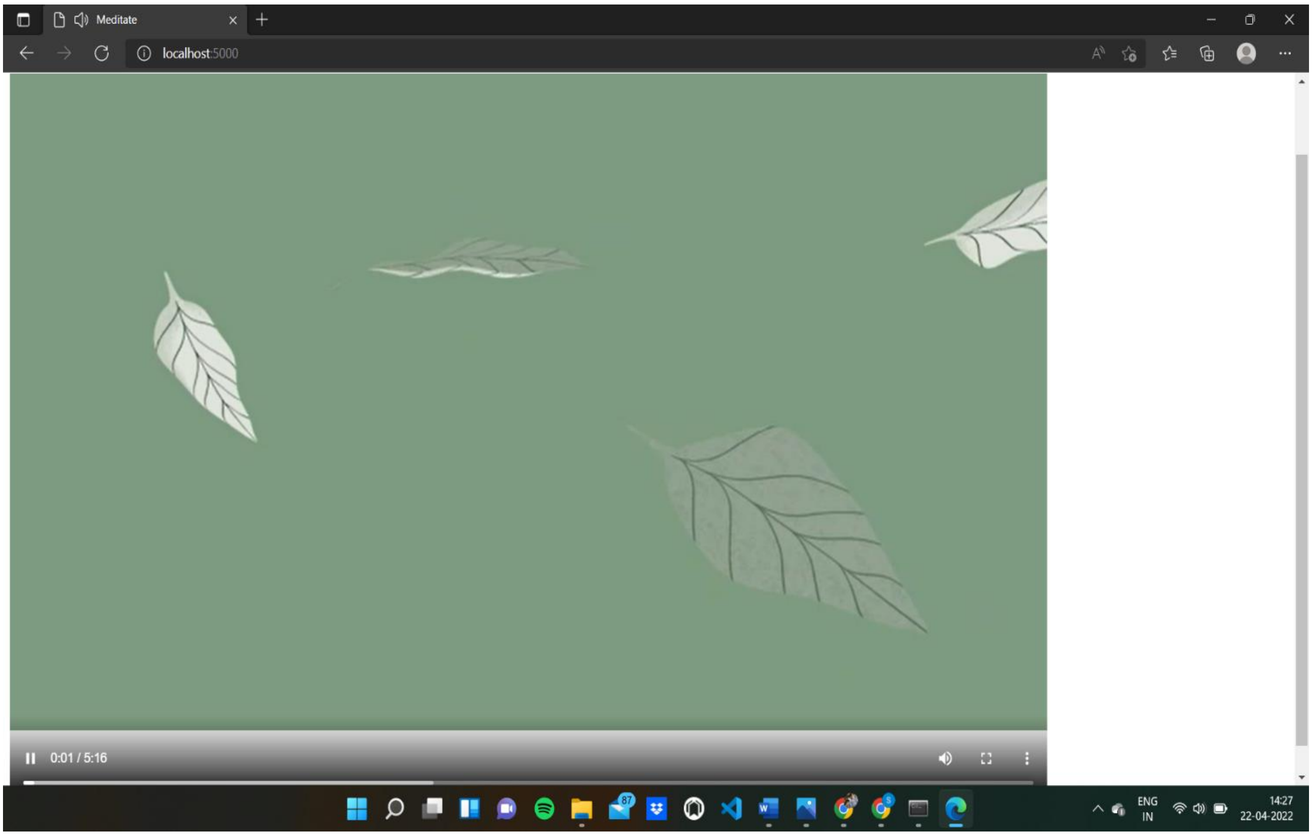


Figure. 4.9 Meditation video Remedy Page

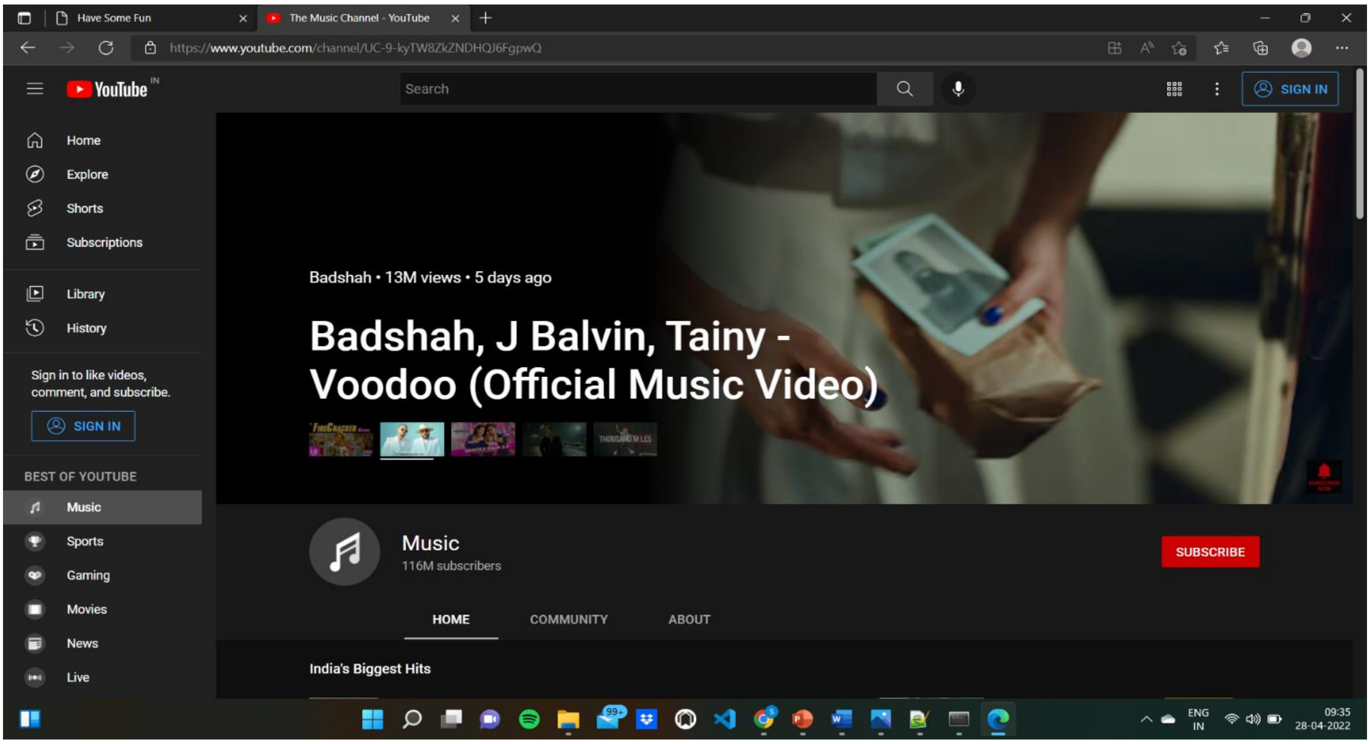


Figure 4.10 YouTube Access

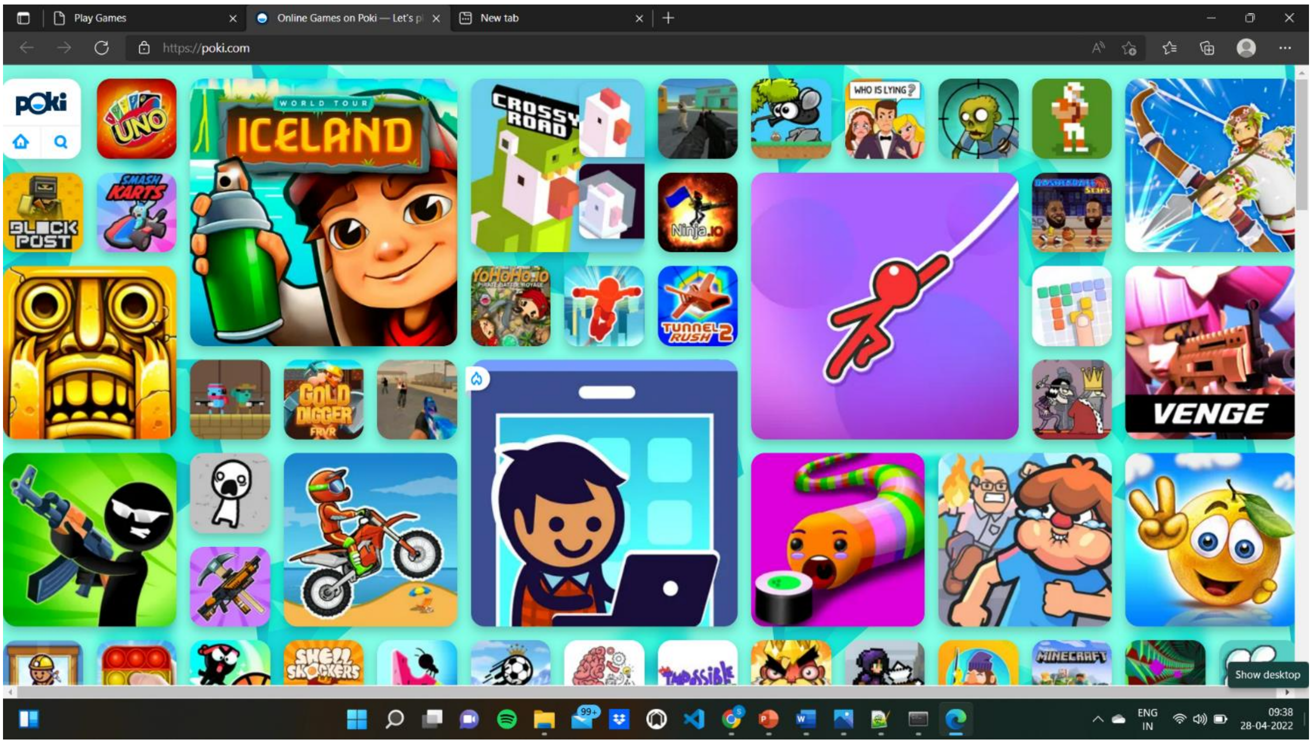


Figure 4.10 Access to Online Gaming Platform

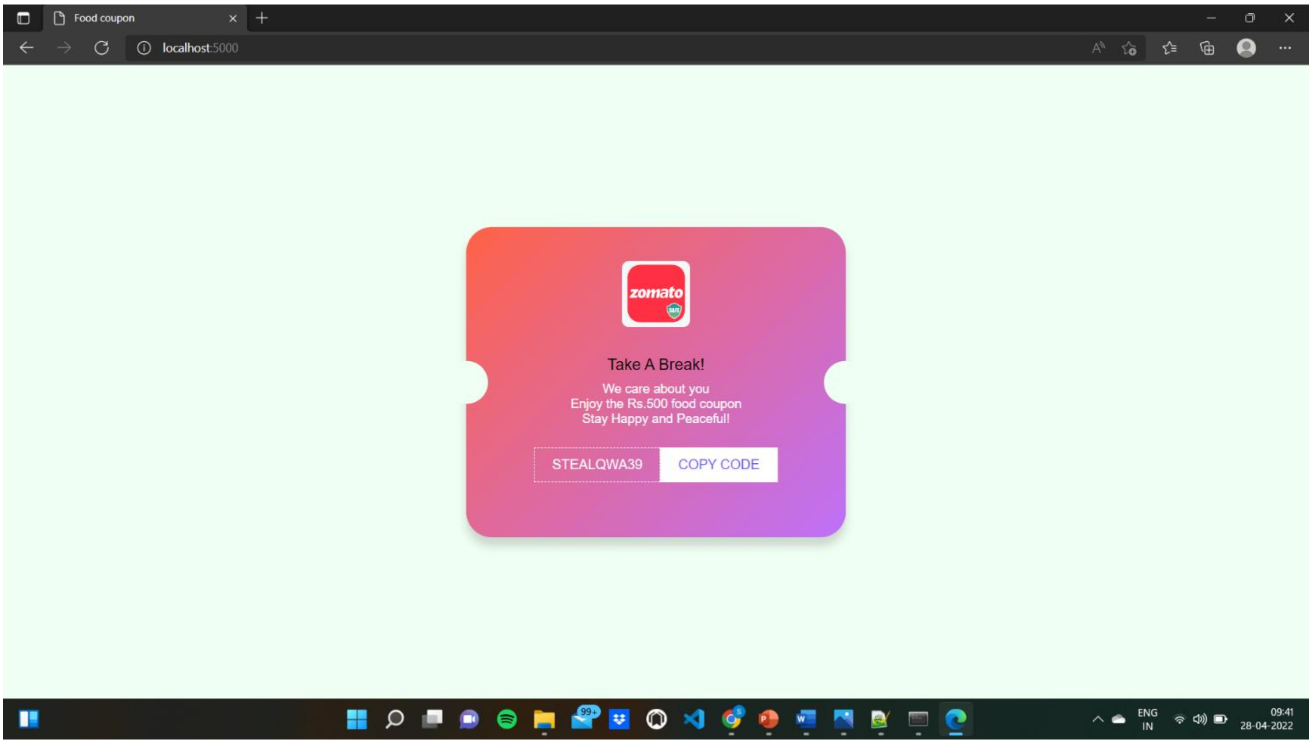


Figure 4.11 Food Coupon

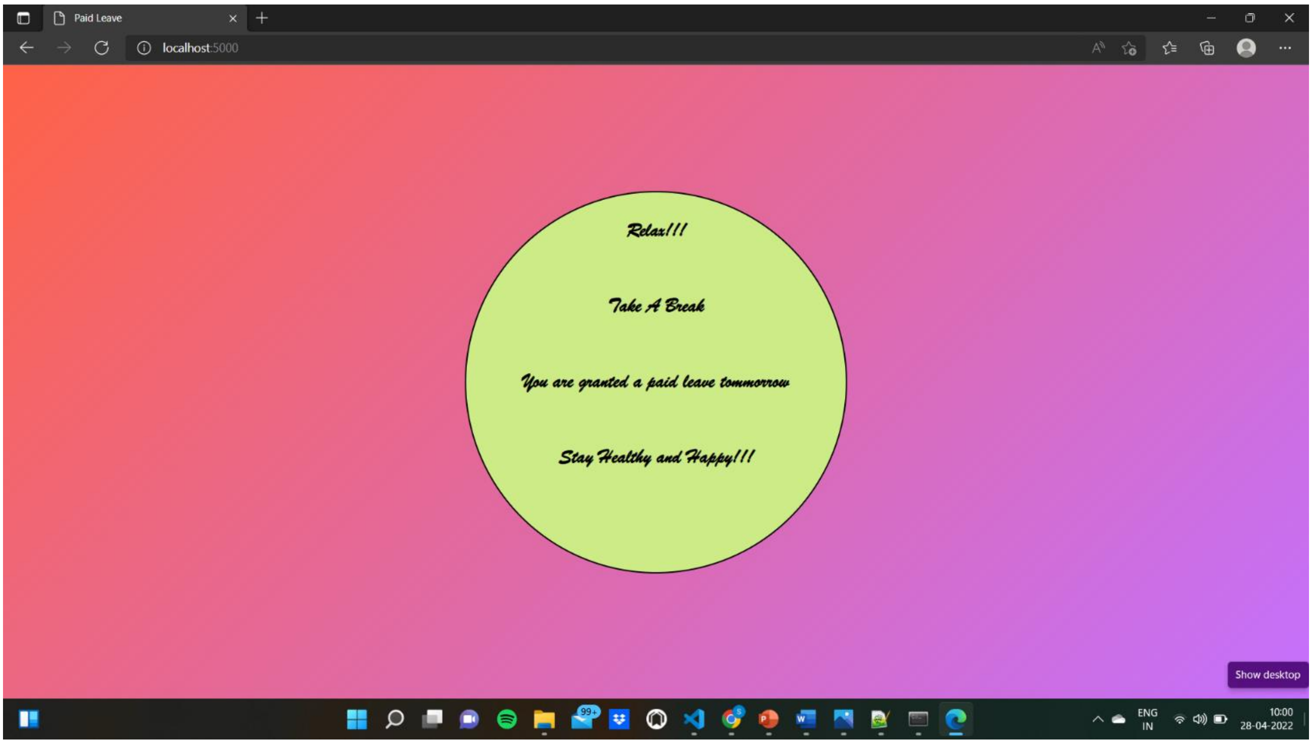


Figure 4.12 Paid Leave

4.5 MODULE WORKFLOW

The login page accepts only the employee and admin. The admin is provided with a special provision to view the previous stress levels recorded in the database. After login the input can be provided either through a pre-recorded video or at real-time.

The Stress detection module checks if the employee or admin is stressed or not. In case the employee or admin is stressed the stress evaluation module calculates stress level through the eyebrow displacement and through the response form the survey form. Based on the obtained stress level the system provides remedies to them.

4.6 CONCLUSION

The system provides the remedy to the stressed employee based on their stress level and gave them to survey form and answer for those questions it is in yes or no format and gave them the remedy based on the project.

CHAPTER 5

PERFORMANCE ANALYSIS

5.1 INTRODUCTION

The model which is developed using mini-Xception model of the CNN algorithm and achieves higher accuracy when compared to the existing approaches. The proposed model achieves 95.6% accuracy which is higher than the existing models.

5.2 TESTING

Software testing is the process of verifying and evaluating the performance of the software and it is mainly used to check the outcome of the software whether it meets the specified business needs as mentioned in the Software Requirement Specification(SRS) document. By performing testing prior to the release of the product or software in the market we can prevent bugs in the software, it also helps in reducing development costs and it is mainly used to improve the performance of the model. By performing testing, errors in software can be identified which helps the organization to deliver better and improved version of the product to the customer. In the traditional methods Software testing has been separated from the rest of the development phase. It is usually performed in the later stage of the software development life cycle, mainly during the execution stage or after the product has been built. The tester usually performs testing sometimes just before the application goes to market. Since the testing is performed during the last stage of the product release the tester will not have enough time to test the product and even if defects are found, there may be little time for correcting the bugs or retesting. It is not the best practice to release the software on time without clearing the bugs, bugs needed 29 to be fixed but the test team may not deliver the product on time. To overcome this problem testing should be done in the earlier stages of development which helps the tester to resolve errors and it is also less time consuming. Many development teams now use a methodology known a continuous testing. It is part of a DevOps approach – where development and operations collaborate over the entire product life cycle. The aim is to accelerate software delivery while balancing cost, quality and risk. With this testing technique, teams don't need to wait for the software to be built before testing starts. They can run tests much earlier in the cycle to discover defects sooner, when they are easier to fix.

5.2.1 SYSTEM TESTING

Testing is a type of software testing that is performed on a complete integrated system to evaluate the compliance of the system with the corresponding requirements. In the integration testing each and individual components are tested separately, the components which pass the integration testing are taken as input for the system testing. The main goal of integration testing is to analyze the components individually and detect any irregularity or malfunctioning between the units that are integrated together. System testing is used for finding the defects within both the integrated units and the whole system. The result of system testing is the observed behavior of a component or a system when it is tested. System Testing is carried out on the whole system in the context of either system requirement specifications or functional requirement specifications or in the context of both. System testing is used to test the design and behavior of the system and also to find whether the system meets the expectations of the customer. System testing is performed to test the system beyond the boundaries mentioned in the Software Requirements Specification (SRS) document. 30 System Testing is basically performed by a testing team that is independent of the development team, testing is usually performed to test the quality of the system impartially. System testing comprises both functional and non-functional testing. System Testing is a type black-box testing. In black box testing the developer is unaware of how the testing is carried out and how the system works. It is a type of testing in which the internal structure or design or implementation of the item or component being tested is known to the user. System Testing is performed after the integration testing and before the acceptance testing.

5.2.1.1 SYSTEM TESTING PROCESS

System Testing is performed in the following steps:

● Test Environment Setup: First step is to create the test environment for the better quality testing

● Create Test Case: The second step is to define and generate a test case for the testing process.

● Create Test Data: Then we need to create and generate the data that needs to be tested.

● Execute Test Case: Once the test data and test cases are generated, execution of test cases will be performed.

● Defect Reporting: The system will be detected for defects.

● Regression Testing: In order to identify the side effects of the testing process regression testing is done

● Log Defects: Once the defects are identified, these defects will be fixed in this step.

● Retest: Retest is performed if the test is not successful then again we need to perform the test.

5.2.2 FUNCTIONAL TESTING

Functional testing is a type of software testing which is mainly used to verify the functionality of the software application, and to check whether the function is working according to the requirement specification. In functional testing, each function is tested by giving the value, determining the output, and verifying the actual output with the expected value. Functional testing is performed as a type of black-box testing which is mainly done to confirm that the functionality of an application or system behaves as we are expecting. It is done to verify the functionality of the application. Functional testing is also called as black-box testing, because it mainly focuses on testing the application specification rather than actual code. Tester has to perform testing only on the program rather than the actual system. It hides the implementation details of the application to the tester. The main goal of the functional testing is to check the primary function of the system, it is used to check the necessarily usable function, the flow of the process and screen GUI. Functional testing displays the error message if any malfunction occurs so that the user can easily navigate throughout the application.

5.2.2.1 FUNCTIONAL TESTING PROCESS

Functional testing is performed by the following steps

- The first step is the tester will perform the verification of the requirement specification in the software application.

- After analyzing all the requirements as mentioned in the requirement specification document the tester will make a plan accordingly.

- After planning the tests, the tester will design the test case.

● Once the test case is designed the tester will make a document of the traceability matrix.

● Once the test case is designed and documented, the tester will execute the test case design.

● Analysis of the coverage to examine the covered testing area of the application.

● Once the defects are identified, defect management will be performed to manage and to resolve the defect.

The main objective of functional testing is to check the functionality of the software system. It mainly focuses on:

● Basic Usability: Functional Testing is mainly used to check the usability testing of the system. It is used to check whether a user can navigate freely without any difficulty through screens and there is no difficulty in using the system.

● Accessibility: Functional testing checks the accessibility of the function.

● Mainline function: Functional testing is used to focus on testing the main feature and functionality of the system.

● Error Condition: Functional testing is also performed to check the error condition. It checks whether the error message is displayed.

5.3 EXPERIMENTAL RESULTS

The proposed system achieves an accuracy of 95.6% by training the model with the hockey dataset, movie dataset and violent flows dataset.

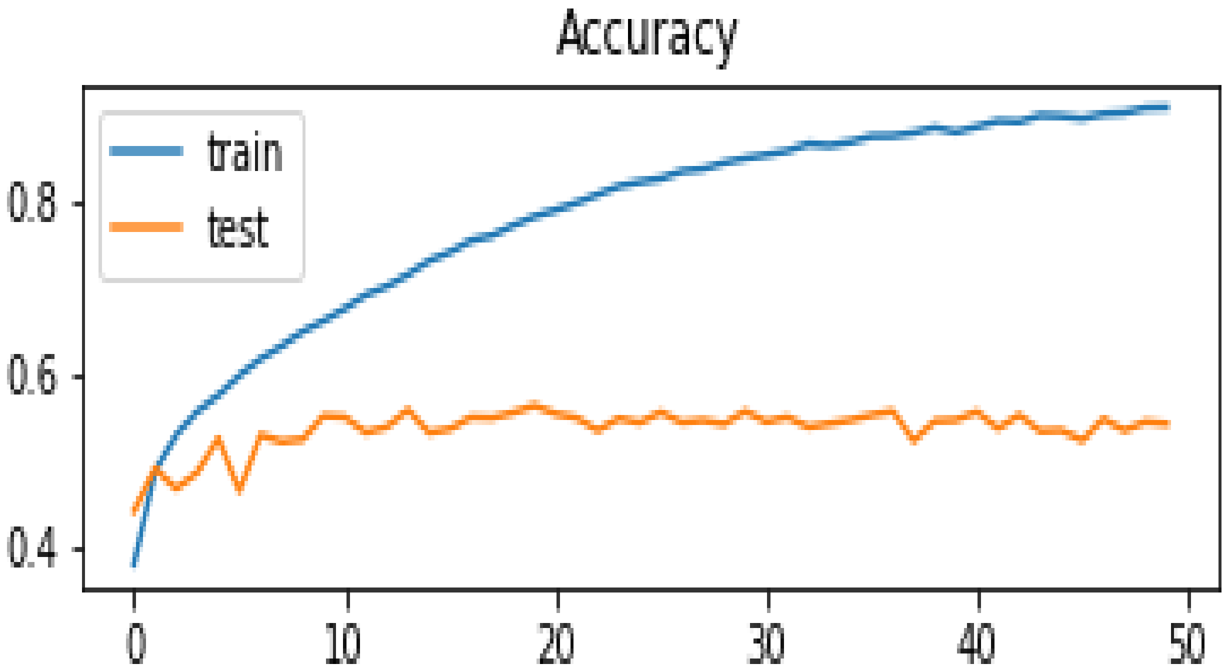


Figure 5.3.1: The graph shows the training and test accuracy of the proposed model in each epoch.

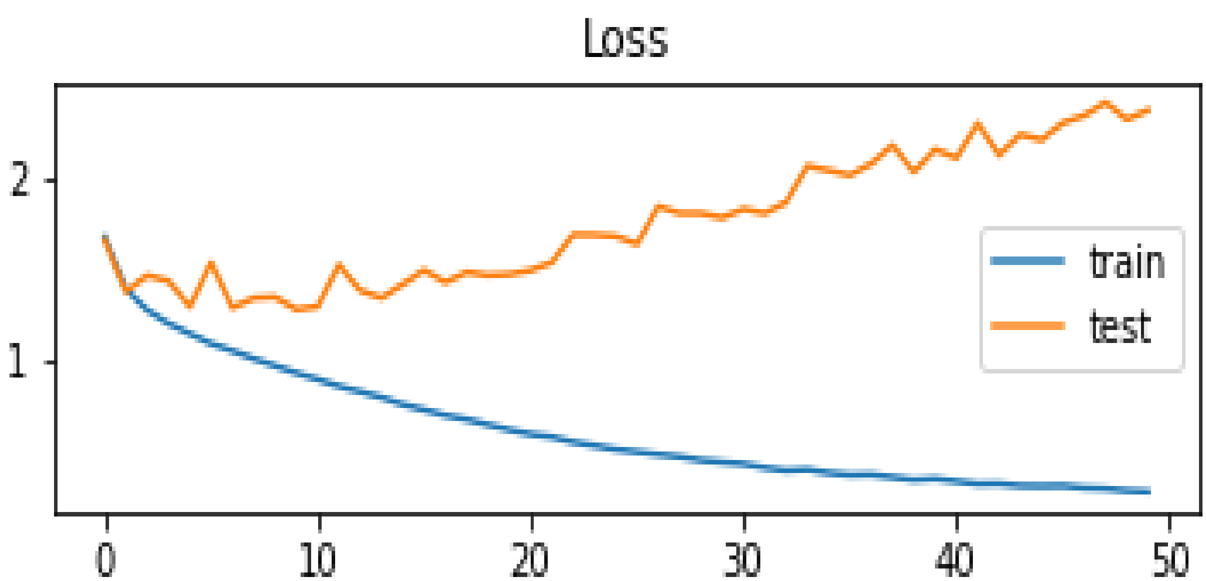


Figure 5.3.2 The graph indicates the cross-entropy loss for both the training and testing data set.

Model: "model_4"

Layer (type)	Output Shape	Param #	Connected to
input_5 (InputLayer)	[(None, 48, 48, 1)]	0	[]
conv2d_28 (Conv2D)	(None, 46, 46, 8)	72	['input_5[0][0]']
batch_normalization_56 (BatchNormalization)	(None, 46, 46, 8)	32	['conv2d_28[0][0]']
activation_24 (Activation)	(None, 46, 46, 8)	0	['batch_normalization_56[0][0]']
conv2d_29 (Conv2D)	(None, 44, 44, 8)	576	['activation_24[0][0]']
batch_normalization_57 (BatchNormalization)	(None, 44, 44, 8)	32	['conv2d_29[0][0]']
activation_25 (Activation)	(None, 44, 44, 8)	0	['batch_normalization_57[0][0]']
separable_conv2d_32 (SeparableConv2D)	(None, 44, 44, 16)	200	['activation_25[0][0]']
batch_normalization_59 (BatchNormalization)	(None, 44, 44, 16)	64	['separable_conv2d_32[0][0]']
activation_26 (Activation)	(None, 44, 44, 16)	0	['batch_normalization_59[0][0]']
separable_conv2d_33 (SeparableConv2D)	(None, 44, 44, 16)	400	['activation_26[0][0]']
batch_normalization_60 (BatchNormalization)	(None, 44, 44, 16)	64	['separable_conv2d_33[0][0]']
conv2d_30 (Conv2D)	(None, 22, 22, 16)	128	['activation_25[0][0]']
max_pooling2d_16 (MaxPooling2D)	(None, 22, 22, 16)	0	['batch_normalization_60[0][0]']
batch_normalization_58 (BatchNormalization)	(None, 22, 22, 16)	64	['conv2d_30[0][0]']
add_16 (Add)	(None, 22, 22, 16)	0	['max_pooling2d_16[0][0]', 'batch_normalization_58[0][0]']
separable_conv2d_34 (SeparableConv2D)	(None, 22, 22, 32)	656	['add_16[0][0]']
batch_normalization_62 (BatchNormalization)	(None, 22, 22, 32)	128	['separable_conv2d_34[0][0]']
activation_27 (Activation)	(None, 22, 22, 32)	0	['batch_normalization_62[0][0]']
separable_conv2d_35 (SeparableConv2D)	(None, 22, 22, 32)	1312	['activation_27[0][0]']

```
Conv2D)

batch_normalization_66 (BatchN  (None, 11, 11, 64)  256      ['separable_conv2d_37[0][0]']
ormalization)

conv2d_32 (Conv2D)             (None, 6, 6, 64)     2048     ['add_17[0][0]']

max_pooling2d_18 (MaxPooling2D  (None, 6, 6, 64)    0        ['batch_normalization_66[0][0]']
)

batch_normalization_64 (BatchN  (None, 6, 6, 64)    256      ['conv2d_32[0][0]']
ormalization)

add_18 (Add)                   (None, 6, 6, 64)     0        ['max_pooling2d_18[0][0]',
                                                              'batch_normalization_64[0][0]']

separable_conv2d_38 (Separable  (None, 6, 6, 128)   8768     ['add_18[0][0]']
Conv2D)

batch_normalization_68 (BatchN  (None, 6, 6, 128)   512      ['separable_conv2d_38[0][0]']
ormalization)

activation_29 (Activation)     (None, 6, 6, 128)    0        ['batch_normalization_68[0][0]']

separable_conv2d_39 (Separable  (None, 6, 6, 128)   17536    ['activation_29[0][0]']
Conv2D)

batch_normalization_69 (BatchN  (None, 6, 6, 128)   512      ['separable_conv2d_39[0][0]']
ormalization)

conv2d_33 (Conv2D)             (None, 3, 3, 128)    8192     ['add_18[0][0]']

max_pooling2d_19 (MaxPooling2D  (None, 3, 3, 128)   0        ['batch_normalization_69[0][0]']
)

batch_normalization_67 (BatchN  (None, 3, 3, 128)   512      ['conv2d_33[0][0]']
ormalization)

add_19 (Add)                   (None, 3, 3, 128)    0        ['max_pooling2d_19[0][0]',
                                                              'batch_normalization_67[0][0]']

conv2d_34 (Conv2D)             (None, 3, 3, 7)      8071     ['add_19[0][0]']

global_average_pooling2d_4 (Gl  (None, 7)           0        ['conv2d_34[0][0]']
obalAveragePooling2D)

predictions (Activation)       (None, 7)            0        ['global_average_pooling2d_4[0][0
                                                              ]']

==================================================================================================
Total params: 58,423
Trainable params: 56,951
Non-trainable params: 1,472
__________________________________________________________________________________________________
```

	precision	recall	f1-score	support
surprise	0.35	0.54	0.42	991
fear	0.65	0.32	0.42	111
angry	0.36	0.47	0.40	1042
neutral	0.75	0.79	0.77	1787
sad	0.50	0.29	0.37	1231
disgust	0.66	0.69	0.67	752
happy	0.57	0.38	0.46	1264
accuracy			0.53	7178
macro avg	0.55	0.50	0.50	7178
weighted avg	0.55	0.53	0.53	7178

Figure 5.3.3 The above figure indicates the performance for each emotion

Figure 5.3.4 The above figure indicates the confusion matrix of emotions

5.4 CONCLUSION

The experimental results discussed above depicts the accuracy of our proposed model. The accuracy best describes oru model when compared to the existing approaches. We have used cross entropy loss as our loss function to calculate the loss of our model.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

Capturing the video and Classification of images and learning of image processing techniques is done in this project. Also, the scheme through which project is achieved is **Convolutional Neural Network** scheme. The result which was got was correct up to more than 95% of the cases, but it would be improved at the end with the help of more graphics. This work is basically focused on stress detection and management that enables that enables a healthy work environment. The method proposed gave efficient and effective result both for reduction and detection.

6.2 FUTURE SCOPE OF THE PROJECT

Now-a-days recent advancement in technologies has pushed the limits further for man to get rid of older equipment which posed inconvenience in using. The system prioritizes and gives the remedy to them. Though there is significant progress in the identification of facial emotions in the past decade, there are some outstanding issues and new avenues exist for future development.

In addition to identifying stress through facial emotions, one can develop it further to identifying stress along with one's body posture which provides much more accurate results.

The extraction of facial emotion heavily depends on the ability to detect facial feature points, the semantic location of which has been predefined. It would therefore be useful in the future to design an end-to-end model capable of both learning the movement of face during facial-emotion recognition and detecting facial feature points. In this way, one could arrive at a more accurate system.

APPENDIX

```
from scipy.spatial import distance as dist
from imutils.video import VideoStream
from imutils import face_utils
import numpy as np
import imutils
import time
import dlib
import cv2
import matplotlib.pyplot as plt
from keras.preprocessing.image import img_to_array
from keras.models import load_model
import pandas   # changed 10/4/22
import webbrowser   #changed today
import form_ex
import sqlite3
import datetime
import os
from sys import argv
from flask import Flask, flash, request, redirect,
Url_for,send_from_directory,render_template,Response, jsonify
from werkzeug.utils import secure_filename
from flask_cors import CORS

detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
emotion_classifier = load_model("_mini_XCEPTION.102-0.66.hdf5",
compile=False)
points=[]
argss = './data/video/test.mp4'
comp_stress = 0
score = 0
final_stress_value = 0
form_status = 0
```

```
submittion_status= -1
user_details = ""
final_stress_value_final = 0

def eye_brow_distance(leye,reye):
    distq = dist.euclidean(leye,reye)
    points.append(int(distq))
    return distq

def emotion_finder(faces,frame):
    global emotion_classifier
    EMOTIONS = ["angry" ,"disgust","scared", "happy", "sad", "surprised","neutral"]
    x,y,w,h = face_utils.rect_to_bb(faces)
    frame = frame[y:y+h,x:x+w]
    roi = cv2.resize(frame, (64,64))
    roi = roi.astype("float") / 255.0
    roi = img_to_array(roi)
    roi = np.expand_dims(roi,axis=0)
    preds = emotion_classifier.predict(roi)[0]
    emotion_probability = np.max(preds)
    label = EMOTIONS[preds.argmax()]
    if label in ['scared','sad']:
        label = 'stressed'
    else:
        label = 'not stressed'
    return label

def normalize_values(points,disp):
    normalized_value = abs(disp - np.min(points))/abs(np.max(points) - np.min(points))
    stress_value = np.exp(-(normalized_value))
    if stress_value>=75:
        return stress_value,"High Stress"
    else:
        return stress_value,"low_stress"
```

```
def main(arg):
    global submittion_status
    cap = cv2.VideoCapture(arg)
    flag = 0
    while(flag <= 0):
        _, frame = cap.read()
        if arg == 0:
            frame = cv2.flip(frame,1)

        frame = imutils.resize(frame,width=800,height=800)
        gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
        detections = detector(gray,0)
        for detection in detections:
            emotion = emotion_finder(detection,gray)
            cv2.putText(frame, emotion, (10,10),cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(0, 255, 0), 2)

            if emotion == 'stressed':
                print(emotion)
                comp_stress = calc_stress(cap,frame)
                print("This is the final stress level", comp_stress)
                flag = 1

def calc_stress(cap,frame):
    global comp_stress
    global points
    capture_duration = time.time() + 10
    while time.time() < capture_duration:
        (lBegin, lEnd) = face_utils.FACIAL_LANDMARKS_IDXS["right_eyebrow"]
        (rBegin, rEnd) = face_utils.FACIAL_LANDMARKS_IDXS["left_eyebrow"]
        gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
        detections = detector(gray,0)
        for detection in detections:
            emotion = emotion_finder(detection,gray)
            #cv2.putText(frame,    emotion,    (10,10),cv2.FONT_HERSHEY_SIMPLEX,
```

```
0.5, (0, 255, 0), 2)
            shape = predictor(frame,detection)
            shape = face_utils.shape_to_np(shape)

            leyebrow = shape[lBegin:lEnd]
            reyebrow = shape[rBegin:rEnd]

            reyebrowhull = cv2.convexHull(reyebrow)
            leyebrowhull = cv2.convexHull(leyebrow)

            cv2.drawContours(frame, [reyebrowhull], -1, (0, 255, 0), 1)
            cv2.drawContours(frame, [leyebrowhull], -1, (0, 255, 0), 1)

            distq = eye_brow_distance(leyebrow[-1],reyebrow[0])
            if(len(points)) > 1:
                stress_value,stress_label = normalize_values(points,distq)
                print("stress",stress_value)
                if stress_value > comp_stress:
                    comp_stress = int(stress_value * 100)

    if comp_stress > 50:
        comp_stress = comp_stress - 50
        return comp_stress

if __name__ == '__main__':
    app.run(debug=True,host="0.0.0.0",use_reloader=False)
```

REFERENCES

[1] B. Vimaleswaran and G. S. Ratnayake, "E - Therapy Improvement Monitoring Platform for Depression using Facial Emotion Detection of Youth," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021, pp. 950-955, doi: 10.1109/ICICV50876.2021.9388447.

[2] M. R. Koujan, L. Alharbawee, G. Giannakakis, N. Pugeault and A. Roussos, "Real-time Facial Expression Recognition In the Wild" by Disentangling 3D Expression from Identity," 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020), 2020, pp. 24-31, doi: 10.1109/FG47880.2020.00084.

[3] A. Verma, P. Singh and J. S. Rani Alex, "Modified Convolutional Neural Network Architecture Analysis for Facial Emotion Recognition," 2019 International Conference on Systems, Signals and Image Processing (IWSSIP), 2019, pp. 169-173, doi: 10.1109/IWSSIP.2019.8787215.

[4] Yue Lu, Xinsha Fu, Enqiang Guo, Feng Tang "XGBoost Algorithm-Base Monitoring Model for Urban Driving Stress: Combining Driving Behaviour, Driving Environment, and Route Familiarity", January 2021.

[5] X. Ben et al., "Video-based Facial Micro-Expression Analysis: A Survey of Datasets, Features and Algorithms," in IEEE Transactions on Pattern Analysis and Machine Intelligence, doi: 10.1109/TPAMI.2021.3067464.

[6] G. Giannakakis, M. R. Koujan, A. Roussos and K. Marias, "Automatic stress detection evaluating models of facial action units," 2020 15th IEEE International

Conference on Automatic Face and Gesture Recognition (FG 2020), 2020, pp. 728-733, doi: 10.1109/FG47880.2020.00129.

[7] W. Carneiro de Melo, E. Granger and A. Hadid, "A Deep Multiscale Spatiotemporal Network for Assessing Depression from Facial Dynamics," in IEEE Transactions on Affective Computing, doi: 10.1109/TAFFC.2020.3021755.

[8] A. Tazarv, S. Labbaf, S. M. Reich, N. Dutt, A. M. Rahmani and M. Levorato, "Personalized Stress Monitoring using Wearable Sensors in Everyday Settings," 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), 2021, pp. 7332-7335, doi: 10.1109/EMBC46164.2021.9630224.

[9] Ryota R. Mitsuhashi et al., "Video-Based Stress Level Measurement Using Imaging Photoplethysmography," 2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), 2019, pp. 90-95, doi: 10.1109/ICMEW.2019.0-105.

[10] G. -M. Lin, M. Nagamine, S. -N. Yang, Y. -M. Tai, C. Lin and H. Sato, "Machine Learning Based Suicide Ideation Prediction for Military Personnel," in IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 7, pp. 1907-1916, July 2020, doi: 10.1109/JBHI.2020.2988393.

[11] S. Kumar, D. Varshney, G. Dhawan and H. Jalutharia, "Analyzing the Effective Psychological State of Students using Facial Features," 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), 2020, pp. 648-653, doi: 10.1109/ICICCS48265.2020.9120909.

[12] D. Girardi, F. Lanubile, N. Novielli and A. Serebrenik, "Emotions and Perceived Productivity of Software Developers at the Workplace," in IEEE Transactions on Software Engineering, doi: 10.1109/TSE.2021.3087906.

[13] A. Greco et al., "Acute stress state classification based on electrodermal activity modeling," in IEEE Transactions on Affective Computing, doi: 10.1109/TAFFC.2021.3055294.

[14] P. Arpaia, N. Moccaldi, R. Prevete, I. Sannino and A. Tedesco, "A Wearable EEG Instrument for Real-Time Frontal Asymmetry Monitoring in Worker Stress Analysis," in IEEE Transactions on Instrumentation and Measurement, vol. 69, no. 10, pp. 8335-8343, Oct. 2020, doi: 10.1109/TIM.2020.2988744.

[15] C. Bustos, N. Elhaouij, A. Solé-Ribalta, J. Borge-Holthoefer, A. Lapedriza and R. Picard, "Predicting Driver Self-Reported Stress by Analyzing the Road Scene," 2021 9th International Conference on Affective Computing and Intelligent Interaction (ACII), 2021, pp. 1-8, doi: 10.1109/ACII52823.2021.9597438.