

PROJECT TITLE

Project Synopsis

Major Project (ICI651)

Degree

BACHELOR OF COMPUTER APPLICATION

PROJECT GUIDE:

VIJAY PRAKASH

SUBMITTED BY:

ABHAY SAXENA (TCA2056003)

VIKESH KUMAR(TCA2056026)

BOBY SHARMA (TCA2056007)

MARCH, 2023

FACULTY OF ENG
TEERTHANKER MA



JTING SCIENCES
FY,MORADABAD

Table of Contents

1	Project Title.....	3
2	Domain.....	3
3	Problem Statement.....	3
4	Project Description	3
4.1	Scope of the Work	3
4.2	Project Modules.....	4
5	Implementation Methodology	4
6	Technologies to be used	4
6.1	Software Platform.....	4
6.2	Hardware Platform	5
6.3	Tools.....	5
7	Advantages of this Project	5
8	Future Scope and further enhancement of the Project	5
9	Team Details	6
10	Conclusion.....	6
11	References	6

1 Project Title

Driver drowsiness detection system

2 Domain

Python development application

3 Problem Statement

The objectives of this intermediate python project is to build a drowsiness detection system that can detect that a person's eyes are closed for a few seconds .

4 Project Description

With this Python project, we will be making a drowsiness detection system. A countless number of people drive on the highway day and night. Taxi drivers, bus drivers, truck drivers and people traveling long-distance suffer from lack of sleep. Due to which it becomes very dangerous to drive when feeling sleepy.

The majority of accidents happen due to the drowsiness of the driver. So, to prevent these accidents we will build a system using Python, OpenCV, and Keras which will alert the driver when he feels sleepy.

4.1 Scope of the Work

The dataset used for this model is created by us. To create the dataset, we wrote a script that captures eyes from a camera and stores in our local disk. We separated them into their respective labels 'Open' or 'Closed'. The data was manually cleaned by removing the unwanted images which were not necessary for building the model. The data comprises around 7000 images of people's eyes under different lighting conditions. After training the model on our

dataset, we have attached the final weights and model architecture file “models/cnnCat2.h5”.

4.2 Project Modules

The CNN model architecture consists of the following layers:

- Convolutional layer; 32 nodes, kernel size 3
- Convolutional layer; 32 nodes, kernel size 3
- Convolutional layer; 64 nodes, kernel size 3
- Fully connected layer; 128 nodes

The final layer is also a fully connected layer with 2 nodes. A Relu activation function is used in all the layers except the output layer in which we used softmax.

5 Implementation Methodology

The requirement for this Python project is a webcam through which we will capture images. You need to have Python (3.6 version recommended) installed on your system, then using pip, you can install the necessary packages.

OpenCV – pip install opencv-python (face and eye detection).

TensorFlow – pip install tensorflow (keras uses TensorFlow as backend).

Keras – pip install keras (to build our classification model).

Pygame – pip install pygame (to play alarm sound).

5.1 Software Platform

- a) Open Cv
- b) Tensor Flow
- c) Keras
- d) Pygame

5.2 Hardware Platform

Ram – 8 gb

Hard disk – 512 gb

Processor - i5

5.3 Tools

OpenCV – pip install opencv-python (face and eye detection).

TensorFlow – pip install tensorflow (keras uses TensorFlow as backend).

Keras – pip install keras (to build our classification model).

Pygame – pip install pygame (to play alarm sound).

6 Advantages of this Project

With this Python project, we will be making a drowsiness detection system. A countless number of people drive on the highway day and night. Taxi drivers, bus drivers, truck drivers and people traveling long-distance suffer from lack of sleep. Due to which it becomes very dangerous to drive when feeling sleepy.

The majority of accidents happen due to the drowsiness of the driver. So, to prevent these accidents we will build a system using Python, OpenCV, and Keras which will alert the driver when he feels sleepy.

7 Future Scope and further enhancement of the Project

To detect the face in the image, we need to first convert the image into grayscale as the OpenCV algorithm for object detection takes gray images in the input. We don't need color information to detect the objects. We will be using haar cascade classifier to detect faces. This line is used to set our

classifier **face = cv2.CascadeClassifier(' path to our haar cascade xml file')**. Then we perform the detection using **faces = face.detectMultiScale(gray)**. It returns an array of detections with x,y coordinates, and height, the width of the boundary box of the object. Now we can iterate over the faces and draw boundary boxes for each face.

8 Team Details

Project Name & ID	Course Name	Student ID	Student Name	Role	Signature
DRIVER DROWSINESS DETECTION SYSTEM	MAJOR PROJECT ICI651	TCA2056026	VIKESH KUMAR	Developer, Leader	VIKESH KUMAR
		TCA2056003	ABHAY SAXENA	DESIGNER, CO-LEADER	ABHAY SAXENA
		TCA2056005	BOBY SHARMA	TESTER, MEMBER	BOBY SHARMA

9 Conclusion

In this Python project, we have built a drowsy driver alert system that you can implement in numerous ways. We used OpenCV to detect faces and eyes using a haar cascade classifier and then we used a CNN model to predict the status

10 References

Adrian Rosebrock, PyimageSearch Blog