MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR (M.P.)

Minor Project

A PROJECT REPORT

on

"DRIVER DROWSINESS DETECTION SYSTEM"

Submitted By:

Aayushi Gupta-

0901CS191002

Devashish Ojha-

0901CS191032

Submitted to:

Prof. Rajni Ranjan Makwana

ABSTRACT

This document is a review report on the research conducted and the project made in the field of computer engineering to develop a system for driver drowsiness detection to prevent accidents from happening because of driver fatigue and sleepiness. The report proposed the results and solutions on the limited implementation of the various techniques that are introduced in the project. Whereas the implementation of the project give the real world idea of how the system works and what changes can be done in order to improve the utility of the overall system.

Furthermore, the paper states the overview of the observations made by the authors in order to help further optimization in the mentioned field to achieve the utility at a better efficiency for a safer road.

Keywords—Driver drowsiness; eye detection; yawn detection; blink pattern; fatigue

Contents

1 Introduction

1.1 PURPOSE . . . . . .

1.1.1 HUMAN PSYCHOLOGY. . . . . .

1.1.2 CURRENT STATISTICS . . . . . .

1.2 DOCUMENT CONVENTIONS . . . . . .

1.3 INTENDED AUDIENCE. . . . . .

1.4 PRODUCT SCOPE. . . . . .

1.5 PROBLEM DEFINITION. . . . . .

2 Literature Survey 2.1

SYSTEM REVIEW . . . . . .

2.2 TECHNOLOGY USED. . . . . .

3 Software Requirements Specification

3.1 PYTHON . . . . . .

3.1.1 LIBRARIES . . . . . .

3.2 OPERATING SYSTEM . . . . . .

3.3 HARDWARE . . . . . .

4 Requirement Analysis

4.1 PYTHON . . . . . .

4.1.1 LIBRARIES . . . . . .

4.2 OPERATING SYSTEM . . . . . .

4.3 HARDWARE . . . . . .

4.3.1 LAPTOP . . . . . .

4.3.2 WEBCAM . . . . . .

5 System Design

5.1 USE CASE MODEL . . . . . .

5.2 ACTIVITY DIAGRAM . . . . . .

5.3 CLASS DIAGRAM . . . . . .

6 Project Planning

6.1 SYSTEM MODEL . . . . . .

7 Implementation

8 Screenshots of Project

8.1 NON-DROWSY STATE . . . . . .

8.2 DROWSY STATE . . . . . .

9 Conclusion and Future Scope

9.1 CONCLUSION . . . . . .

9.2 FUTURE SCOPE . . . . . .

List of Figures

5.1 USE CASE MODEL.....4

5.2 ACTIVITY DIAGRAM.....4

5.3 CLASS DIAGRAM 4

5.3 BLOCK DIAGRAM 18

Chapter 1

Introduction

1.1 PURPOSE

1.1.1 HUMAN PSYCHOLOGY WITH CURRENT TECHNOLOGY

Humans have always invented machines and devised techniques to ease and protect their lives, for mundane activities like traveling to work, or for more interesting purposes like aircraft travel. With the advancement in technology, modes of transportation kept on advancing and our dependency on it started increasing exponentially. It has greatly affected our lives as we know it. Now, we can travel to places at a pace that even our grandparents wouldn't have thought possible. In modern times, almost everyone in this world uses some sort of transportation every day. Some people are rich enough to have their own vehicles while others use public transportation. However, there are some rules and codes of conduct for those who drive irrespective of their social status. One of them is staying alert and active while driving.

Neglecting our duties towards safer travel has enabled hundreds of thousands of tragedies to get associated with this wonderful invention every year. It may seem like a trivial thing to most folks but following rules and regulations on the road is of utmost importance. While on road, an automobile wields the most power and in irresponsible hands, it can be destructive and sometimes, that carelessness can harm lives even of the people on the road. One kind of carelessness is not admitting when we

are too tired to drive. In order to monitor and prevent a destructive outcome from such negligence, many researchers have written research papers on driver drowsiness detection systems. But at times, some of the points and observations made by the system are not accurate enough. Hence, to provide data and another perspective on the problem at hand, in order to improve their implementations and to further optimize the solution, this project has been done.

1.1.2 FACTS & STATISTICS

Our current statistics reveal that just in 2015 in India alone, 148,707 people died due to car related accidents. Of these, at least 21 percent were caused due to fatigue causing drivers to make mistakes. This can be a relatively smaller number still, as among the multiple causes that can lead to an accident, the involvement of fatigue as a cause is generally grossly underestimated. Fatigue combined with bad infrastructure in developing countries like India is a recipe for disaster. Fatigue, in general, is very difficult to measure or observe unlike alcohol and drugs, which have clear key indicators and tests that are available easily. Probably, the best solutions to this problem are awareness about fatigue-related accidents and promoting drivers to admit fatigue when needed. The former is hard and much more expensive to achieve, and the latter is not possible without the former as driving for long hours is very lucrative. When there is an increased need for a job, the wages associated with it increases leading to more and more people adopting it. Such is the case for driving transport vehicles at night. Money motivates drivers to make unwise decisions like driving all night even with fatigue. This is mainly

because the drivers are not themselves aware of the huge risk associated with driving when fatigued. Some countries have imposed restrictions on the number of hours a driver can drive at a stretch, but it is still not enough to solve this problem as its implementation is very difficult and costly.

1.2 DOCUMENT CONVENTIONS

Main Heading Font size: 24 (bold fonts)

Sub-headings Font size: 16 (bold fonts)

Sub-headings Content Font size: 14 (normal fonts)

1.3 INTENDED AUDIENCE

The intended audience for this document are the development team, the project evaluation jury, and other tech-savvy enthusiasts who wish to further work on the project.

1.4 PRODUCT SCOPE

There are many products out there that provide the measure of fatigue level in the drivers which are implemented in many vehicles. The driver drowsiness detection system provides the similar functionality but with better results and additional benefits. Also, it alerts the user on reaching a certain saturation point of the drowsiness measure.

1.5 PROBLEM DEFINITION

Fatigue is a safety problem that has not yet been deeply tackled by any country in the world mainly because of its nature. Fatigue, in general, is

very difficult to measure or observe unlike alcohol and drugs, which have clear key indicators and tests that are available easily. Probably, the best solutions to this problem are awareness about fatigue-related accidents and promoting drivers to admit fatigue when needed. The former is hard and much more expensive to achieve, and the latter is not possible without the former as driving for long hours .

Chapter 2

Literature Survey

2.1 SYSTEM REVIEW

This survey is done to comprehend the need and prerequisite of the general population, and to do as such, we went through different sites and applications and looked for the fundamental data. Based on these data, we made an audit that helped us get new thoughts and make different arrangements for our task. We reached the decision that there is a need of such application and felt that there is a decent extent of progress in this field too.

2.2 TECHNOLOGY USED

a. PYTHON - Python is an interpreted, high-level, general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed AND supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

b. JUPYTER Lab - Project Jupyter is a nonprofit organization created to develop open-source software, open-standards, and services for interactive computing across dozens of programming languages.

c. IMAGE PROCESSING - In computer science, digital image processing is the use of computer algorithms to perform image processing on digital images.

d. MACHINE LEARNING - Machine learning is the scientific study of algorithms and statistical models that computer systems use in order to perform a specific task effectively without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly told.

Chapter 3

Software Requirements Specification

3.1 Python:

- Python 3

3.2 Libraries

- Numpy
- matplotlib
- Pygame
- tensorflow
- opencv

3.3 Operating System

- Windows or mac Hardware

Requirements Specification I. Laptop

with basic hardware.

II. Webcam

Chapter 4

Requirement Analysis

4.1 Python: Python is the basis of the program that we wrote. It utilizes many of the python libraries.

4.2 Libraries:

- Numpy
- Tensorflow :to build model
- Pygame: Used for sounding the alarm
- Opencv: Used to get the video stream from the webcam, etc.

4.3 OS: Program is tested on Windows 10 build 1903 and mac 19.04

4.3 Laptop: Used to run our code.

4.4 Webcam: Used to get the video feed.

Chapter 5

System Design

5.1 USE CASE DIAGRAM

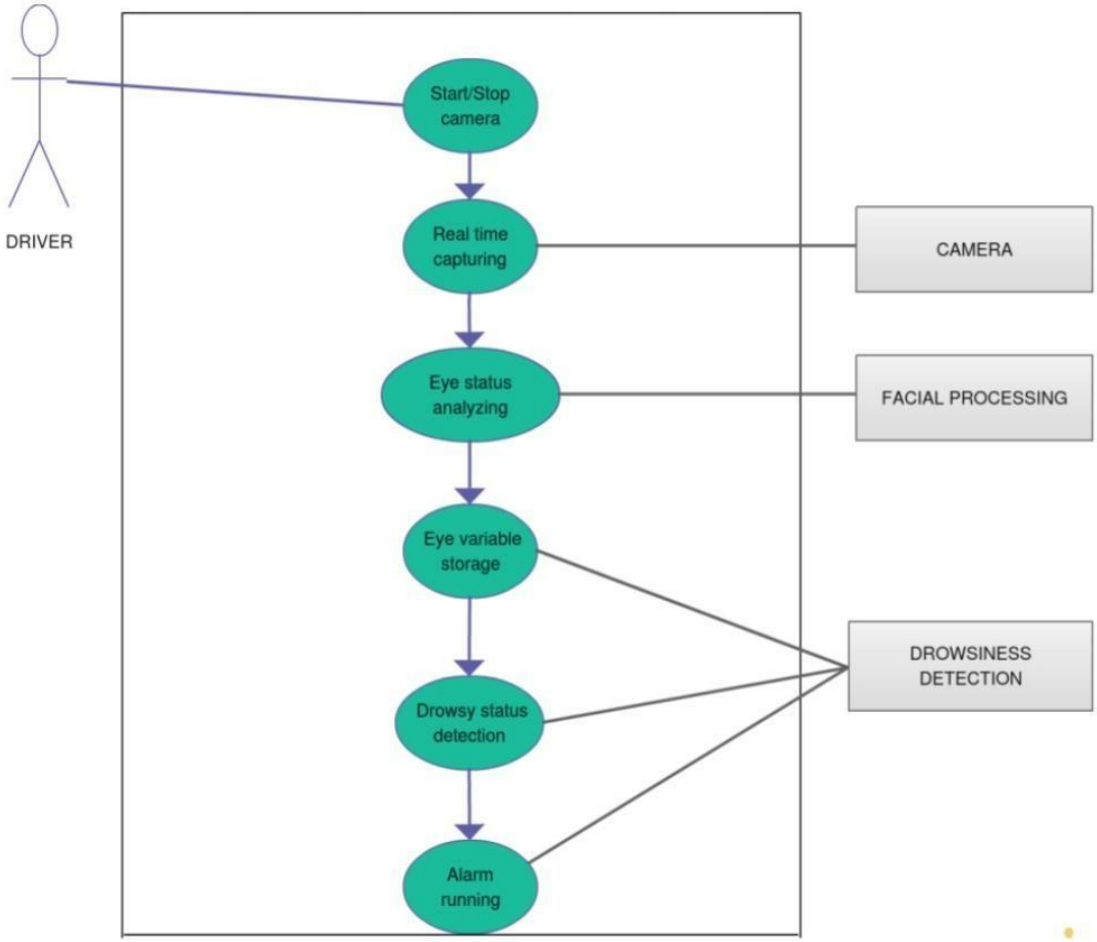


FIGURE 1

5.2 ACTIVITY DIAGRAM

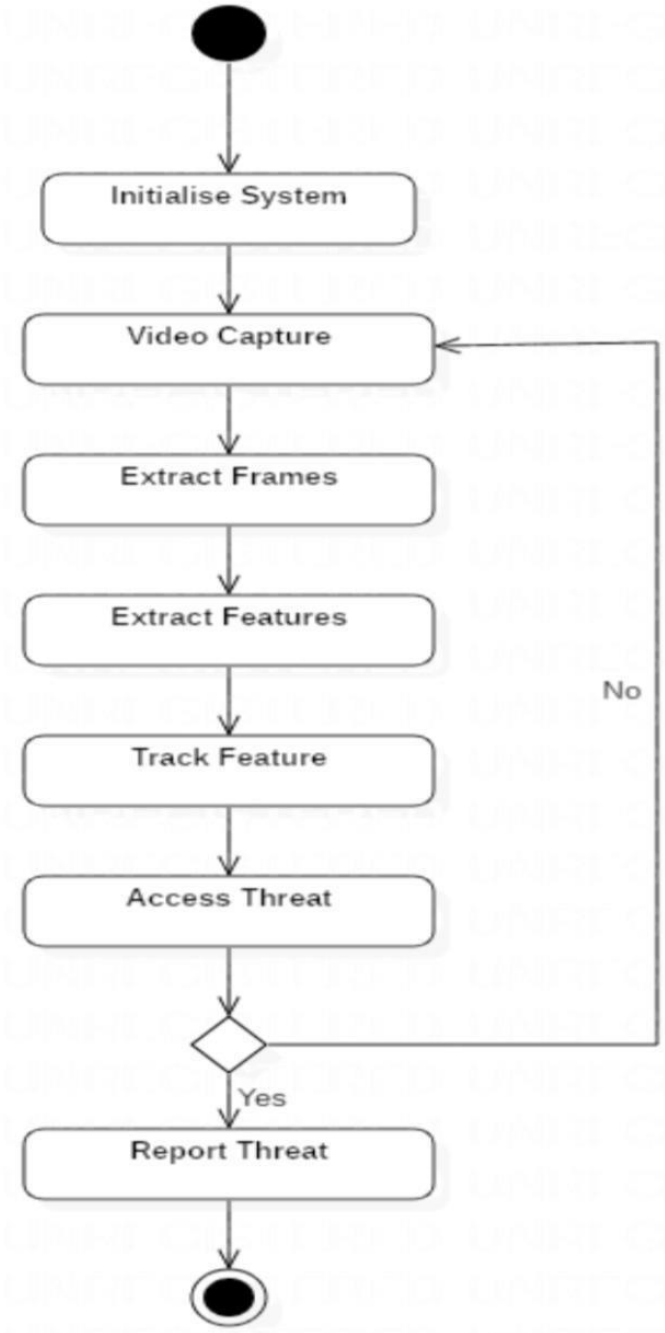
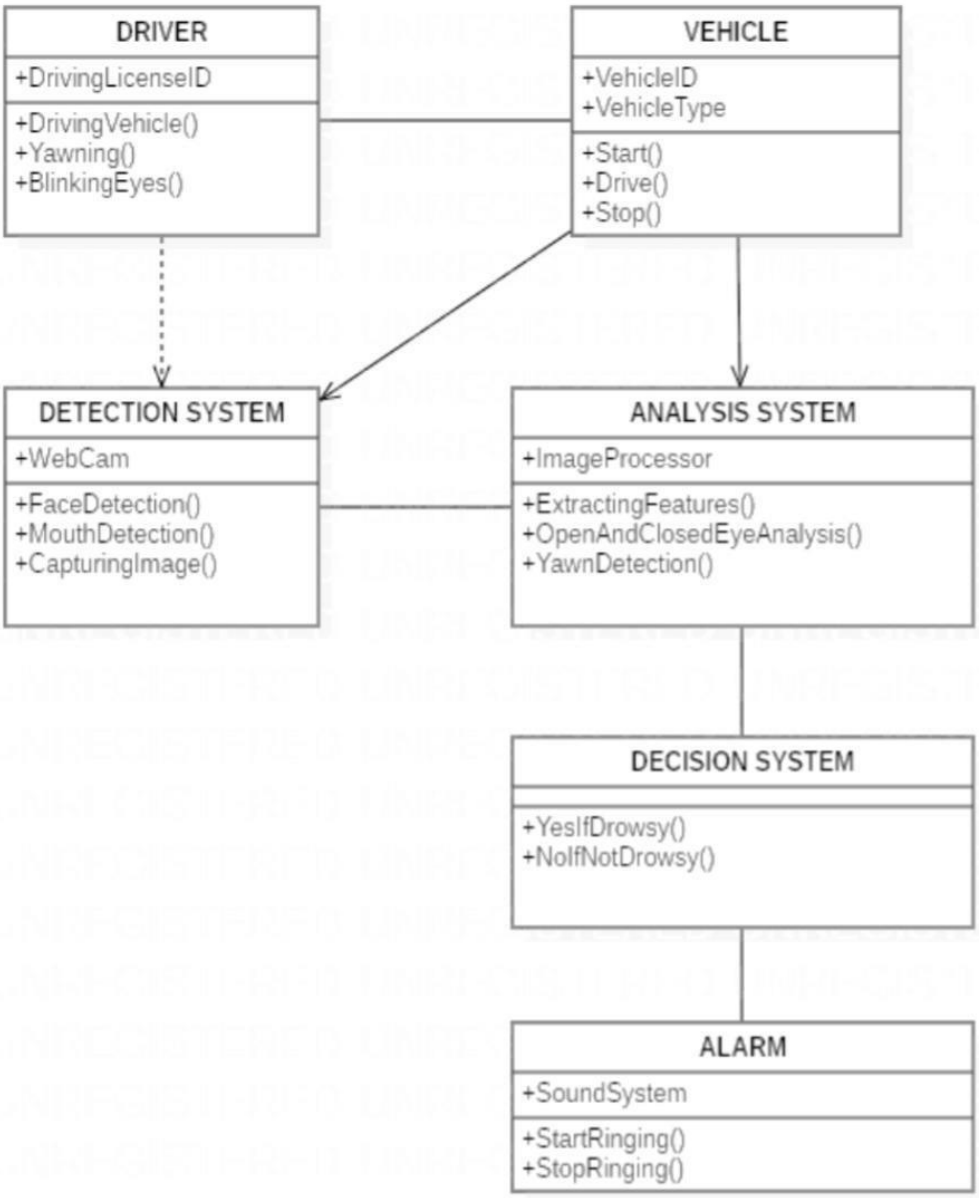


FIGURE 2

5.3 CLASS DIAGRAM



6.Project Planning

6.1 SYSTEM MODEL

The framework is created utilizing the incremental model. The center model of the framework is first created and afterwards augmented in this way in the wake of testing at each turn. The underlying undertaking skeleton was refined into expanding levels of ability.

At the following incremental level, it might incorporate new execution backing and improvement.

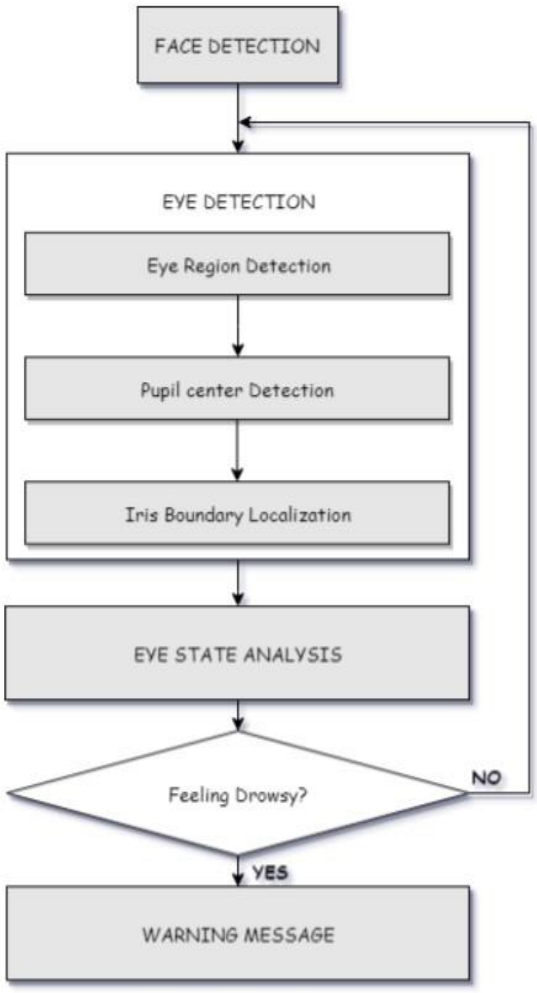


Figure 4: Block diagram

Chapter 7

Implementation

- We have used already trained mobilenet neural network present in tensorflow.keras.applications
- Used eyes dataset to train the model
- Used open cv for face recognition and extracting region of interest and feed the classifier
- Now we check if the aspect ratio value is less than 0.25 (0.25 was chosen as a base case after some tests). If it is less an alarm is sounded and user is warned.

8 Screenshot of Project-

- Creating train dataset

converting all images to (244,244)

```
In [7]: training_Data = []

        def create_training_Data():
            for category in Classes:
                path = os.path.join(DataDirectory , category)
                class_num = Classes.index(category) # 0 or 1
                for img in os.listdir(path):
                    try:
                        img_array = cv2.imread(os.path.join(path,img), cv2.IMREAD_GRAYSCALE)
                        backtorgb = cv2.cvtColor(img_array,cv2.COLOR_GRAY2RGB)
                        new_array = cv2.resize(backtorgb ,(img_size,img_size))
                        training_Data.append([new_array,class_num])
                    except Exception as e:
                        pass

In [8]: create_training_Data()

In [9]: print(len(training_Data))
        2890

In [10]: X = []
         y = []
         for features,label in training_Data:
             X.append(features)
             y.append(label)

         X=np.array(X).reshape(-1, img_size,img_size,3)
         X=X/255.0 #normalizing the data
         y = np.array(y)

In [11]: X.shape,y.shape
Out[11]: ((2890, 224, 224, 3), (2890,))
```

- Training for 10 epochs or number of passes

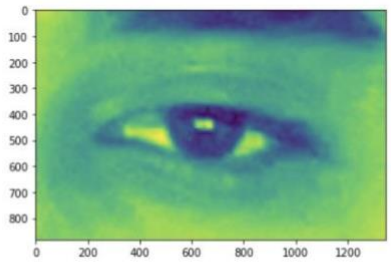
```
In [22]: my_model.fit(X,y, epochs=10,validation_split = 0.1) ## training the model
         2021-11-17 14:08:40.670647: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:185] None of the MLIR Optim
         ization Passes are enabled (registered 2)
         2021-11-17 14:08:40.688483: W tensorflow/core/platform/profile_utils/cpu_utils.cc:128] Failed to get CPU frequency
         : 0 Hz
         Epoch 1/10
         2021-11-17 14:08:41.399357: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:112] Plugin o
         ptimizer for device_type GPU is enabled.
         82/82 [==============================] - ETA: 0s - loss: 0.0902 - accuracy: 0.9758
         2021-11-17 14:09:17.958977: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:112] Plugin o
         ptimizer for device_type GPU is enabled.
         82/82 [==============================] - 39s 457ms/step - loss: 0.0902 - accuracy: 0.9758 - val_loss: 0.1030 - val
         _accuracy: 0.9516
         Epoch 2/10
         82/82 [==============================] - 35s 428ms/step - loss: 0.0163 - accuracy: 0.9958 - val_loss: 0.1879 - val
         _accuracy: 0.9135
         Epoch 3/10
         82/82 [==============================] - 35s 422ms/step - loss: 0.0035 - accuracy: 0.9988 - val_loss: 0.2657 - val
         _accuracy: 0.9170
         Epoch 4/10
         82/82 [==============================] - 36s 434ms/step - loss: 0.0021 - accuracy: 0.9988 - val_loss: 0.0492 - val
         _accuracy: 0.9792
         Epoch 5/10
         82/82 [==============================] - 35s 431ms/step - loss: 0.0083 - accuracy: 0.9977 - val_loss: 0.1788 - val
         _accuracy: 0.9654
         Epoch 6/10
         82/82 [==============================] - 35s 428ms/step - loss: 0.0165 - accuracy: 0.9958 - val_loss: 4.8743 - val
         _accuracy: 0.0000e+00
         Epoch 7/10
         82/82 [==============================] - 36s 434ms/step - loss: 0.0121 - accuracy: 0.9962 - val_loss: 0.1162 - val
         _accuracy: 0.9689
         Epoch 8/10
         82/82 [==============================] - 36s 438ms/step - loss: 0.0045 - accuracy: 0.9985 - val_loss: 0.0158 - val
         _accuracy: 1.0000
         Epoch 9/10
         82/82 [==============================] - 37s 447ms/step - loss: 0.0069 - accuracy: 0.9973 - val_loss: 0.0398 - val
         _accuracy: 0.9758
         Epoch 10/10
         82/82 [==============================] - 38s 458ms/step - loss: 0.0036 - accuracy: 0.9985 - val_loss: 0.0106 - val
         _accuracy: 0.9965
Out[22]: <keras.callbacks.History at 0x16513ec10>
```

using trained model

In [24]:
```
model = tf.keras.models.load_model('my_model.h5')
```

In [25]:
```
img_array = cv2.imread( '/Users/devashish/Documents/Drowsiness_Detection_Tranfer_Learning_Algo/Screenshot 2021-11-1
plt.imshow(img_array)
```

Out[25]:
```
<matplotlib.image.AxesImage at 0x29d58c8b0>
```



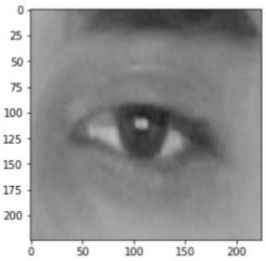
In [26]:
```
backtorgb = cv2.cvtColor(img_array, cv2.COLOR_GRAY2RGB)
new_array = cv2.resize(backtorgb, (img_size,img_size))
```

In [27]:
```
X_input = np.array(new_array).reshape(1,img_size,img_size,3)
```

In [28]:
```
X_input.shape
```

Out[28]:
```
(1, 224, 224, 3)
```

In [29]:
```
_ = plt.imshow(new_array)
```



In [30]:
```
X_input=X_input/255.0
```

In [31]:
```
prediction = model.predict(X_input)
print('done')
```

```
done
```

```
2021-11-17 14:14:42.988588: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:112] Plugin o
ptimizer for device_type GPU is enabled.
```

In [39]:
```
if(prediction<0.5):
    print('Closed Eye')
```

```
Closed Eye
```

Chapter 9

Conclusion and Future Scope

10.1 Conclusion

It completely meets the objectives and requirements of the system. The framework has achieved an unfaltering state where all the bugs have been disposed of. The framework cognizant clients who are familiar with the framework and comprehend it's focal points and the fact that it takes care of the issue of stressing out for individuals having fatigue-related issues to inform them about the drowsiness level while driving.

10.2 Future Scope

The model can be improved incrementally by using other parameters like blink rate, yawning, state of the car, etc. If all these parameters are used it can improve the accuracy by a lot.

We plan to further work on the project by adding a sensor to track the heart rate in order to prevent accidents caused due to sudden heart attacks to drivers.

Same 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.