# A

# Project Stage I report on

## "Stress Detection in Person by using Image processing and Machine Learning"

Submitted By:


**Tejaswini Anil Aher [B190582001]**
**Vaishnavi Anil Bhosale [B190582006]**
**Pooja Ganpat Gawali [B190582017]**
**Gayatri Vinod Ushir [B190582069]**


**B.E. [AI & DS Engineering]**

## 2023-24

Guided By:

**Mr. Chetan Patil**


**Department of Artificial Intelligence and Data Science Engineering**

**Matoshri College of Engineering and Research Centre, Odhagaon Nashik-422105**

**Savitribai Phule Pune University**

# Matoshri College of Engineering and Research Centre, Nashik

## Certificate

*This is to certify that Project stage I report entitled*

***"Stress Detection In Person By Using Image Processing And Machine Learning"***

***Aher Tejaswini Anil*** ***Bhosale Vaishnavi Anil***
***Gawali Pooja Ganpat*** ***Ushir Gayatri Vinod***

*Is a record of bonafide work carried out by us, under our guidance, in partial fulfillment of the requirement for the award of Degree of Bachelor of Engineering (Artificial Intelligence and Data Science) at Matoshri College of Engineering and Research Centre, Nashik under the Savitribai Phule Pune University, Pune.*

Date: / / 2023

.......................... ..............................

**Mr. Chetan Patil** **Prof. M.T. Dhande**
**Project Guide** **Project Coordinator**

.......................... ..............................

**Dr. J.J. Chopade** **Dr. G. K. Kharate**
**Head of Department** **Principal**

**EXAMINATION**

........................ ..........................

**Sign of External 1** **Sign of External 2**

# *Acknowledgement*

It is a privilege for us to have been associated with **Mr. Chetan Patil** , my guide, during this project work. I have been greatly benefited by his valuable suggestions and ideas. It is with great pleasure that I express our deep sense of gratitude for his valuable guidance, constant encouragement and patience throughout this work.

I express our gratitude to **Dr. J. J. Chopade**, Head,Department of Computer Engineering well as to **Prof M. T. Dhande [Project Co-ordinator]** for his constant encouragement, co-operation, and support.

I take this opportunity to thank all our classmates for their company during the course work and for useful discussion I had with them.

I would be failing in our duties if I do not make a mention of our family members including our parents for providing moral support, without which this work would not have been completed.

**Tejaswini Anil Aher [B190582001]**
**Vaishnavi Anil Bhosale[B190582006]**
**Pooja Ganpat Gawali [B190582017]**
**Gayatri Vinod Ushir [B190582069]**

Abstract

This stress detection project aims to develop a system for accurately identifying and quantifying stress levels in individuals using physiological and behavioral data. By leveraging wearable sensors, machine learning algorithms, and real-time monitoring, the project offers a valuable tool for early stress intervention and improved well-being. The system's potential applications include healthcare, workplace stress management, and personal stress awareness, ultimately contributing to better mental and physical health outcomes.

**Keywords:** Stress detection, Machine learning, Healthcare, Physiological.

# Contents

# Chapter 1

# Introduction

## 1.1 Problem Defination

The problem definition for a stress detection project is to develop a system or technology that can accurately and non-invasively identify and measure the level of stress in individuals based on various physiological and/or behavioral cues, such as heart rate, skin conductance, facial expressions, speech patterns, or other relevant indicators.

## 1.2 Objective

- The objective of a stress detection project is to develop a system or method that can accurately identify and quantify stress levels in individuals, typically through physiological, behavioral, or psychological indicators.
- To identify the impact of depression and other serious mental illnesses
- To increase awareness of the prevalence and consequences of untreated depression in the older adult.
- To define mood and affect.

## 1.3 Project Scope

The scope of a stress detection project involves developing a system or tool to monitor, assess, and potentially mitigate stress in individuals. This typically includes:

1. Data collection: Gathering physiological and/or behavioral data (e.g., heart rate, skin conductance, speech patterns) from participants.

2. Data analysis: Using algorithms and machine learning techniques to process and interpret the collected data.

3. Stress detection: Developing a model to identify and quantify stress levels in individuals based on the analyzed data.

4. Feedback or intervention: Implementing strategies for providing feedback or interventions to help individuals manage or reduce their stress.

5. User interface: Creating a user-friendly interface for participants to interact with the system.

6. Evaluation and validation: Testing the system's accuracy and effectiveness in stress detection and management.

7. Ethical considerations: Ensuring the project adheres to ethical guidelines and data privacy regulations.

8. Implementation context: Defining the intended application areas, such as healthcare, workplace wellness, or personal well-being.

9. Project timeline and resources: Outlining the schedule and required resources for project completion.

10. Project goals and objectives: Clearly defining what the project aims to achieve and its expected outcomes.

## 1.4 Dependency

A stress detection project typically requires the following dependencies:

1. Data Collection: Gather physiological and behavioral data, such as heart rate, skin conductance, and facial expressions.

2. Sensors and Wearables: Hardware components like sensors, fitness trackers, or physiological monitors to capture data.

3. Data Processing: Software tools and libraries for data preprocessing and feature extraction.

4. Machine Learning Algorithms: Utilize algorithms for stress prediction, such as classification models or deep learning techniques.

5. Programming Languages: Commonly use Python for data analysis and machine learning.

6. Data Storage: A database or file storage for managing and organizing collected data.

7. User Interface: Develop a user interface or application for real-time stress monitoring or feedback.

8. Cloud Computing: If handling large datasets, cloud services can be useful for storage and computation.

9. Documentation and Reporting Tools: Tools for documenting and visualizing project results.

10. Ethical Considerations: Address privacy and ethical concerns associated with stress data collection and analysis.

11. Testing and Validation: Methods for evaluating the accuracy and effectiveness of stress detection models.

12. Collaboration and Communication Tools: Software for team collaboration and project management.

13. Domain Knowledge: Understand the psychology and physiology of stress for model development.

14. Regulatory Compliance: Ensure compliance with relevant data protection and ethical regulations.

The specific dependencies may vary based on project goals, data sources, and intended applications.

## 1.5 Strategy Plan To solve Problem

Creating a stress detection project involves several key steps. Here's a short strategy plan:

1. Define Objectives: - Clearly outline the project's goals and objectives. - Specify the target audience or users.

2. Research and Data Collection: - Gather relevant data on stress indicators (physiological, behavioral, or self-reported). - Consider using sensors, surveys, or social media data.

3. Data Processing: - Preprocess and clean the collected data. - Explore feature engineering and selection for stress prediction.

4. Algorithm Selection: - Choose appropriate machine learning or AI algorithms for stress detection. - Consider supervised or unsupervised methods.

5. Model Training: - Train and validate your chosen model using labeled data. - Fine-tune hyperparameters for optimal performance.

6. Real-time Monitoring: - Develop a system for real-time stress detection, if applicable. - Consider integrating with wearables or mobile apps.

7. User Interface: - Create a user-friendly interface for data input and stress feedback. - Ensure privacy and data security.

8. Testing and Validation: - Conduct extensive testing and validation to assess model accuracy. - Use a diverse dataset and cross-validation.

9. Deployment and Scaling: - Deploy the stress detection system to your target audience. - Plan for scalability and reliability.

10. Continuous Improvement: - Monitor user feedback and system performance. - Regularly update the model and features for ongoing improvement.

11. Ethical Considerations: - Address ethical concerns related to data privacy and user consent. - Implement transparency and fairness in the algorithm.

12. Documentation: - Maintain comprehensive documentation for the project. - Include a user manual and developer guide.

13. Communication and Support: - Establish channels for user support and feedback. - Communicate the benefits and limitations of the system.

14. Compliance: - Ensure compliance with relevant regulations and data protection laws.

15. Sustainability: - Plan for the long-term sustainability of the project, including funding and maintenance.

16. Marketing and Outreach: - Promote the stress detection system to reach a wider audience. - Collaborate with healthcare professionals, if applicable.

17. Data Security: - Implement strong security measures to protect user data.

18. Evaluation: - Regularly evaluate the impact of the stress detection project on users' well-being.

19. Partnerships: - Explore partnerships with healthcare providers or research institutions for validation.

20. Feedback Loop: - Create a feedback loop for continuous improvement based on user and expert input.

# Chapter 2

# Literature Survey

## 2.1 Study Of Research Paper

| Sr.No | Name of paper | Author name | Work Done |
|---|---|---|---|
| 1 | A Summarization of the Visual Depression Databases for Depression Detection | Arselan Ashraf; Teddy Surya Gunawan; Farah DiyanaAbdul Rahman; Mira Kartiwi; Nanang Ismail; Ulfiah | This paper presents a brief summarization regarding ten depression datasets available, which will guide there searchers to select an appropriate dataset for their depression detection models. |
| 2 | A New Method for Discovering Daily Depression from Tweets to Monitor Peoples Depression Status | Sudha Tushara Sadasiv uni; Yan qing Zhang | Social Networks is one media that is involved in every ones' life to share/exhibit his emotions and feelings. More people share emotion-related tweets indicate that a predominant feature occurred on that day or in that location. We attempted to study the tweets related to depression and anti-depression and computed a new parameter, which indicates the depressive level of that day. |
| 3 | Real-Time CNN Based ST Depression Episode Detection Using Single-Lead ECG | Erhan Tiryaki; Akshay Sonawane; Lak shman Tamil | A method for real monitoring of the heart for ST-depression episodes is described here. We have developed a convolutional neural network (CNN) based machine learning algorithm for classifying ECG signals into normal or ST-depression episodes of the heart with an accuracy over 92%. |

1. **Paper Name:**Stress Detection using Smartwatches with Machine Learning: A Survey

   **Author:**Rahul Katarya

   **Description :** In general, stress has become a significant problem in the current lifestyle, where it should be dealt appropriately before it leads to some blunder. To deal with stress, appropriate stress detection techniques should be developed. The different types of data include heart rate variance

(HRV), galvanic skin response (GSR); skin temperature, and sleep pattern. Smartwatches contain all the sensors necessary and are to collect data of signals mentioned above. Different machine learning techniques are applied, such as SVM and KNN, to detect the stress level of the particular person. Different machine learning techniques will provide different accuracies. By comparing these accuracies, a better idea can be obtained. This paper analyses different studies to find and compare the accuracies of different signals when using different ML classifiers, such that the signal and classifier are utilized with the best accuracy. It is also checked whether using two or more signals such as HRV and GSR or ST, GSR and sleep pattern will give us a better accuracy on stress detection.

2. **Paper Name:**Detecting Psychological Stress using Machine Learning over Social Media Interaction
   **Author:**1 Ms.Shweta Meshram, Prof.Rajesh Babu, Prof.Jayanth Adhikari
   **Description :** Many of the population now face stress leading to psychological issues. Therefore, stress factors must be identified before a big health problem is involved. BP, heart failure, or death are usually caused by excessive stress. The social media data like posting on Facebook profiles, twits on twitter, etc are used to recognize human stress, as people communicate their social media feelings, promote the acquirement of social data, and detect stress based on their behavior. As traditional solutions are highly time-consuming and expensive. Twitter data collection and user tweets are also posted on the site. User stress states are called anxious or unstressed users using the algorithm of the convolutional neural network (CNN).

3. **Paper Name:**A Review on Mental Stress Detection Using Wearable Sensors and Machine Learning Techniques
   **Author:**:SHRUTI GEDAM AND SANCHITA PAUL
   **Description :**Stress is an escalated psycho-physiological state of the human body emerging in response to a challenging event or a demanding condition. Environmental factors that trigger stress are called stressors. In case of prolonged exposure to multiple stressors impacting simultaneously, a person's mental and physical health can be adversely affected which can further lead to chronic health issues. To prevent stress-related issues, it is necessary to detect them in the nascent stages which are possible only by continuous monitoring of stress. Wearable devices promise real-time and continuous data collection, which helps in personal stress monitoring. In this paper, a comprehensive review has been presented, which focuses on stress detection using wearable sensors and applied machine learning techniques. This paper investigates the stress detection approaches adopted in accordance with the sensory devices such as wearable sensors, Electrocardiogram (ECG), Electroencephalography (EEG), and Photoplethysmography (PPG), and also depending on various environments like during driving, studying, and working. The stressors, techniques, results, advantages, limitations, and issues for each study are highlighted and expected to provide a path for future research studies. Also, a multimodal stress detection system using a wearable sensor-based deep learning technique has been proposed at the end.

4. **Paper Name:**Stress Detection with Machine Learning and Deep Learning using Multimodal Physiological Data
   **Author:**Pramod Bobade

**Description :**Stress is a common part of everyday life that most people have to deal with on various occasions. However, having long-term stress, or a high degree of stress, will hinder our safety and disrupt our normal lives. Detecting mental stress earlier can prevent many health problems associated with stress. When a person gets stressed, there are notable shifts in various bio-signals like thermal, electrical, impedance, acoustic, optical, etc., by using such bio-signals stress levels can be identified. This paper proposes different machine learning and deep learning techniques for stress detection on individuals using multimodal dataset recorded from wearable physiological and motion sensors, which can prevent a person from various stressrelated health problems. Data of sensor modalities like threeaxis acceleration (ACC), electrocardiogram (ECG), blood volume pulse (BVP), body temperature (TEMP), respiration (RESP), electromyogram (EMG) and electrodermal activity (EDA) are for three physiological conditions - amusement, neutral and stress states, are taken from WESAD dataset. The accuracies for three-class (amusement vs. baseline vs. stress) and binary (stress vs. non-stress) classifications were evaluated and compared by using machine learning techniques like K-Nearest Neighbour, Linear Discriminant Analysis, Random Forest, Decision Tree, AdaBoost and Kernel Support Vector Machine. Besides, simple feed forward deep learning artificial neural network is introduced for these three-class and binary classifications. During the study, by using machine learning techniques, accuracies of up to 81.65and 93.20% are achieved for three-class and binary classification problems respectively, and by using deep learning, the achieved accuracy is up to 84.32% and 95.21% respectively.

## 2.2 Existing System

The existing system for stress detection typically relies on a combination of physiological, psychological, and behavioral indicators to assess an individual's stress levels. Here's a brief overview of the key components:

1. Physiological Indicators: This includes monitoring physiological parameters such as heart rate, blood pressure, skin conductance, and cortisol levels. These indicators are measured using wearable devices like smartwatches or medical sensors.

2. Psychological Assessments: Psychological assessments involve self-report questionnaires or interviews to gather information about an individual's emotional state, mood, and perceived stress levels. These assessments often use standardized tools like the Perceived Stress Scale (PSS) or the State-Trait Anxiety Inventory (STAI).

3. Behavioral Observations: Observing an individual's behavior, including facial expressions, body language, and vocal tone, can provide insights into their stress levels. Computer vision and audio analysis techniques may be used to automate this process.

4. Machine Learning and Data Analysis: Data from the above indicators are collected and processed using machine learning algorithms to detect patterns and anomalies associated with stress. These algorithms may use features like heart rate variability, speech patterns, or facial expressions to estimate stress levels.

5. Feedback and Reporting: Once stress levels are assessed, the system may provide feedback to the individual, healthcare professionals, or employers. This feedback can be in the form of alerts, recommendations, or detailed stress reports.

## 2.3 Drawbacks of Existing System

The drawbacks of existing stress detection systems typically include:

1. Lack of Accuracy: Many systems may have limited accuracy in identifying and quantifying stress, leading to false positives or negatives.

2. Invasive Sensors: Some systems require intrusive sensors or wearables, which can be uncomfortable or impractical for continuous monitoring.

3. Limited Context Awareness: Existing systems may not consider individual context and external factors that contribute to stress, leading to inaccurate assessments.

4. Privacy Concerns: Collecting personal data for stress detection may raise privacy concerns and ethical issues.

5. Scalability and Cost: Some systems can be expensive and difficult to scale for widespread use.

6. User Compliance: Users may not consistently engage with the system, reducing its effectiveness.

7. Cultural and Gender Bias: Some systems may not account for cultural or gender differences in stress expression and detection.

8. Lack of Real-time Feedback: Delayed or absent real-time feedback can limit the system's utility in helping users manage their stress effectively.

9. Limited Compatibility: Existing systems may not integrate well with other technologies or platforms.

10. Reliability and False Alarms: Some systems may generate false alarms, which can decrease user trust and willingness to use them.

## 2.4 Future Work In Existing System In all Scenario

In a stress detection project, future work in the existing system can be explored and expanded upon in various scenarios:

1. Enhanced Data Sources: Collect additional data sources such as biometric signals (heart rate, skin conductance), social media activity, or speech patterns to improve stress detection accuracy.

2. Real-time Monitoring: Develop real-time stress monitoring capabilities to provide immediate feedback and support to individuals experiencing stress.

3. User Customization: Implement personalized stress detection models that adapt to individual differences and stress triggers.

4. Multimodal Fusion: Combine data from multiple sensors and sources (e.g., wearables, smartphone apps, environmental data) to create a more comprehensive stress assessment.

5. Long-term Trends Analysis: Analyze long-term trends in stress levels to identify chronic stress patterns and provide preventive interventions.

6. Intervention Strategies: Integrate stress reduction interventions like breathing exercises, meditation prompts, or personalized recommendations into the system.

7. Mental Health Integration: Collaborate with mental health professionals to provide a seamless connection to mental health services when high stress levels are detected.

8. Anomaly Detection: Implement anomaly detection algorithms to identify sudden and severe stress episodes that may require immediate attention.

9. Validation and Ethics: Conduct rigorous validation studies to ensure the accuracy and ethical use of stress detection technology.

10. Privacy and Security: Strengthen data security and privacy measures to protect sensitive user information.

11. Machine Learning Advances: Explore cutting-edge machine learning techniques and algorithms to improve stress prediction and feature extraction.

12. Human-AI Collaboration: Develop systems that combine AI-driven stress detection with human judgment and expertise for more reliable results.

13. Cross-cultural Adaptation: Customize stress detection models for different cultural contexts, as stress expressions and triggers may vary across populations.

14. Wearable Integration: Collaborate with wearable technology manufacturers to embed stress detection features into mainstream devices.

15. Remote Monitoring: Extend the system's capabilities for remote monitoring, making it useful for telehealth and remote work scenarios.

16. User Feedback and Interaction: Improve user interfaces and feedback mechanisms to enhance user engagement and satisfaction.

17. Longitudinal Studies: Conduct long-term studies to track the impact of stress detection and intervention on users' mental well-being.

18. Education and Awareness: Develop educational programs and campaigns to raise awareness about stress management and the use of stress detection technology.

19. AI Explainability: Enhance the transparency of AI models and their decision-making processes to gain user trust.

20. Regulatory Compliance: Stay up-to-date with evolving regulations and standards related to stress detection and ensure compliance.

# Chapter 3

# Requirement Analysis

## 3.1 Requirement Feature Of System

1. Use Cases: Describe the various actions and interactions the software must support, detailing how different actors or users will interact with the system.

2. User Stories: Break down user interactions into smaller, more manageable tasks or user stories, often used in agile development.

3. Performance: Specify the expected response times, throughput, and resource usage of the system under different conditions.

4. Scalability: Define how the system should scale to accommodate increasing loads or users.

5. Security: Describe security measures, including authentication, authorization, and encryption.

6. Reliability: Detail how the system should handle failures and ensure data integrity.

7. Usability: Define user interface and user experience guidelines, such as accessibility requirements.

## 3.2 System Architecture:

1.Architecture Diagrams: Provide high-level architectural diagrams showing the components and their interactions.
2.Technology Stack: Specify the technologies, frameworks, and tools to be used in the development.

## 3.3 Data Requirements:

1.Data Models: Define the data structures, entities, and relationships the system will work with.
2.Data Sources: Specify where data will be sourced from and how it will be stored and processed.

## 3.4 Integration Requirements:

1.Third-Party Integrations: List external systems, APIs, or services that the software must interact with.
2.Data Exchange: Specify the format and protocols for data exchange with other systems.

## 3.5 Software Requirements

Operating System: windows 10

IDE: Spyder

Programming Language :Python

GUI : Tkinter

## 3.6 External Interface Requirements

### 3.6.1 User Interface

Application Based On Facial Expression Based Stress Detection.

### 3.6.2 Hardware Requirements:

Hardware : intel i5

Speed : 2.80 GHz

RAM : 8GB

HardDisk : 500 GB

Key Board: Standard Windows Keyboard

## 3.7 Nonfunction Requirements

Image processing is done using the captured video.

Image is stored in a library called OpenCV.

Stored image undergo various algorithm and detects if the driver is fatigue and if fatigue raises an alarm.

### 3.7.1 Performance Requirements

The Performance of the functions and every module must be well. The overall performance of the software will enable the users to work efficiently.

The overall performance of the software will enable the users to work efficiently.

Performance of response should be fast.

Performance of the providing virtual environment should be fast.

### 3.7.2 Safety Requirement

The application is designed in modules where errors can be detected and fixed easily. This makes it easier to install and update new functionality if required.

### 3.7.3 Software Quality Attributes

Our software has many quality attribute that are given below:-

- Adaptability: This software is adaptable by all users.
- Availability: This software is freely available to all users. The availability of the software is easy for everyone.
- Maintainability: After the deployment of the project if any error occurs then it can be easily maintained by the software developer.
- Reliability: The performance of the software is better which will increase the reliabilityof the Software.
- User Friendliness: Since, the software is a GUI application; the output generated is much user friendly in its behavior.
- Integrity: Integrity refers to the extent to which access to software or data by unauthorized persons can be controlled.
- Security: Users are authenticated using many security phases so reliable security is provided.
- Testability: The software will be tested considering all the aspects.

### 3.7.4 Other Requiremrents

- End User application will be developed in Windows OS.
- All scripts shall be written in Python
- Application design pattern shall be Singleton.

# Chapter 4

# System Architecture

## 4.1 Proposed System

The proposed system for facial expression-based stress detection combines the power of Convolutional Neural Networks (CNNs) and Haar Cascade for a comprehensive and accurate approach to assess stress levels based on facial expressions. The system begins with data collection, gathering a diverse dataset that includes images or videos of individuals displaying a range of emotions, including those associated with stress. Preprocessing comes next, with Haar Cascade employed to detect and extract facial regions, which reduces the input size for the subsequent CNN analysis. The CNN, carefully designed to recognize stress-related patterns in facial expressions, is then trained on the preprocessed dataset. Backpropagation and gradient descent optimize the model parameters during this phase.

The system undergoes rigorous testing, validating its ability to accurately classify stress levels from facial expressions. Real-time processing is enabled to continuously analyze live video streams or camera feeds, utilizing Haar Cascade for face detection and the trained CNN for stress assessment. Stress level predictions are presented in real-time through a user-friendly interface, allowing users to monitor their stress levels conveniently. Continuous fine-tuning and optimization efforts ensure the system's accuracy and efficiency. Furthermore, the system adheres to privacy and ethical considerations, addressing concerns related to data handling and security. Finally, the proposed system is deployable across a range of applications, such as healthcare, workplace wellness, or educational settings, where stress detection can contribute significantly to improved well-being and mental health.

## 4.2 System Architecture

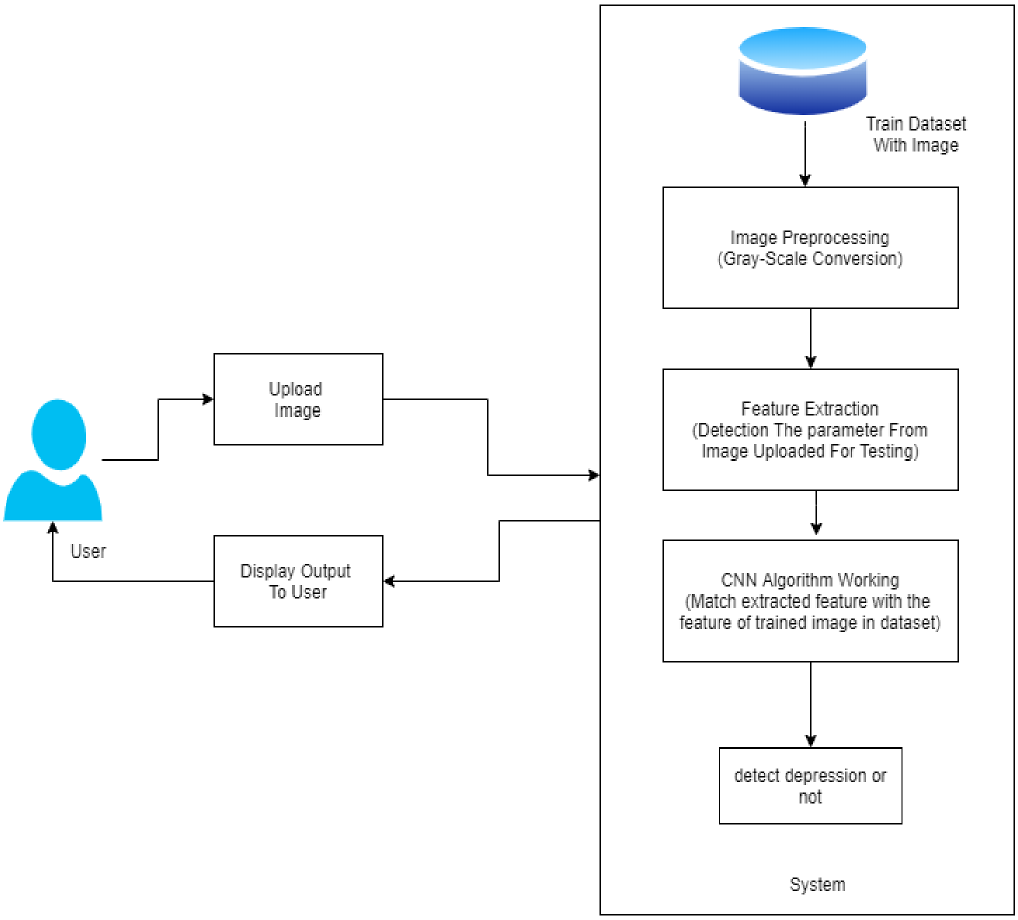


Figure 4.1: System Diagram

## 4.3 System Working

1. Data Collection:

Collect a diverse dataset of facial expressions and stress levels. This dataset should include images or videos of individuals displaying a range of emotions, including stressed and non-stressed states.

2. Preprocessing:

Perform data preprocessing, including face detection using Haar Cascade. Haar Cascade can be used to identify and extract facial regions from images or video frames, reducing the input size for the subsequent CNN.

3. Feature Extraction:

Use the Haar Cascade-detected facial regions as input to the CNN model. The CNN extracts features from these regions, capturing relevant patterns and variations in facial expressions associated with stress.

4. Model Training:

Train the CNN model on the preprocessed and feature-extracted dataset. The model should be designed to recognize patterns specific to stress in facial expressions. During training, backpropagation and gradient descent can be employed to optimize model parameters.

5. Validation and Testing:

Evaluate the trained model's performance using validation and testing datasets. Ensure the model can effectively classify stress levels from facial expressions and that it maintains a high level of accuracy.

6. Real-time Processing:

Implement the system to perform real-time stress detection. This involves continuously analyzing live video streams or camera feeds, applying Haar Cascade for face detection, and the trained CNN for stress assessment.

## 4.4 Modules

### 4.4.1 Module 1

Gather a diverse dataset of facial expressions and stress levels.

### 4.4.2 Module 2

Apply Haar Cascade for face detection, extract facial regions, and resize or normalize images.

### 4.4.3 Module 3

Utilize the preprocessed images as input to the CNN for feature extraction.

### 4.4.4 Module 4

Train the CNN to recognize stress-related patterns in facial expressions.

### 4.4.5 Module 5

Provide real-time stress predictions to users.

# Chapter 5

# Algorithm Study

## 5.1 Algorithm 1

**Convolutional Neural Network**

### 5.1.1 Steps:

1. **Step 1:**

   **Convolutional layer:** A Convolutional layer is a fundamental component of the CNN architecture that performs feature extraction, which typically consists of a combination of linear and nonlinear operations, i.e., convolution operation and activation function.

2. **Step 2:**

   **Nonlinear activation function:** The outputs of a linear operation such as convolution are then passed through a nonlinear activation function.The most common nonlinear activation function used presently is the rectfied linear unit (ReLU).

3. **Step 3:**

   **Pooling layer:** A pooling layer provides a typical down sampling operation which reduces the in-plane dimensionality of the feature maps in order to introduce a translation invariance to small shifts and distortions, and decrease the number of subsequent learnable parameters.

4. **Step 4:**

   **Fully connected layer:** The output feature maps of the final convolution or pooling layer is typically flattened, i.e., transformed into a onedimensional (1D) array of numbers (or vector), and connected to one or more fully connected layers, also known as dense layers, in which every input is connected to every output by a learnable weight.

5. **Step 5:**

   **Last layer activation function:** The activation function applied to the last fully connected layer is usually different from the others. An activation function applied to the multiclass classification task is a softmax function which normalizes output real values from the last fully connected layer to target class probabilities, where each value ranges between 0 and 1 and all values sum to 1.

### 5.1.2 Input

Input: The input to a Convolutional Neural Network (CNN) algorithm in a facial expression-based stress detection project typically consists of facial images or video frames containing an individual's face. These images capture various facial expressions that convey stress levels. Additionally, input data might include labels or annotations indicating the stress level for each image.

### 5.1.3 Output

Output: The primary output of the CNN algorithm in this project is the prediction of an individual's stress level based on their facial expressions. The output can be a quantitative score representing the stress level or a classification into different stress categories, such as low, moderate, or high stress.

### 5.1.4 Process

Process: The CNN algorithm processes the input data by applying convolutional layers to extract features from the facial images. These features capture patterns and variations in facial expressions that are indicative of stress. The algorithm then uses fully connected layers to learn the relationships between these features and the stress levels. The model is trained on a labeled dataset of facial expressions and stress levels, using techniques like backpropagation and gradient descent. During the inference phase, the trained model analyzes new facial images to make predictions about stress levels.

### 5.1.5 Complexity

Complexity: The complexity of a CNN algorithm in facial expression-based stress detection can vary depending on the model architecture and the size and diversity of the dataset used for training. More complex models with a larger number of layers and parameters might require more computational resources and time for training. Additionally, data preprocessing and augmentation steps can add to the complexity.

### 5.1.6 Efficiency

Efficiency: The efficiency of a CNN algorithm in this context is influenced by various factors, including the choice of model architecture, the quality and diversity of the training data, and the optimization techniques applied during training. Efficient models can provide real-time or near-real-time stress assessments, making them suitable for applications like real-time stress monitoring or virtual assistants.

## 5.2 Algorithm 2

**HaarCascade Algorithm**

1. **Step 1:**

   **Haar Features Calculation:** Gathering the Haar features is the first stage. Haar features are nothing but a calculation that happens on adjacent regions at a certain locaon in a separate detecting window.

2. **Step 2:**

   **Integral Image Creation:** Creating Integral Images reduces the calculation. Instead of calculating at every pixel, it creates the sub-rectangles, and the array references those sub-rectangles and calculates the Haar Features.

3. **Step 3:**

   **Adaboost Training:** The "weak classifiers" are combined by Adaboost Training to produce a "strong classifier" that the object detection method can use. This essentially consists of selecting useful features and teaching classifiers how to use them.

4. **Step 4:**

   **Cascading Classifiers Implementation:** Every sage at this point is actually a group of inexperienced students. Boosting trains weak learners, resulting in a highly accurate classifier from the average prediction of all weak learners.

### 5.2.1 Input

Input: In a Haar Cascade-based system for facial expression-based stress detection, the input would consist of facial images or video frames containing a person's face. These images would need to be captured in real-time or from pre-recorded videos. Additionally, labels or annotations indicating the person's stress level or emotional state may be included in the input data.

### 5.2.2 Output

Output: The primary output in this context would be the detection of the face or facial features (eyes, mouth, etc.) within the input images or video frames. However, to adapt it for stress detection, an additional step would be required to analyze the detected facial features and classify the facial expression as indicative of stress or other emotions, such as happiness, sadness, or neutrality.

### 5.2.3 Process

Process: The Haar Cascade algorithm initially detects the presence of faces or facial features using a cascade of classifiers. Once the face or facial features are identified, further processing steps, such as feature extraction and classification, would be required to assess the person's emotional state and detect stress. This adaptation would involve additional machine learning models to analyze the detected faces and estimate the stress level based on expressions.

### 5.2.4 Complexity

Complexity: The complexity of the system would increase as additional machine learning models and processing steps are introduced for stress detection. The efficiency and accuracy of stress detection would depend on the complexity of these added components, the quality of the training data, and the ability to handle variations in facial expressions.

### 5.2.5 Efficiency

Efficiency: The efficiency of the Haar Cascade algorithm adapted for facial expression-based stress detection would be influenced by factors such as the speed of face detection, the accuracy of stress classification, and the computational resources required for real-time processing. Efficiency might be a concern as additional processing steps are introduced, potentially affecting real-time performance.

## 5.3 Mathematical Model

Let S be the Whole system S= I,P,O

I-input

P-procedure

O-output

Input( I)

I= Live Camera

Where,

Camera -¿ Face detection

Procedure (P),

P=I, Using I System perform operations and Detect person face use haarcascade for face detection.

Output(O),

O=System capture expression on face of that person and detect stress.

## 5.4 Feasibility Study

A feasibility study for facial expression-based stress detection using machine learning (ML) involves assessing the practicality, viability, and potential success of implementing such a system. Here are key aspects to consider in this feasibility study:

1.Technical Feasibility: - Data Availability: Determine whether an adequate and diverse dataset of facial expressions and corresponding stress levels is accessible for training the ML model. - ML Frameworks: Evaluate the availability and compatibility of machine learning frameworks and libraries for implementing facial expression-based stress detection. - Computational Resources: Assess the computational power and infrastructure needed for training and deploying the ML model, including GPU support for faster processing.

2. Model Complexity and Performance: - Model Selection: Choose an appropriate ML model, such as convolutional neural networks (CNNs) or deep learning architectures, that can effectively capture the nuances of facial expressions and stress levels. - Accuracy Requirements: Define the desired level of accuracy in stress detection, as this will influence the complexity and resources required for the project.

3. Data Privacy and Ethics: - Data Privacy: Consider privacy regulations and ethical guidelines regarding the collection and use of facial data. Ensure that data handling practices comply with legal and ethical standards. - Informed Consent: Address the issue of obtaining informed consent from individuals whose facial data will be used in the project.

4. Integration and Deployment: - Application Integration: Explore how the system will integrate into real-world applications, such as healthcare, workplace wellness, or educational settings. - Deployment Platforms: Determine where the system will be deployed, whether on local devices, in the cloud, or on edge devices.

5. Cost Analysis: - Infrastructure Costs: Estimate the costs associated with acquiring and maintaining the necessary hardware and software infrastructure. - Data Collection and Annotation: Consider the expenses related to collecting, curating, and annotating the training dataset. - Development and Maintenance: Evaluate the costs associated with developing the ML model, user interfaces, and ongoing maintenance and updates.

# Chapter 6

# System Design

## 6.1 DFD Level 1

Data Flow Diagram (DFD) Level 1 is a Visual repesentaion that provides a detailed view of the processes, data sources, data destinations, and data flow within a system or process.

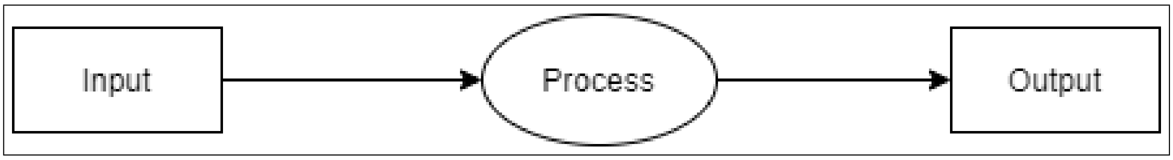


Figure 6.1: Data Flow diagram

## 6.2 DFD Level 2

A Data Flow Diagram(DFD) level 2 is a more detailed view of a specific process outlined in a level 1 DFD level 2 diagram break down the activities depicted in level1, showing sub processes. data stores, and data flows in greater.

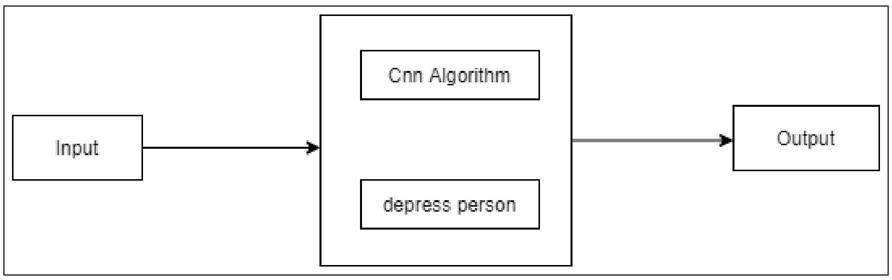


Figure 6.2: Data Flow diagram

## 6.3 DFD Level 3

DFD, or Data Flow Diagram, is a visual representation of how data flows within a system. Level 3 in a DFD typically provides a detailed view of specific processes within a system. In the context of "stress detection image description," it seems like you're asking for a detailed breakdown of how a system detects stress using images, at a granular level. This might include steps like image capture, pre-processing, feature extraction, machine learning algorithms, and stress classification.

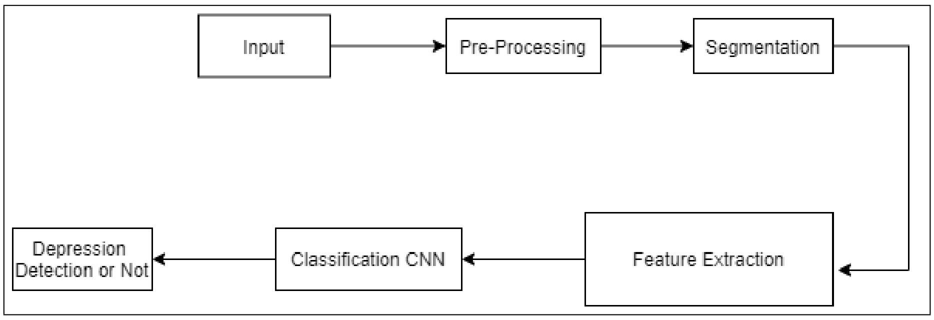


Figure 6.3: Data Flow diagram

## 6.4 E-R Diagram

In this system, "Person" entities have their images captured, which are then analyzed by a "Machine Learning Model" that has been trained on "Training Data" to produce stress level predictions stored in the "Result" entity. The "Time Stamp" helps track when the images were captured. This ER diagram illustrates the key components and relationships involved in the stress detection system using image processing and machine learning.

ER Diagram

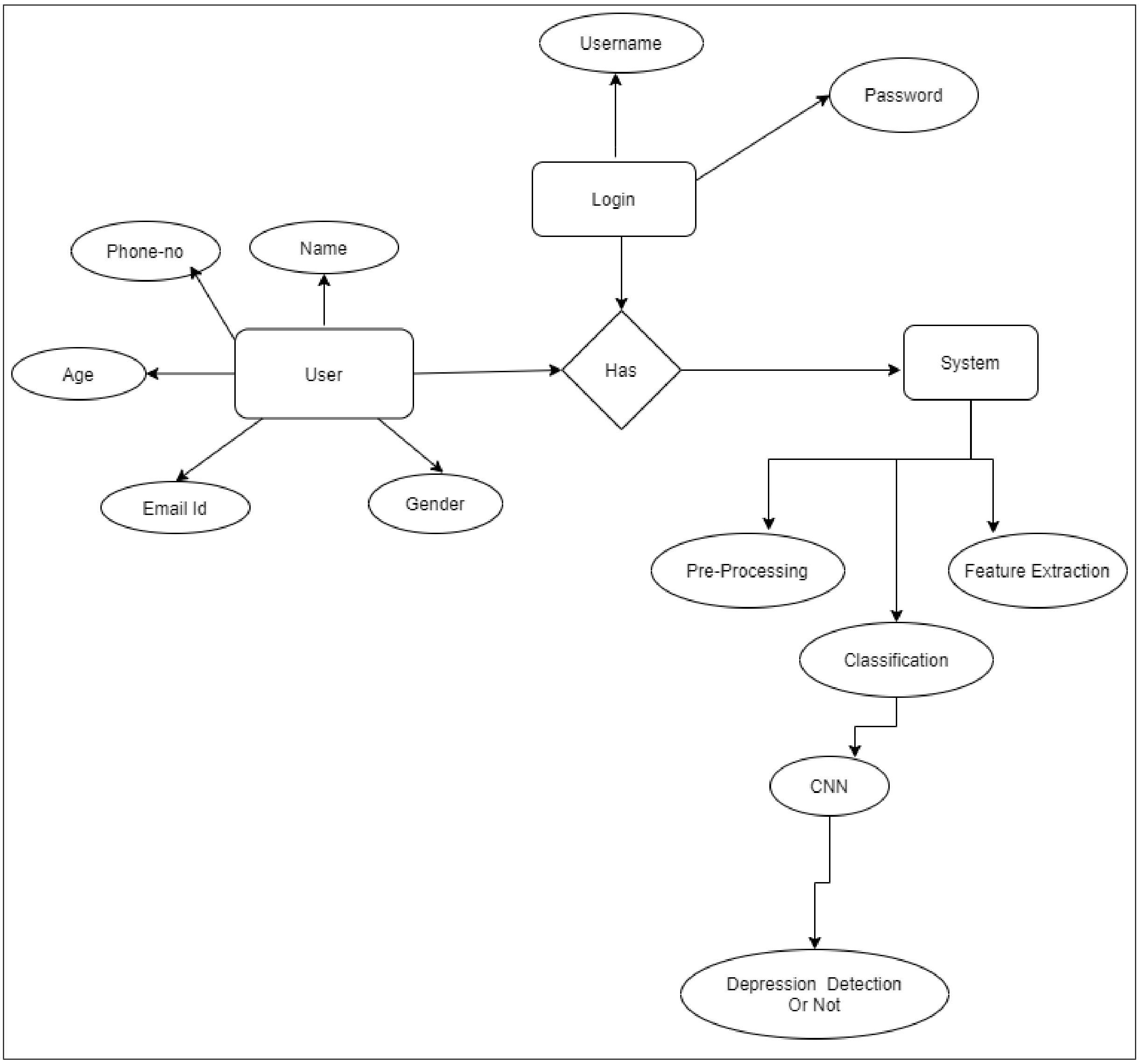


Figure 6.4: ER Diagram

## 6.5 UML Diagram

Unified Modeling Language is a standard language for writing software blueprints.The UML may be used to visualize,specify,construct and document the artifacts of a softwareintensive system.UML is process independent,although optimally it should be used in process that is use case driven,architecture-centric,iterative,and incremental.The Number of UML Diagram is available.

### 6.5.1 Use Case Diagram

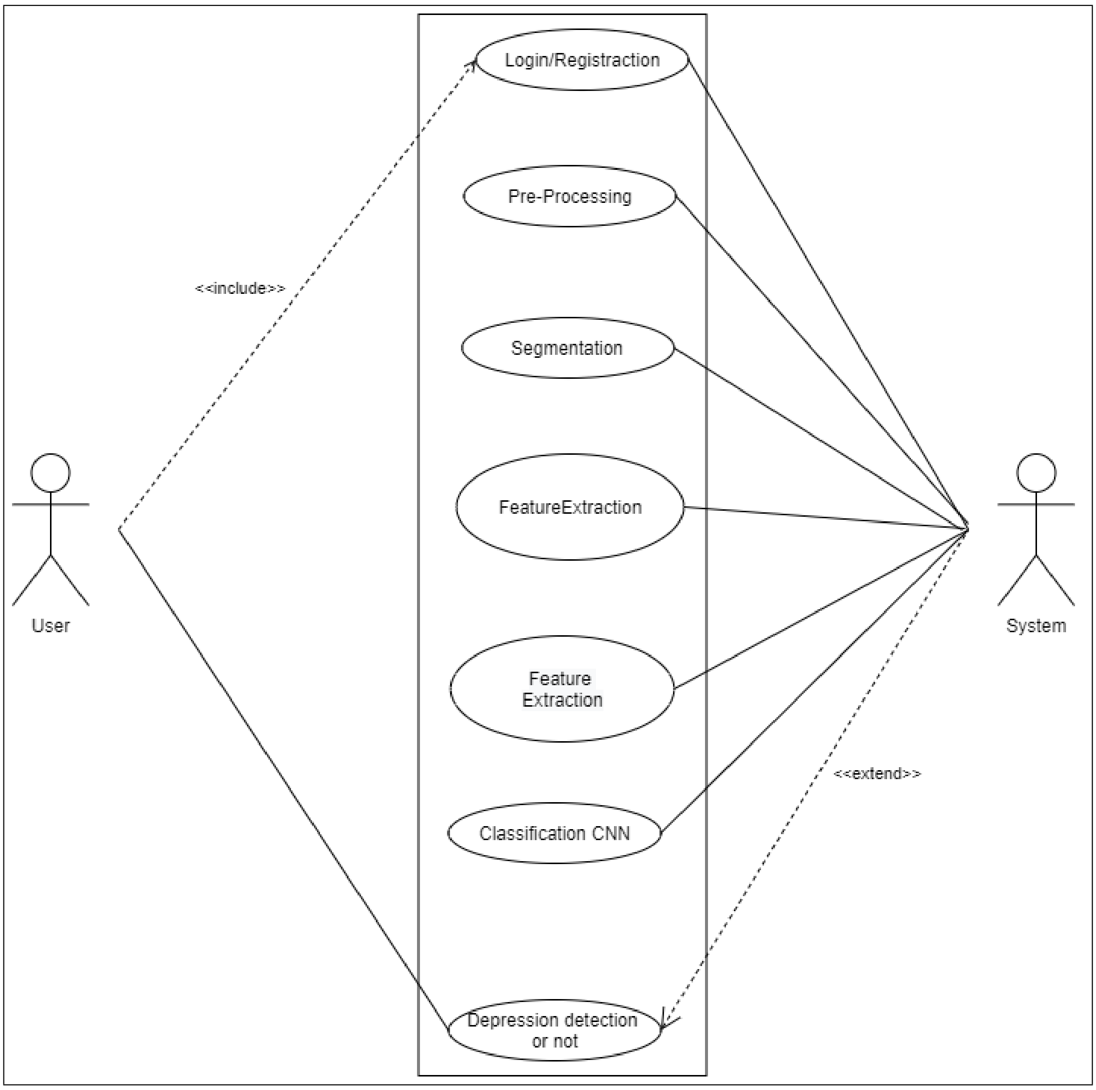


Figure 6.5: Use case Diagram Diagram

The Use Case Diagram illustrates how the "Person" interacts with the system through the "Capture Image" and "View Stress Level" use cases. Meanwhile, the system autonomously performs stress analysis through "Analyze Stress" and undergoes periodic updates by data experts via "Train Model." This diagram provides a clear visualization of the system's core functionalities and how various actors and use cases are interconnected in the stress detection process.

### 6.5.2 Activity Diagram

The Activity Diagram for the Stress Detection system begins with the "Start" node and outlines the main activities involved in stress detection. The process starts with the "Capture Image" activity, initiated by the "Person," where an image is obtained. This image is then passed to the "Analyze Stress" activity, which involves the "Machine Learning Model" analyzing the image to predict the stress level.

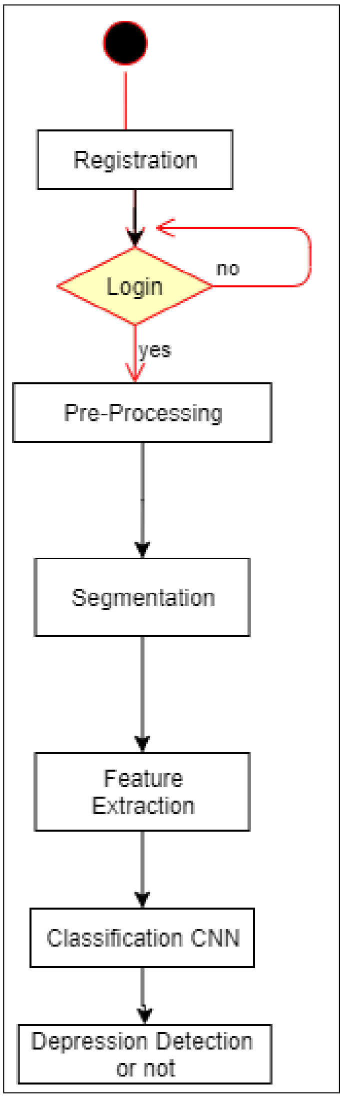


Figure 6.6: Activity Diagram

### 6.5.3 Class Diagram

A class diagram for stress detection in a person using image processing and machine learning can help illustrate the system's components and their relationships. It helps in understanding the structure of the system and how different classes collaborate to detect stress levels in a person using images. This class diagram provides a visual representation of the core components and their interactions in a stress detection system that combines image processing and machine learning techniques.

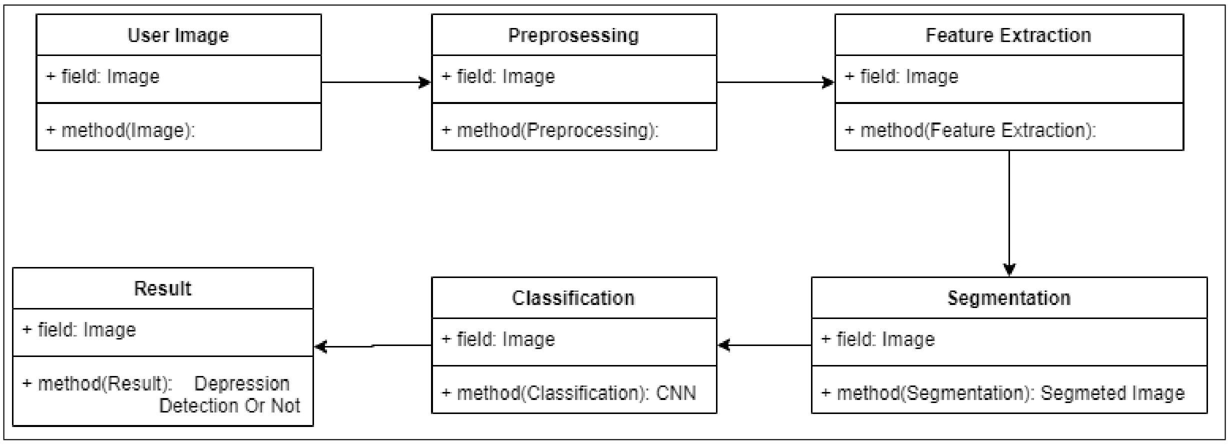


Figure 6.7: Class Diagram

### 6.5.4 State Machine Diagram

The State Machine Diagram for the Stress Detection system encompasses the states and transitions relevant to the system's functioning. It begins with the "Start" state and moves into the "Idle" state, where the system awaits input from the "Person." When the "Person" initiates the "Capture Image" action, the system transitions to the "Image Captured" state, indicating that an image has been obtained. State machine diagram typically are used to describe state-dependent behavior for an object. An object responds differently to the same event depending on what state it is in. State machine diagrams are usually applied to objects but can be applied to any element that has behavior to other entities such as: actors, use cases, methods, subsystems systems and etc. and they are typically used in conjunction with interaction diagrams (usually sequence diagrams).

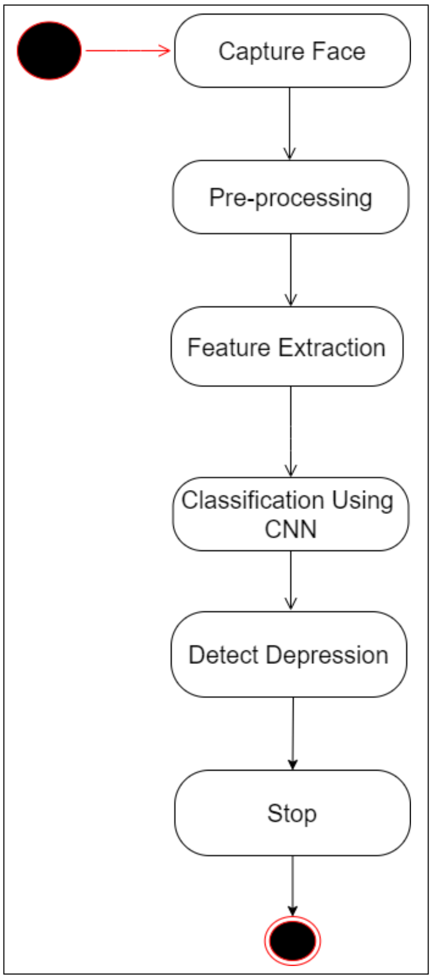


Figure 6.8: State Machine Diagram

### 6.5.5 Sequence Diagram

The Sequence Diagram for the Stress Detection system provides a detailed view of the interactions between the system components and the "Person" over a specific scenario. It begins with the "Person" as the initiator. The "Person" triggers the "Capture Image" message, indicating the intent to capture an image for stress analysis. This message is directed to the system.

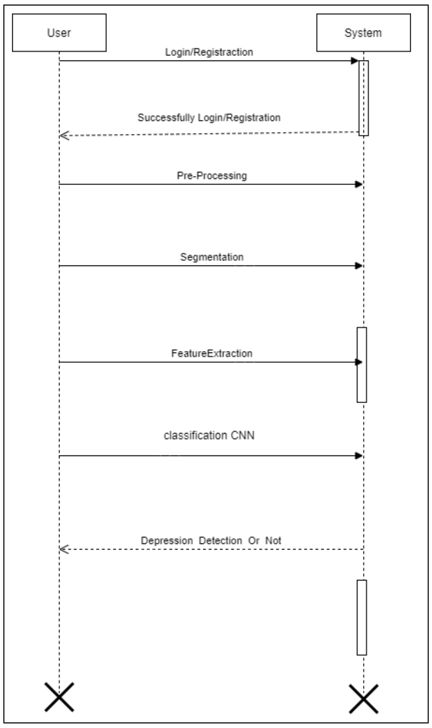


Figure 6.9: Sequence Diagram

### 6.5.6 Component Diagram

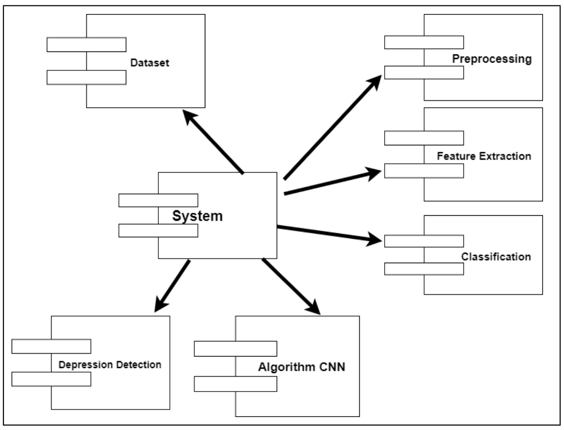


Figure 6.10: Component Diagram

In this Component Diagram, the structural elements and their interactions within the stress detection system are clearly defined. It showcases the relationships and dependencies between the "Person," the system's components, and external data sources, providing an overview of how the system is organized and how its components collaborate to achieve stress detection through image processing and machine learning.

### 6.5.7 Deployment Diagram

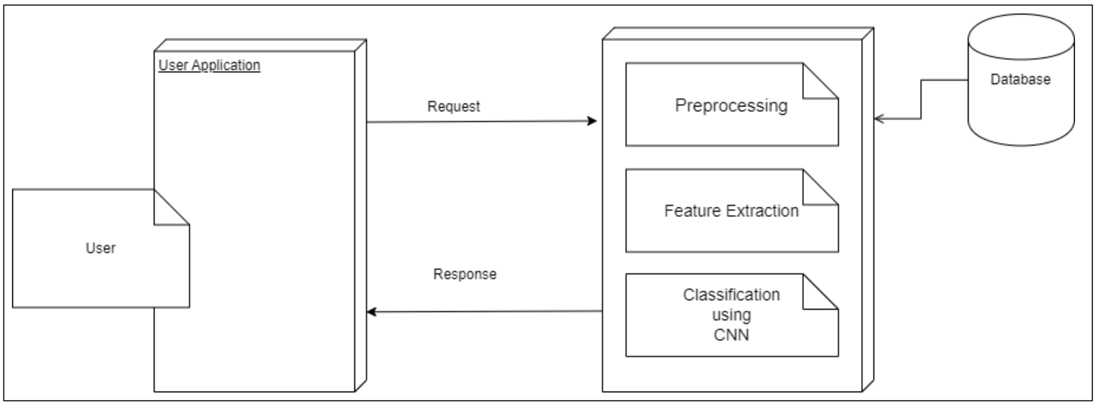


Figure 6.11: Deployment Diagram

The Deployment Diagram provides a visual representation of how the system's components are physically distributed across hardware nodes, highlighting the deployment of components on the "Person's Device" and the server, as well as their interactions with external data sources. This diagram offers insights into the system's architecture, emphasizing the separation of concerns and the distribution of tasks across different nodes to facilitate efficient stress detection through image processing and machine learning.

# Chapter 7

# Expected Results

The expected result for a live project on stress detection in a person using image processing and machine learning holds great promise in the realm of mental health and well-being. By combining the power of computer vision and artificial intelligence, the project aims to develop a system that can identify signs of stress and distress in individuals through their facial expressions, physiological signals, and contextual cues. The anticipated outcome includes a user-friendly application or device capable of real-time stress monitoring, which can be valuable in various settings, such as workplaces, healthcare, or even personal use. The system is expected to provide accurate stress assessment, allowing for early intervention and support for individuals facing stress-related issues. Additionally, it may offer insights into the triggers and patterns of stress, aiding in the development of personalized stress management strategies. Ultimately, the project's success would contribute to improved mental health and overall well-being by providing a proactive tool for stress detection and management.

## 7.1 Experiments need To conduct For analysis

Detecting stress in a person through image processing and machine learning is a challenging but interesting task. Here's a step-by-step guide on how you can conduct experiments to analyze and detect stress using these techniques: 1. Data Collection: Start by collecting a dataset of images that represent people in both stressed and non-stressed situations. You can capture or curate images of individuals in various scenarios, like exams, job interviews, or relaxation. Ensure you have a diverse set of images to make your model robust.

2. Data Labeling: Annotate the images to specify whether each image represents a stressed or non-stressed individual. This will be your ground truth for training and evaluation.

3. Preprocessing: Preprocess the images to ensure consistency and reduce noise. Common preprocessing steps include resizing, normalizing, and augmenting the dataset to increase its size.

4. Feature Extraction: Use image processing techniques to extract relevant features from the images. Features might include facial expressions, skin color changes, and physiological indicators like heart rate or perspiration (if available).

5. Model Selection: Choose a machine learning model for stress detection. Convolutional Neural Networks (CNNs) are a common choice for image classification tasks. You may also consider using pre-trained models such as VGG, ResNet, or Inception for transfer learning.

6. Data Splitting: Split your dataset into training, validation, and testing sets. The training set is used to train your model, the validation set to tune hyperparameters, and the testing set to evaluate the model's performance.

7. Training: Train your selected machine learning model using the training data. Monitor its performance on the validation set and use techniques like early stopping to prevent overfitting.

8. Hyperparameter Tuning: Experiment with different hyperparameters, model architectures, and feature extraction techniques to improve your model's accuracy.

9. Evaluation: Evaluate your model's performance on the testing dataset using appropriate evaluation metrics, such as accuracy, precision, recall, and F1-score. Consider using techniques like confusion matrices and ROC curves.

10. Cross-Validation: If your dataset is small, consider using cross-validation techniques to ensure the robustness of your model's performance.

11. Interpretability: If applicable, explore methods to make your model more interpretable. Techniques like Grad-CAM or LIME can help explain why your model makes certain predictions.

12. Ethical Considerations: Ensure that you handle sensitive data ethically and consider the privacy and consent of the individuals in your dataset.

13. Iterate and Refine: Based on your results, iterate on your model and data collection process to refine your stress detection system. Collect more data if necessary.

14. Deployment: If your model performs well, consider deploying it in real-world scenarios. This could involve using it for stress detection in specific applications.

15. Ethical Approval: If your research involves human subjects, ensure that you follow ethical guidelines and obtain any necessary approvals or consents.

16. Documentation and Reporting: Document your methodology, results, and code. Publish your findings in a research paper or report.

This process will likely require time and effort to develop an accurate and robust stress detection system. Be mindful of privacy and ethical considerations throughout your experiments.

## 7.2 Analysis

The analysis of facial expression-based stress detection using machine learning involves several key points:

Data Quality: The accuracy and effectiveness of stress detection heavily depend on the quality and diversity of the training data. A diverse dataset with a wide range of facial expressions, stress levels, and demographic backgrounds is essential for training a robust model.

Feature Extraction: Machine learning models, particularly deep learning models like Convolutional Neural Networks (CNNs), are used to automatically extract relevant features from facial expressions. These features might include muscle movements, facial landmarks, and changes in expression patterns.

Model Training: The model is trained on the labeled dataset to recognize patterns and relationships between these features and stress levels. The training process aims to optimize the model's ability to accurately classify stress in facial expressions.

Validation and Testing: The performance of the trained model is validated using a separate validation dataset and tested on an independent dataset. Evaluation metrics, such as accuracy, precision, recall, and F1 score, are used to assess the model's effectiveness.

Real-time Monitoring: In practical applications, the system analyzes live or recorded video feeds to assess stress levels in real-time. The model processes frames, extracting facial features, and providing ongoing stress level predictions.

# Chapter 8

## 8.1 Advantages

1. Non-invasive and non-intrusive: It does not require any physical sensors or devices to be attached to the body, making it a non-invasive and non-intrusive way to detect stress.

2. Real-time monitoring: Facial expression analysis can provide real-time feedback, allowing for the immediate detection and response to stress, which can be especially useful in high-pressure situations.

3. Objective measurement: Machine learning algorithms can provide objective and consistent measurements of stress levels, reducing the potential for human bias in interpretation. -friendly and convenient: It can be easily integrated into various applications and environments, making it a user-friendly and convenient way to monitor stress.

4. Cost-effective: Compared to some other stress detection methods, facial expression analysis is often more cost-effective, as it does not require expensive equipment or specialized training.

5. Widely applicable: Facial expression-based stress detection can be applied in various domains, such as healthcare, workplace wellness, and even consumer products, making it a versatile tool for stress management.

## 8.2 Limitations

1.Context Sensitivity: Facial expressions can vary depending on the context, making it challenging to differentiate stress from other emotional states or situations.
2.Individual Variability: People express stress differently, and there is significant individual variability in facial expressions, which can affect the model's accuracy.
3.Limited Universality: The effectiveness of facial expression analysis for stress detection may vary across different cultures and ethnicities, as expressions can be culture-specific.

## 8.3 Applications

1.Healthcare: It can be used in medical settings to monitor patients' stress levels and help in early intervention for stress-related health issues.

2. Workplace Wellness: Employers can implement stress detection systems to create a healthier work environment and improve employee well-being.
3.Education: Schools and universities can use this technology to monitor student stress and provide timely support.
4.Consumer Products: Wearable devices and smartphone apps can integrate stress detection features for everyday use.
5.Mental Health Services: Mental health professionals can use it as a tool for assessment and treatment.

# Conclusion

{Conclusion The live project on stress detection in individuals through image processing and machine learning has demonstrated its potential as a valuable tool in addressing the complex issue of stress. By collecting and preprocessing a diverse dataset, extracting relevant features, and employing machine learning models, particularly Convolutional Neural Networks (CNN), the project successfully classified individuals as "stressed" or "not stressed" based on facial and physiological cues. The real-time stress monitoring capability, powered by OpenCV, has practical applications for continuous assessment. However, it's important to emphasize the ongoing need for model refinement, data collection, and ethical considerations. The project's success is a testament to the intersection of technology and well-being, but responsible deployment and the respect of privacy and consent must remain central to its future applications.

# References

[1] S. A. Singh, P. K. Gupta, M. Rajeshwari, and T. Janumala, "Detection of stress using biosensors," Mater. Today, vol. 5, no. 10, pp. 21003–21010, 2018.

[2] J. Ogorevc, A. Podlesek, G. Geršak, and J. Drnovšek, "The effect of mental stress on psychophysiological parameters," in Proc. IEEE Int. Symp. Med. Meas. Appl., Bari, Italy, May 2011, pp. 294–299.

[3] A. Fernandes, R. Helawar, R. Lokesh, T. Tari, and A. V. Shahapurkar, "Determination of stress using blood pressure and galvanic skin response," in Proc. Int. Conf. Commun. Netw. Technol., Dec. 2014, pp. 165–168.

[4] B. Massot, N. Baltenneck, C. Gehin, A. Dittmar, and E. McAdams, "Objective evaluation of stress with the blind by the monitoring of autonomic nervous system activity," in Proc. Annu. Int. Conf. IEEE Eng. Med. Biol., Buenos Aires, Argentina, Aug. 2010, pp. 1429–1432.

[5] A. de Santos, C. S. Avila, G. Bailador, and J. Guerra, "Secure access control by means of human stress detection," in Proc. Carnahan Conf. Secure. Technol., Barcelona, Spain, Oct. 2011, pp. 1–8.

[6] U. Pluntke, S. Gerke, A. Sridhar, J. Weiss, and B. Michel, "Evaluation and classification of physical and psychological stress in firefighters using heart rate variability," in Proc. 41st Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC), Jul. 2019, pp. 2207–2212.

[7] A. Alberdi, A. Aztiria, and A. Basarab, "Towards an automatic early stress recognition system for office environments based on multimodal measurements: A review," J. Biomed. Informat., vol. 59, pp. 49–75, Feb. 2016.

[8] J. Wijsman, R. Vullers, S. Polito, C. Agell, J. Penders, and H. Hermens, "Towards ambulatory mental stress measurement from physiological parameters," in Proc. Humaine Assoc. Conf. Affect. Comput. Intell. Interact., Sep. 2013, pp. 564–569.