

Ai Monitoring and Response System for Anti-Accident Mechanism

To counter present day challenges of accidents, specially in hilly areas, we aim to build an AI mechanism which inculcates Deep Learning Algorithms that autonomously detect, prevent and alert on any possibility of accidents in the most viable & effective way possible.



OUR TEAM



SHIVAM



SARTHAK



YUVRAJ



RAJVEER

Direction And Roadmap



01 Problem Statement

Describing the Real-Life issues faced and explaining how they impact an individual in real life



02 Introduction And Motivation

Getting to know more on project we will be working on and how we faced the problem and developed motivation on building the model.



03 Solution Proposed

The ideal solution required to solve the concern and address the issue from its roots.



01 Result Obtained

Showing the outcome of our project and the output of our code.



05 Step-By-Step Solution

Describing the Real-Life issues faced and explaining how they impact an individual in real life



04 Our Solution And USP

Providing our solution to the concern and providing appropriate USP's that improvises our model and differentiates from existing.



01 Thank You

Gratitude for giving us this opportunity to present our work and participate in hackathon at IIT Mandi.



PROBLEM STATEMENT

Why do we need this App?



The issue of driver drowsiness, distraction, and inattentiveness during driving poses a significant risk to road safety. According to the National Highway Authority of India (NHAI), driver fatigue and distraction are major factors in approximately 20% of all fatal car crashes in India. The need is to monitor driver behavior and alert them when signs of fatigue or distraction are detected. Additionally, the model should also suggest actions such as rest breaks or changes in lighting or music to prevent further distraction which would improve road safety and save lives.

WELCOME WELCOME

Introduction and Motivation

This Model is based on a Deep Learning Algorithm that detects , prevents, analyses behaviour and predicts and alarm the Driver for any potential upcoming accidents through his facial gestures and head movements.



Coming from Chandigarh to NIT Hamirpur at 3 AM, we realised that the driver was dizzy on the curvy roads of Himachal Pradesh. This is when we realised the need of this Application to not only be easily accessible but also in the most convenient and inexpensive form.

NEED OF THE HOUR

SOLUTION PROPOSED

Ai Monitoring and Response System for Anti-Accident Mechanism

The Web-App will monitor driver's face, eyes and head gesture movements to recognise his behaviour , alert and prompt on identifying unusual activity such as closed eyes and send suggestions or emergency notices as required.



KEY FEATURE 01

Web-Application to Monitor & Respond Behaviour using Gesture Movements

App to Monitoring Driver's eye, face and head gesture and beep alert on any unusual or unsafe practice.



KEY FEATURE 02

High Accuracy of Model, offline functionality and 0 cost of functioning

The model works at ~95% accuracy, without any internet connectivity and no external instrument required.



KEY FEATURE 03

AI Based Recommendation System to generate prompt on users' behaviour.

Generating prompt to avoid user from getting distracted or confused such as taking breaks, etc.



KEY FEATURE 04

Emergency Situations and Error Handling

Just like cars, we provide error handling of beep alarms and emergency notification to family in event of accident or crisis.

WHY US?



Works Without Internet even in remote areas

Works even without any internet connectivity or network



Exception and Error Handling of resources.

To have manual turn off beep and accident notification system to family members in accident or crisis



Remotely Accessible without any electronic Sensor

No need of additional machineries



Application with 95% Accuracy

Assures to provide one of the best accuracy of model available till date.

OUR USP



Beep Alert during unusual behaviour or gesture

To generate beep alerts during monitoring if the driver is unattentive or distracted.



inexpensive and can be accessed independently

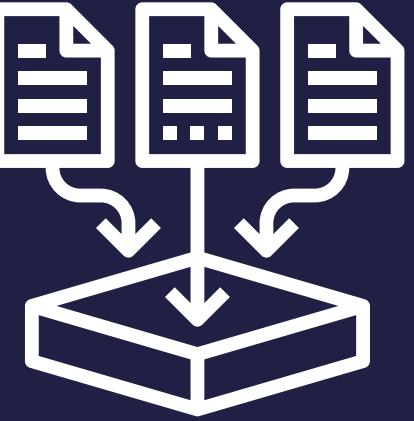
0 cost is required and can be accessed even by a layman



Independent of Steering Movements

No Need to additionally equip device on steering for focus on face.

Step-by-Step Solution



...

To leverage deep learning, the following four steps are crucial: collect relevant data, analyze and understand the data, preprocess it to ensure quality, and finally train and deploy the model. Each step plays a critical role in ensuring the accuracy and efficiency of the model, enabling it to make intelligent predictions and insights.

Step 1: Collection of data

Our First Step involves collection and storing of dataset and then sending it for processing.

Step 2: Analysing the Data

The data will be send to analytics, where the model will be accessed and moderated. This is done to reduce noise as well.

Step 3: Data Pre - Processing

Data-Prepossessing will make it ready to send for training and refining, and will be repeated for better accuracy.

Step 4: Training & Deploying the Network

Data will be trained and deployed and the process will repeat until a good accuracy is achieved.

STEP 1: Collection of Data

MRL EYE DATASET



Gathering and Storage of Data

Dataset is broadly collected and stored for upcoming use in the Deep Learning models.



The data which we use for this project is MRL eye dataset which contains infrared images of human eye in low and high resolution all captured in various lighting conditions.



Packages and Software Used

RESOURCES USED



Gathering and Storage of Data

Dataset is broadly collected and stored for upcoming use in the Deep Learning models.

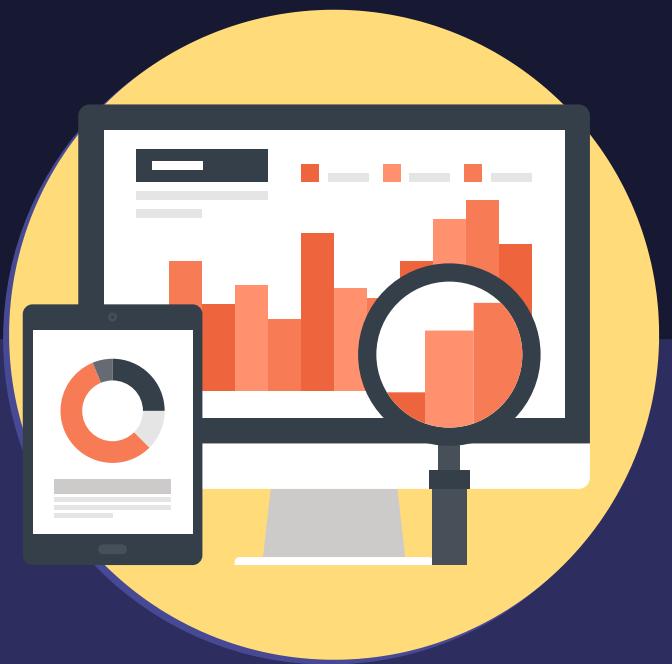
- Jupyter notebook
- Pandas, numpy
- Matplotlib,seaborn
- Os, shutil
- tensorflow,keras
- Skiimage,OpenCv
- Webcam/mobile phone cameras
- Mixer , Alarm.wav file
- Heroku
- Docker
- HTML,CSS, Javascript
- Anaconda navigator
- Git, github, git CLA



STEP 2: Analyzing the Data



PYTHON LIBRARIES



Analyzing the Data

Analysing the stored data to process and feature engineering. This is done to study more on the dataset used.

After getting our dataset our next step will be to analyze the dataset so we can do data processing and feature engineering efficiently. During the analysis we will focus on the answers of below mentioned questions:

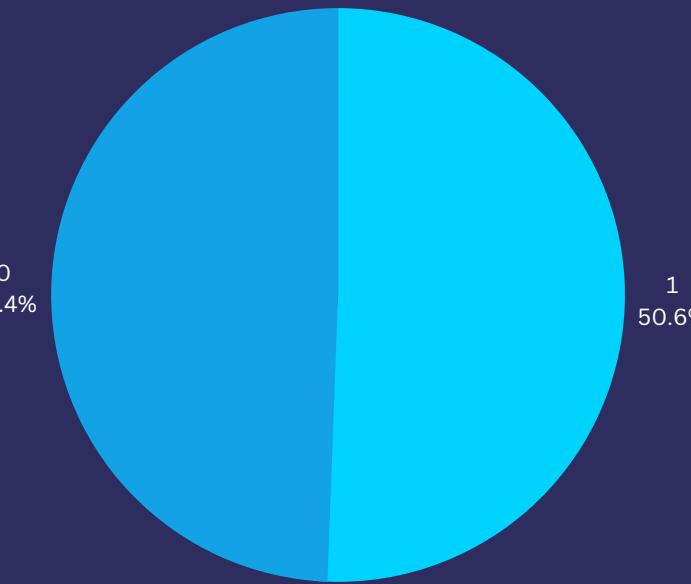
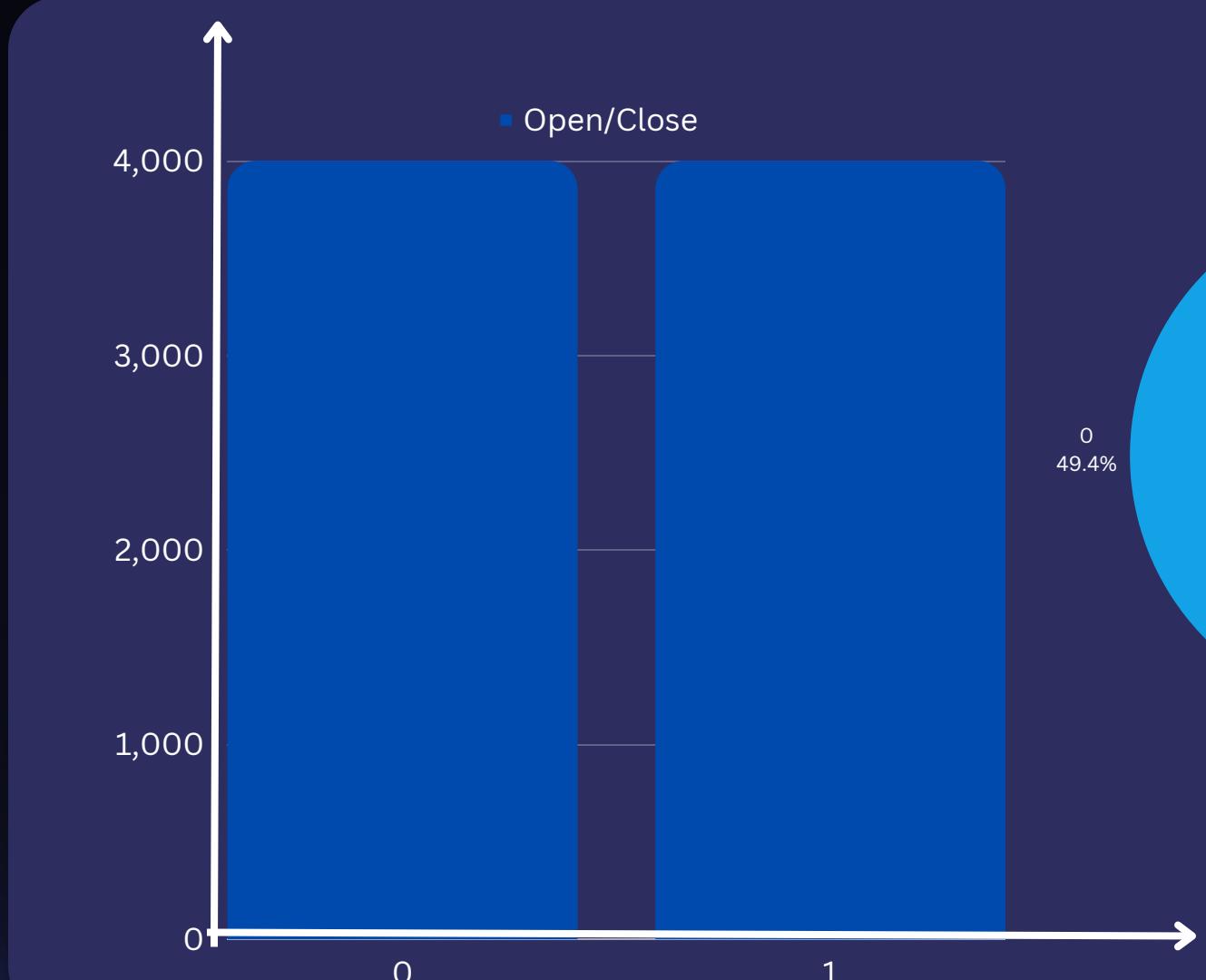
- Shape of dataset
- Various Columns present in the Dataset
- Labelling of distribution
- Type of image (RGB or grayscale)
- Dimension of the image

OPEN EYE

CLOSED EYE

Report After EDA

...



1 76,000 images

2 8 different columns

3 37,000 closed and 39,000 open eye image

4 Gray scale images

Reports successfully matched with ~ 95 percent accuracy.

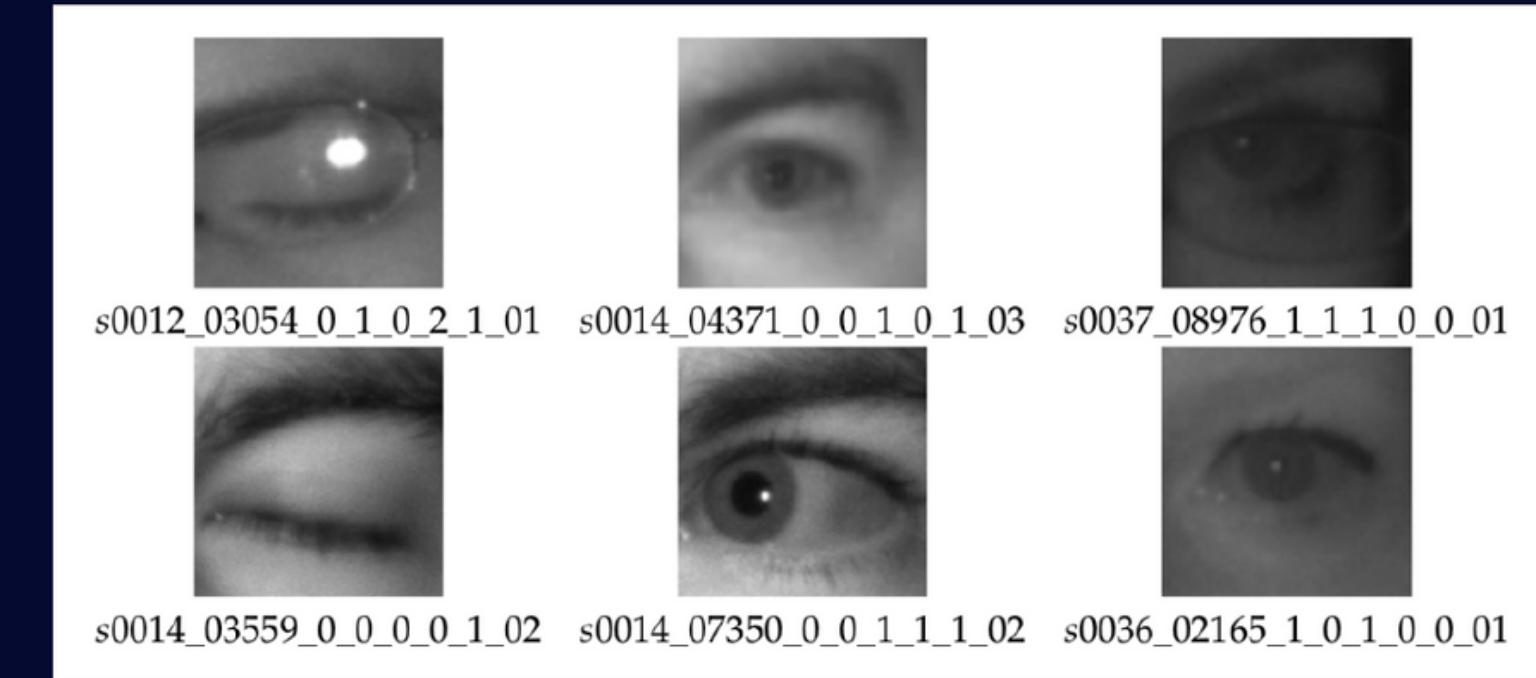
STEP 3: Data Pre - Processing

OPERATIONS ON DATA



Data is Pre-Processed before algorithms are performed.

Data is mainly send to be prepared for deep learning algorithms and operations at this stage.



Dropping the unnecessary columns from the dataset Separating images into open and closed eyes Creation of more data using data augmentation Dividing the data into train ,test and validation data.

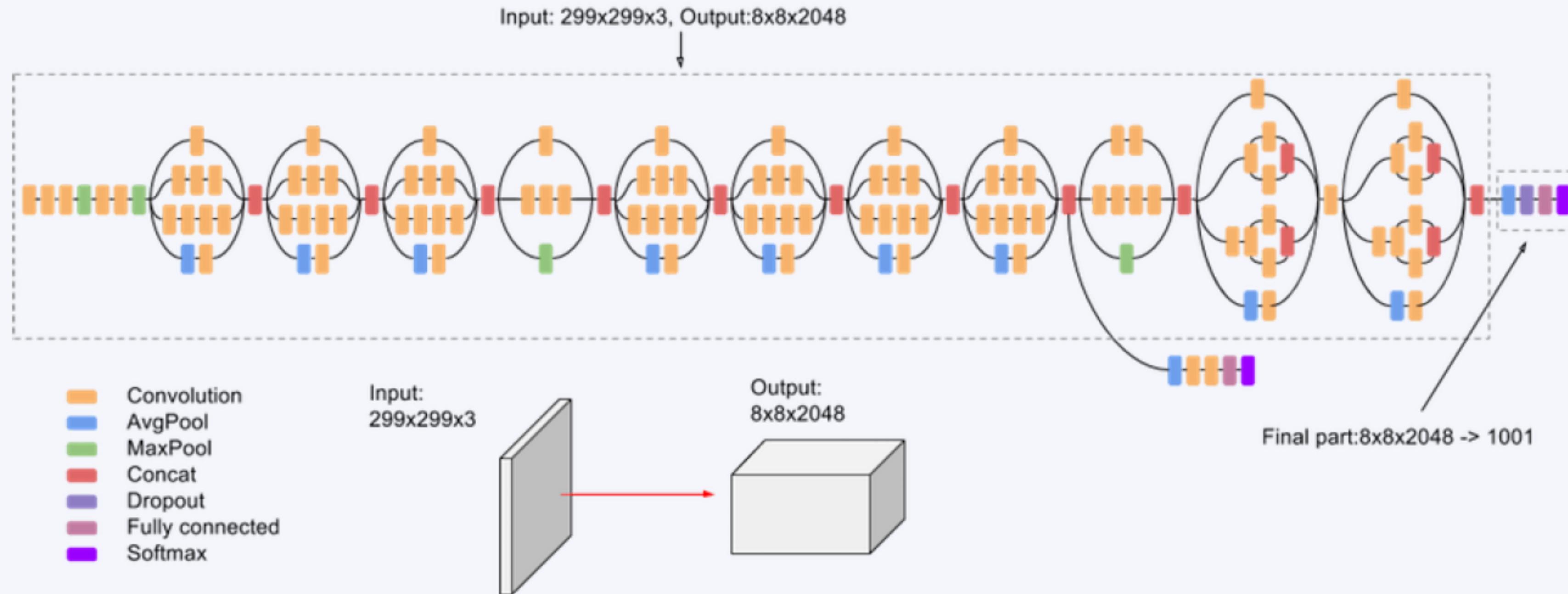
Results of Data Augmentation



- Height_shift
- Width_shift
- Rotation_range
- Zoom_range
- Horizontal_flip



Training Pretrained InceptionV3 CNN model



Pretrained CNN model by google on Imagenet dataset
with million of images and 1000 different classes



STEP 4: Training & Deployment of network



DEEP LEARNING & AI



Gathering and Storage of Data

Dataset is broadly collected and stored for upcoming use in the Deep Learning models.

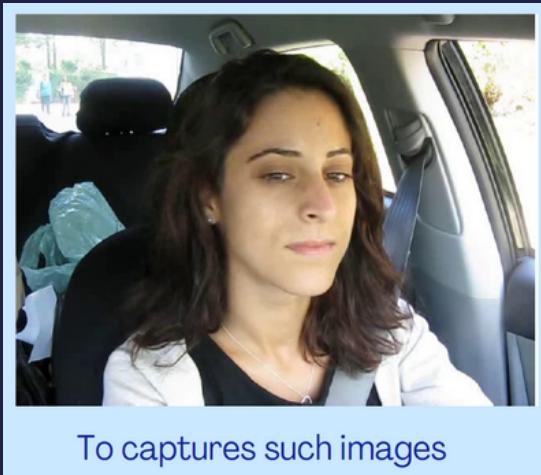
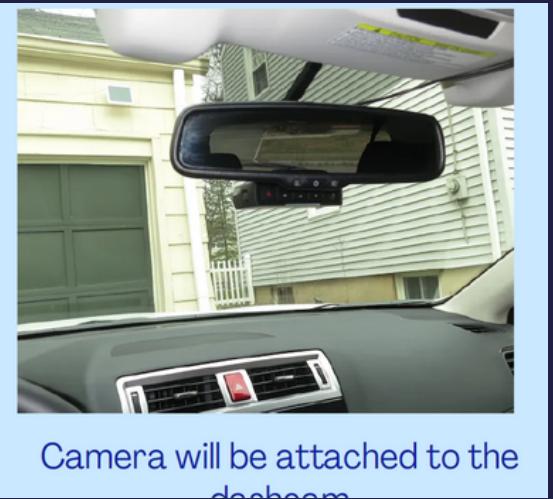
For the training our deep learning architecture we will focus on CNN model and also we will try to combine some pre-trained CNN models for achieving better accuracy. Regarding the pre-trained models we will focus on YOLO, InceptionV3, RetinaNet , Mobilenet, VGG19 and Resnet CNN architectures.

EXPECTED OUTPUT AND CALCULATED OUTPUT



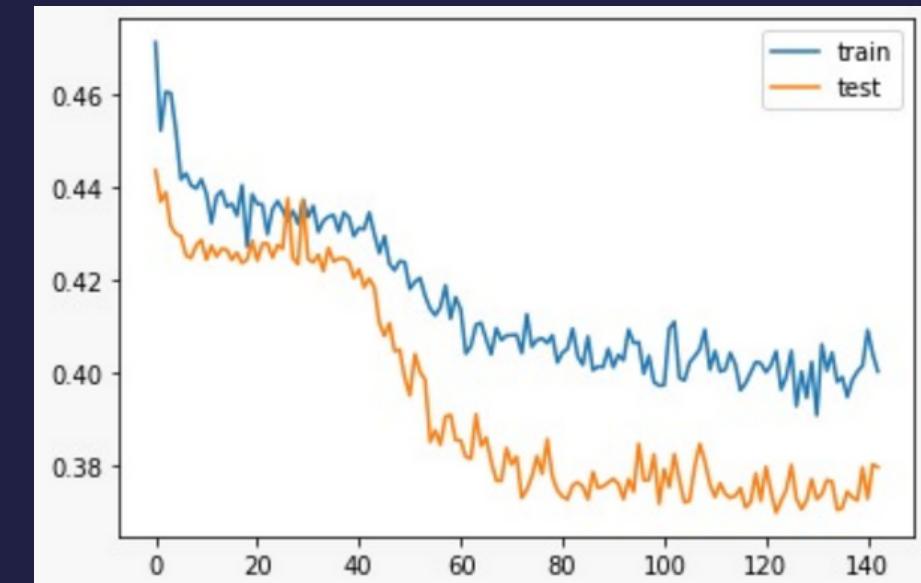
Expected Output

We expect an output that can successfully analyse, predict, notify and alarm the driver based on his/her behavioural changes in gestures.

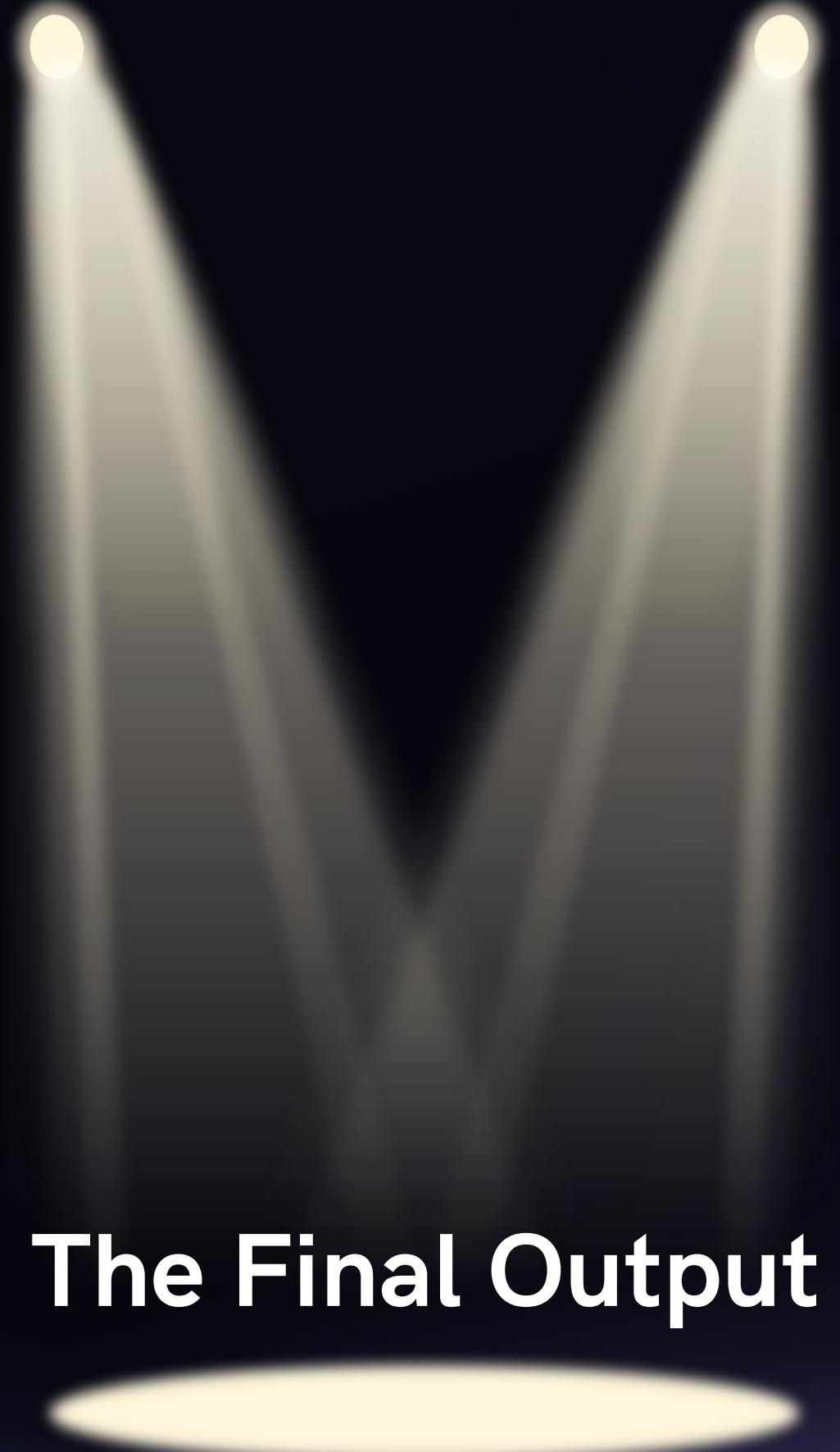


Calculated Output

This process ensures that our model is on the right track and our measures of accuracy are up-to the mark. The output also validates our progress in front of accuracy obtained



And Here Comes...



The Final Output

...



THANK YOU

We are grateful to present our project for this hackathon and expect your valuable feedback to inculcate in our value standards