**ACKNOWLEDGEMENT**

First and for most, I convey my reverential salutation to the almighty for keeping us fit

for the successful completion of this project.

I am extremely thankful and profound my deepest gratitude to my respected principal

Dr. Beshiba Wilson Ma’am for her guidance, motivation, encouragement and given throughout

the course.

I express my deep sense of gratitude to Miss. Bismi k Charleys, Head of the MCA

Department, Lourdes Matha College of Science and Technology and my guide for her keen

interest, kind advice, valuable guidance, suggestions and encouragement throughout the

making of the project. I sincerely thank all faculty members of the MCA department for

their valuable suggestions and inspiration.

I wish to express my deep sense of gratitude and sincere thanks to my parents and all

well wishers who have directly and indirectly contributed a lot towards our project work.

**ABSTRACT**

Drowsiness are one of the main causes leading to road accidents. They can be prevented by taking effort to get enough sleep before driving, drink coffee or energy drink, or have a rest when the signs of drowsiness occur. The popular drowsiness detection method uses complex methods, such as EEG and ECG. This method has high accuracy for its measurement but it need to use contact measurement and it has many limitations on driver drowsiness monitor. Thus, it is not comfortable to be used in real time driving. This paper proposes a way to detect the drowsiness signs among drivers by measuring the eye opening and closing rate.

This project describes on how to detect the eyes in a video recorded. In the video, a participant will drive the driving simulation system and a webcam will be place in front of the driving simulator. The video will be recorded using the webcam to see the transition from awake to drowsy. The designed system deals with detecting the face area of the image captured from the video. The purpose of using the face area so it can narrow down to detect eyes within the face area. Once the face is found, the eyes are found by creating the eye for left and right eye detection.

The parameters of the eyes detection are created within the face image. From there, locating the eyes can be performed. Once the eyes are located, measuring the intensity changes in the eye area determine the eyes are open or closed. If the eyes are found closed for 4 consecutive frames, it is confirm that the driver is in drowsiness condition

**CHAPTER 1**

**INTRODUCTION**

INTRODUCTION

Drowsiness is a state of near sleep, where the person has a strong desire for sleep. It has two distinct meanings, referring both to the usual state preceding falling asleep and the chronic condition referring to being in that state independent of a daily rhythm . Sleepiness can be dangerous when performing tasks that require constant concentration, such as driving a vehicle. When a person is sufficiently fatigue while driving, they will experience drowsiness and this leads to increase the factor of road accident.

The development of technologies for detecting or preventing drowsiness while driving is a major challenge in the field of accident avoidance system. Because of the hazard that drowsiness presents on the road, methods need to be developed for counteracting its affects. The aim of this project is to develop a simulation of drowsiness detection system. The focus will be placed on designing a system that will accurately monitor the open or closed state of the driver’s eyes. By monitoring the eyes, it is believed that the symptoms of driver's drowsiness can be detected in sufficiently early stage, to avoid a car accident.

Detection of drowsiness involves a sequence of images of a face, and the observation of eyes open or closed duration. Another method to detect eye closure is PERCLOS. This detection method is based on the time of eyes closed which refers to percentage of a specific time.

The analysis of face images is a popular research area with applications such as face recognition, and human identification and tracking for security systems. This project is focused on the localization of the eyes, which involves looking at the entire image of the face, and determining the position of the eyes, by applying the existing methods in image processing algorithm. Once the position of the eyes is located, the system is designed to determine whether the eyes are opened or closed, and drowsiness.

**CHAPTER 2**

**SYSTEM ANALYSIS**

**2. SYSTEM ANALYSIS**

System analysis is a detailed study of various operation performed by a system and their relationship within and outside of the system. System analysis definition simply means “figuring out what to make it before”.

System analysis can be categorized into five parts:

* System planning and investigation
* Information gathering
* Applying analysing tools for structured analysis
* Feasibility study
* Cost/Benefit analysis

**2.1 REQUIREMENT ANALYSIS**

Requirement analysis is the process of gathering and interpreting facts, diagnosing the problems using the information to recommend improvements on system analysis is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to be minute test detail and analysed. The system analyser plays the role of an interrogator well deep into the working of the present system. The system is viewed as a whole and inputs from the organization are traced through various phases of the processing of the inputs.

**2.2 EXISTING SYSTEM**

Drowsiness is a state of near sleep, where the person has a strong desire for sleep. It has two distinct meanings, referring both to the usual state preceding falling asleep and the chronic condition referring to being in that state independent of a daily rhythm . Sleepiness can be dangerous when performing tasks that require constant concentration, such as driving a vehicle. When a person is sufficiently fatigue while driving, they will experience drowsiness and this leads to increase the factor of road accident.

The development of technologies for detecting or preventing drowsiness while driving is a major challenge in the field of accident avoidance system. Because of the hazard that drowsiness presents on the road, methods need to be developed for counteracting its affects.

**2.3 PROPOSED SYSTEM**

In order to solve the problems in the previous system, the existing system is needed to evolve.

Current drowsiness detection systems monitoring the driver’s condition requires complex computation and expensive equipment, not comfortable to wear during driving and is not suitable for driving conditions; for example, Electroencephalography (EEG) and Electrocardiography (ECG), i. e. detecting the brain frequency and measuring the rhythm of heart, respectively. A drowsiness detection system which use a camera placed in front of the driver is more suitable to be use but the physical signs that will indicate drowsiness need to be located first in order to come up with a drowsiness detection algorithm that is reliable and accurate.

Therefore, this project aims to analyse all the previous research and method, hence propose a method to detect drowsiness by using video or webcam. It analyses the video images that have been recorded and come up with a system that can analyses each frame of the video.

**3.1 Overview of Python**

**CHAPTER 3**

**PROGRAMMING ENVIRONMENT**

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages. Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to

7 compile your program before executing it. This is similar to PERL and PHP. Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

**3.1.1 History of Python**

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages. Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL). Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

**3.1.2 Python Features**

Python's features include –

* Easy-to-learn − Python has few keywords, simple structure, and a clearly defined syntax.This allows the student to pick up the language quickly.
* Easy-to-read − Python code is more clearly defined and visible to the eyes
* Easy-to-maintain − Python's source code is fairly easy-to-maintain.
* A broad standard library − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
* Interactive Mode − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
* Portable − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
* Extendable − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
* Databases − Python provides interfaces to all major commercial databases.
* GUI Programming − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix
* Scalable − Python provides a better structure and support for large programs than shell scripting. Apart from the above-mentioned features, Python has a big list of good features, few are listed below –

It supports functional and structured programming methods as well as OOP.

It can be used as a scripting language or can be compiled to byte-code for building large applications.

It provides very high-level dynamic data types and supports dynamic type checking.

It supports automatic garbage collection.

It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java. Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python.

**3.2 FEASIBILITY STUDY**

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden. For feasibility analysis, some understanding of the major requirements for the system is essential.

* Determining the potentials and drawbacks of the existing system.
* Identification of the user requirements and the benefits expected by the user from the resulting system.
* Finding out the various alternatives available.
* Knowing what should incorporate in the new system
* Defining the ingredients and objectives involved in the project.
* Identifying whether the proposed system could meet the end needs of the users.

Three key considerations involved in the feasibility analysis are:

* Economical feasibility
* Technical feasibility
* Operational feasibility

**3.2.1 ECONOMICAL FEASIBILITY**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Our project is economically feasible because it has minimum cost.

**3.2.2 TECHNICAL FEASIBILITY**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. Our system is technically feasible since it does not use more resources than available**.**

**3.2.3 OPERATIONAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of shipping the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**4.1 Software Requirements**

**CHAPTER 4**

**SYSTEM REQUIREMENTS**

Operating System : Windows 2000/XP/7/8

Programming language : Python IDLE

Front-End : GUI (Anaconda Navigator)

Back-End : Python

**4.2 Hardware Requirements**

Processor : Intel Pentium IV/AMD

RAM : 256MB Hard Disk : 40GB Drives : Optional

Display Size : 15” Color Monitor

Screen Resolution : 800x600 pixels

Color Palette : True Color (24 bit)

Keyboard : PC/AT enhanced type

Mouse : Logitech PS/2 port mouse

**5.** **SYSTEM DESIGN**

**CHAPTER 5**

**SYSTEM DESIGN**

Design of a system can be defined as the process of applying various techniques and principles for the purpose of defining a device, a process or system is sufficient details to permit its physical realization. Thus, system design is a solution for 'how to' approach to the creation of a new system. This important phase provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study.

The data design transforms the information domain model created during analysis into the data structure that will be required to implement the software. The architectural design defines the relationship among major structural components into a procedural detail necessary for implementing the system recommended in the feasibility study.

The data design transforms the information domain model created during analysis into the data structure that will be required to implement the software. The architectural design defines the relationship among major structural components into a procedural description of the software. Source code is generated and testing is conducted to integrate and validate the software. Project management point of view software design is conducted in two into data and software architecture. There are two levels of the system design:

* Logical design
* Physical design

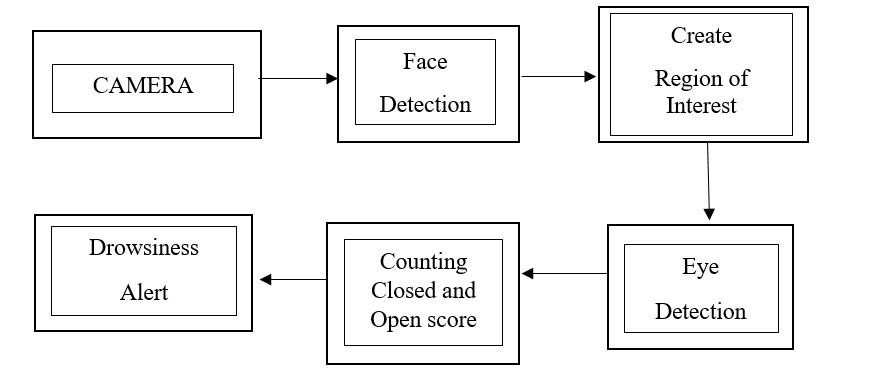
**5.2 INPUT DESIGN**

In my input design, the webcam starts capturing video of face and detect the movement of the eyes. Then if the eyes are closed, a function named closed score will get implemented and it starts to count till the eyes are opened. We can manually close frame and alarm by pressing “q” key.

**5.3 OUTPUT DESIGN**

If the closed score is above 5, then the buzzer gives the user an alert as drowsiness detection. The alarm gets automatically off only after the open score is decremented to zero from the closed score.

**5.1 PROCESS FLOW DIAGRAM**



**CHAPTER 6**

**IMPLIMENTATION**

**6.1 CODING**

The input to the coding phase is the design document. During the coding phase, different modules identify the design document are coded according to the respective module specifications. Recall that the end of the design space, we not only have the module structure of the system but also the module specifications where the data structures and algorithms for each module are specified. Therefore we can say that the objective of the coding phase is transform the design of a system has given by its module specification, into a high level language code and to this code. Our project is coded according to the design specified in design phase.

**6.2 IMPLEMENTATION**

A crucial phase in the system development of cycle is the successful implementation of a new system design. Implementation simply means converting a new system design into operation.

An important aspect of the system analyst’s job is to make sure that the design is implemented to establish the standard. A critical phase in the system development is the implementation of the new system. Implementation is the process of converting a new system design into an operational one. Implementation includes all the activities that take place to convert from the old system into a new one. The new system may be totally new. Replacing an existing system on it may be a major modification to a system currently put into use.

Implementation is the stage of the project when its theoretical design is turned into a working system. At this stage the main work load, the greatest up level and the major impact on existing practices shift to user department. If the implementation stage is not vehicle fully planned and controlled, it can cause errors. Thus it can’t be considered to be the more crucial stage in achieving a successful new stage and in giving the user confidence that the system work and be effective. Our project is successfully implemented by replacing the old system into the proposed system.

**6.3 MODULE LIBRARIES**

A module is a Python object with arbitrarily named attributes that you can bind and reference. Simply, a module is a file consisting of Python code. A module can define functions, classes and variables. A module can also include runnable code. A module allows you to logically organize your Python code. Grouping related code into a module makes the code easier to understand and use.

**6.3.1 CV2**

OpenCV is a huge open-source library for computer vision, machine learning and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human. OpenCV contains implementations of more than 2500 algorithms and it is freely available for commercial as well as academic purposes.

**6.3.2 OS**

The OS module in python provides functions for interacting with the operating system. OS comes under Python’s standard utility modules. This module provides a portable way of using operating system-dependent functionality. The *os* and *os.path* modules include many functions to interact with the file system.

**6.3.3 NumPy**

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely. NumPy stands for Numerical Python. NumPy is a Python library and is written partially in Python, but most of the parts that require fast computation are written in C or C++.

**6.3.4 Mixer**

The *pygame.mixer* module helps control the music used in *pygame* programs. As of now, there are 15 different functions for the *mixer* module. Similar to how you have to initialize *pygame* with *pygame.init()*, you must initialize *pygame.mixer* as well. By using the first option, we initialize the module using the default values. You can though, override these default options. By using the second option, we can initialize the module using the values we manually put in ourselves. To check whether we have initialized it or not, we can use *pygame.mixer.get.init()*, which returns *True* if it is and *False* if it is not. To quit/undo the initializing, simply use *pygame.mixer.quit()*. If you want to continue playing sounds with the module, you might have to reinitialize the module.

**6.3.5 Keras**

Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow. It was developed with a focus on enabling fast experimentation. In this project we import load\_model() function . The function returns the model with the same architecture and weights.

**6.3.6 ImageTk**

The ImageTk module contains support to create and modify Tkinter Bitmap Image and Photo Image objects from PIL images. This can be used everywhere Tkinter expects an image object. The given image must have mode “1”. Pixels having value 0 are treated as transparent. Options, if any, are passed on to Tkinter. The most commonly used option is foreground, which is used to specify the color for the non-transparent parts. See the Tkinter documentation for information on how to specify colours.

**6.4 Eye Detection**

In this project, we use Python language along with the OpenCV library for the algorithm execution and image processing respectively. The haarcascade we are using in the project are pretrained and stored along with the OpenCV library as haarcascade\_frontalface\_default.xml, haarcascade\_lefteye.xml,haarcascade\_righteye.xml

Files.

**CHAPTER 7**

**SYSTEM TESTING**

**7 SYSTEM TESTING**

Testing is the process of executing a program with the intent of hiding any errors. Testing is vital to the success of the system. Without proper testing hide errors will surface after sometimes of use and perhaps irreversible damage has been done to valuable data. A series of tests like series like responsiveness, its value, stress and security are performed before the system is ready for user acceptance testing. System testing follows the logical conclusion that is all the part of the system are tested and found to be working properly under all kinds of situations, and then the system is achieving its goal of processing the data perfectly according to user rules and requirements.

System testing is aimed at ensuring the system works at accurately before live operation commences. Testing is vital to the system. System testing makes a logical assumption that if all parts of the system are correct, the goal will be successfully achieved. The candidate system is subjected to a variety of tests: online response, volume, stress recovery and security and usable tests. A series of testing are performed for the proposed system before the system is ready for user acceptance testing. Nothing is complete without testing, as it is vital success of the system.

**7.1 TYPES OF TESTING**

The testing process contains:

* Unit Testing
* Integration Testing
* Final/System Testing
* Black Box Testing
* White Box Testing
* Validation Testing

**7.1.1 UNIT TESTING**

Unit testing focuses verification on the smallest unit of software design the module. To check whether each module in the software works properly so that it’s desired outputs to the given inputs. All validations and conditions are tested in the module level in the unit test.

Control paths are tested to ensure the information properly flows into, and out of the program unit and out of the program unit under test. Boundary condition is tested to ensure that the modules operate at boundaries. All independent paths through the control structures ensure that all statements in a module have been executed at least once. Unit test is module by module testing. Our project was tested module by module and it was error free

**7.1.2 INTEGRATION TESTING**

Integration testing is the phase of software in which individual software modules are combined and tested as a group. The major concerns of integration testing are developing an incremental strategy that will limit the complexity of entire actions among components as they are added to the system. Developing a component as they are added to the system, developing an implementation and integration schedules that will make the modules available when needed, designing test cases that will demonstrate the viability of the evolving system.

Link between modules are successfully implemented on our project.

**7.1.3 SYSTEM TESTING**

When a system is developed, it is hoped that it performs properly. In practice, however, some errors always occur. The main purpose of testing an information system is to find the errors and correct them. A successful test is one, which finds an error. The main objectives of system testing are:

* To ensure during operation the system will performs as per specification

• To make sure that the system meets user requirements during operation.

* To verify that the controls incorporated in the system function as intended

• To see that when correct inputs are fed to the system the outputs are correct.

* To make sure that during operation, input and output will be deleted. All the above criteria are satisfied in our system.

**7.1.4 BLACK BOX TESTING**

The testing method focuses on the functional requirements of the software. It attempts to find out the error of the following categories such as incorrect and missing functions, interface error, error in the data structure, performance error and initialization and termination errors. Black box testing enables the software engineer to drive sets of input conditions that will fully exercise all functional requirement of a programmer. It is a complementary approach that is likely to uncover a different class of errors than white box testing. Black box testing attempts to find error in the following categories:

* Incorrect or missing functions.
* Interface errors.
* Errors in data structures or external database access
* Performance errors.
* Initialization and termination errors. In our system for the above categories no errors were identified.

**7.1.5 WHITE BOX TESTING**

This testing method is also called path testing. It is a case design method that uses the control structure of the procedural design to drive test case. In this system, unit testing has been successfully handled. The test data was given to each and every module in all respects and got the desired output. Each module has been tested found working properly. White box testing is a test case designs method that uses to control structure of the procedural design to drive test cases. Using white box testing methods software engineer can drive cases that:

* Guarantees that all-independent paths within modules have to be exercised at least once
* Exercise all logical missions divisions on their true or false sides.
* Exercise all loops at boundaries and within their operational boundaries.
* Exercise all internal data structures to ensure their validity
* Basic path testing and controlled structure testing are some of the methods used for white box testing. The above cases are successfully satisfied in our system.

**7.1.6 VALIDATION TESTING**

Data validation is done to see whether the corresponding entries made in the tables are correct. Proper validations are done in case of insertion and updating of tables. If any such cases arises, proper error message or warning, if any, has to be displayed. At the combination of integration testing software is completely assembled as package, interface errors has been uncovered and corrected and a final series of software tests. Validation succeeds when the software functions in a manner that can be reasonably expected by the customer. After validation tests are conducted one of the two possible condition exists. The function or performance characteristics conform to specification and are expected. A deviation from specification is uncovered and deficiency list is created. Proposed system under consideration has been tested by using validation testing and found to be working satisfactorily.

**7.1.7 OUTPUT TESTING**

After performing validation test, the next phase is output test of the system, since no system could be useful if it does not produce desired output in the desired format. The output format was considered in two ways: one is on the screen and the other as a printed form.

**7.1.8 USER ACCEPTANCE TESTING**

User Acceptance Testing (UAT) is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done.

CHAPTER 8

SYSTEM MAINTENANCE

**SYSTEM MAINTENANCE**

Maintenance is any work done to change the system after it is in operational. The maintenance phase of the software life cycle is the time period in which software products perform useful works. In this be retrieve the data from the database design by searching the database. So, for maintaining data project has a backup facility so that there is an additional copy of data, which need to be maintained. They may define software maintenance by describing four activities that a program is released for use.

Maintenance is a characteristic of design and implementation. Which is expressed, as the probability that can an item will be retained in or restored to a specific condition within a given period of time, when maintenance is performed to accordance with the prescribed procedures and resource. Maintenance is the enigma of the system development. It holds the software industry captive, tying up programming resources. Analyst and programmers spend far more time maintaining program that they do writing them.

System maintenance activities can be classified into:

* Corrective maintenance
* Adaptive maintenance
* Perspective maintenance

Corrective maintenance removes software fault. Perspective maintenance improves the system without changing the functionality. The objective of perspective is to prevent failure and optimize the software.

Adaptive maintenance modifies the software to keep it up to data with its opening environment. It may be needed be because of changing in the user requirements, changes in the target platform or changes in external interfaces. Minor changes vehicle rise should be vehicle ride out as a separate development project.

Perspective maintenance means enhancing the performance of modifying the program to respond to the user additional or changing needs.

The first maintenance activity occurs because it is unreasonable to assume that software testing will uncover latent errors in a large software system. During the use of ant large program, errors will occur and reported to the developer. The second activity contributes to a definition of maintenance occurs because of the rapid changes that is encountered in every aspect of computing. Therefore adaptive maintenance activity that modifies software to properly interface with a changing environment is both necessary and common in places. The third activity that may apply to a definition of maintenance occurs when a software package is successful. As the software is used recommendation for new capabilities, modifications to existing functions and general enhancement are received from users. To satisfy requests in this category perfect maintenance is performed. The fourth maintenance activity occurs when software is changed to improve future maintainability or reliability, or to provide a better basic for future enhancements. Often called preventive maintenance this activity is characterized by reverse engineering and re-engineering techniques.

CHAPTER 9

SYSTEM SECURITY

**SYSTEM SECURITY**

System security involves protecting compactor organization data assets. System security professionals work with computer and security programs as well as various hardware to ensure that a business as a company important information is kept secure. System security professional test, implement, maintain and repair software and hardware use to protect information. Security managers in the field direct terms of security professional and might coordinate wide initiative to increase security.

**CHAPTER 10**

**SCREENSHOTS**

**CHAPTER 11**

**FUTURE ENCHANCEMENT**

**FUTURE ENCHANCEMENT**

The future work can include integration of the proposed system with globally used applications like Uber and Ola. The system, if integrated, can reduce the number of casualties and injuries that happen regularly due to these drowsy states of the drivers. This experiment can run as a part of pilot plan i.e. for a few days/months in different regions of the world where such incidents occur regularly. Thus, our proposed approach also gives the same accuracy for the people wearing spectacles. Accuracy of our proposed system improves with the increase in brightness of the surrounding environment. The work can be extended for different types users such as bike riders or in different domains like railways, airlines etc.

This system can be implemented in various classroom application such as Google Meet, Zoom etc. If we implement this system in business domain we can use cloud storage to store the huge amount of output data.

Some model and techniques can be used for various other uses like Netflix and other streaming services can detect when the user is asleep and stop the video accordingly. It can also be used in application that prevents user from sleeping.

**CHAPTER 12**

**CONCLUSION**

**CONCLUSION**

This project aims at developing a software tool for timely and accurate detection of drowsiness in a driver while at the wheel by considering facial features as inputs. By capturing normal image of the user and comparing it with the input feed, the system detects eye closures to check whether the user is drowsy or not. Input is captured by the camera, and the output is in the form of a buzzer that gives the user an alert as and when drowsiness is detected. The limitation that can act as a detriment to the precision of this system would be poor lighting. This can be overcome by using an appropriate lighting system, if required.