Journal of Research Administration
Society of Research Administrators International

DRIVER DRIVING SCORE CALCULATION AND DROWSINESS DETECTION USING DEEP LEARNING

Shubham Sanjay Supekar[a,*,1] (Student), Hariharan Rajadurai[a,2] (Professor)

[a]School of Computing Science and Engineering, VIT Bhopal University, Kotri Kalan, Ashta, Near, Indore Road, Bhopal, 466114, India

A B S T R A C T

This initiative was created in order to prevent accidents caused by drowsiness. When the drowsiness in a driver is detected, the system notifies them right away. Subsequently, it records the date, time, and duration of the drowsiness in a database to create a score that indicates how well, the motorist is driving. The project aggressively approaches drivers with high scores to reduce the likelihood of future accidents because prevention is more effective than therapy. To identify drowsiness in real- time, a custom CNN model is used in this project. The Model achieve 98.387% accuracy on the testing dataset. To achieve optimal accuracy under all conditions, the training dataset contains low-light, hazy, and spec as well as reflection-filled images and this is done by using the Media Research Lab(MRL) eye database. It allows the model to detect drowsiness in real-time video with maximum accuracy under all circumstances.

Keywords: convolutional neural network drowsiness, hyperparameters MySQL

1. Introduction

According to the Ministry of Road Transport and High- ways Transport(MRTH), there will be 1,59,972 deaths and 3,84,448 people injured in truck accidents in India by 2021. According to a World Health Organization(WHO) study, India accounts for around 11 percent of global accidents, most of which are caused by trucks/lorries and Commercial Vehicles(CV). The economic cost of these incidents calcu- lated by Bosch India's Advanced Autonomous Safety Sys- tems and Corporate Research department, is 15.71−38.81 billion, or 0.55% -1.35% of India's GDP. Based on the evidence provided, accidents cause both financial loss and serious injuries. The bulk of incidents involve trucks and Commercial Vehicles(CV), as the drivers of these vehicles must put in long hours to travel great distances and make deadlines. During these, extended sessions, a number of factors can cause drowsiness when driving, including lack of sleep, inattention, using a phone, ignoring traffic signs, and distractions, and driving while under the influence of drugs, alcohol, certain medications, and other substances.

Our research aims to tackle the issue of drowsiness detec- tion by alerting the driver at that time and after each session evaluating a driver's performance by calculating a score. This score will be used to inform decision-making processes that will assist in averting future accidents. Hence, not only does our system record information on the date, time, and duration of each session of sleepiness, but it also warns the driver to prevent accidents when it detects drowsiness.

To detect drowsiness, the chosen approach focuses on the most efficient feature: The human eye. If the eye remains closed for 3 seconds, it is considered a drowsy state.

The National Institute of Aging states that age is the factor for safe driving. Advancing age may contribute to joint stiffness and muscular weakening, potentially impact- ing driving abilities. Visual changes associated with aging can impede peripheral vision and diminish the ability to perceive movement beyond direct eyesight. Age-related al- terations in hearing might affect the recognition of auditory cues essential for safe driving, including sirens and vehicle sounds. Furthermore, certain medications commonly used by older individuals may induce drowsiness or reduced alert- ness, posing risks while driving. Slowing reflexes with age could compromise the ability to react promptly in driving scenarios.

According to a study by the National Highway Traffic Safety Administration, diabetes can significantly influence blood glucose levels, leading to potential symptoms like drowsiness, dizziness, confusion, blurred vision, loss of con- sciousness, or seizures. Over time, diabetes may engender issues that impact driving capabilities, potentially causing nerve damage in the extremities or eyes. In severe cases, diabetes-related complications might result in blindness or necessitate amputation.

To calculate the score, age, diabetes status (yes or no), and the duration of drowsiness in each driving session are considered. Considering these safety implications, a 10% increase is applied to the score and for drivers above 50 years old. Additionally, if the driver has diabetes, a 20% increment is allocated to the computed score.

2. Literature Review

In (1) this research paper, the Night-Time Yawning- Microsleep-Eyeblink-driver Distraction (NITYMED) database is used on this database model achieving 99.71% accuracy in the detection of Drowsiness. MediaPipe Face Mesh is used for ROI selection; it projects 468 landmarks for face and eye detection. This proposed model uses three CNN architectures: InceptionV3, VGG16, and ResNet50V2, and trained these architectures on 6800 images. In this research, the data for training and testing is too low, for testing author uses only 1020 images.

In (2) this research paper, the UTA RealLife Drowsiness Dataset (UTA-RLDD) is used.DLIB's 68 landmark detec- tion is used for ROI selection. For drowsiness detection, the author uses two methods first method is based on a recurrent and convolutional neural network and the other method is deep learning. In both methods, the accuracy is 65 over training data and 60 over testing data. In this paper the accuracy of the model is too low.

In (3) this research project, the author use Support vector machines(SVM), electroencephalogram(EEG) sig- nals. They use Sleep EDF(Expanded) database. Based on the electroencephalogram(EEG) signals Support vector ma- chines(SVM) model is trained. Accuracy of the model is 89% which is too low to implemented in real world applications.

In (4) this research project, the author compares the accuracy of different classifiers these are Support vector machines (SVMs), Random forest classifiers, logistic regres- sion(LR) classifier, Multi-layer Perceptron(MLP) classifier, k-nearest neighbors(KNN) classifier and Quadratic discrim- inant analysis (QDA) author compare the learning rate, accuracy and receiver operating characteristic curve(ROC) of these classifiers Among all of them SVM scored most accuracy of 98 compared to other classifiers.

3. Methodology

3.1. Creating Model

3.1.1. Dataset

This model was created using the MRL(Media Research Lab) eye database. It is a large-scale database on human eyes since it comprises 83,059 images of open and closed eyes in various settings such as eyes with specs, excellent or terrible lighting conditions, reflections, and so on. RealSense SR300, IDS Imaging, and Aptina Imagin sensors are used to collect these pictures. All of the photos are properly captioned. They annotated the photos with information such as gender, eye condition, spectacles, and image quality. For this research, we divide the dataset into two folders: "open" for open-eye photographs and "close" for closed-eye images.

3.1.2. Data Pre-Processing

Because the photos in this collection have been grayscaled and shrunk, we may utilize them directly. For each image, load the open-eye photos into the x variable and the value 1 into the y variable. Repeat the process for close-up photos, loading close-eye photos into the x variable and 0 into the y variable. Normalize the photos using a max-min scaling technique. Normalizing aids in the stabilization of the train- ing process. It prevents the model from being overfitting.

Min-Max Scaling: (Original Value – Minimum Value) / (Maximum Value – Minimum Value)

Then divide the variable X by 255. The photos will then be rescaled between 0 and 1.

float

Stages	Filters	Kernel Size	Activation	Padding	Pool Size
Stage 1	32	3	relu	same	2x2
Stage 2	64	3	relu	same	2x2
Stage 3	128	3	relu	same	2x2

Table 1

Parameters for Convolutional layers

3.1.3. Creating Custom CNN Model

Our custom CNN model contains 12 layers and for understanding these layers are converted into 4 stages:

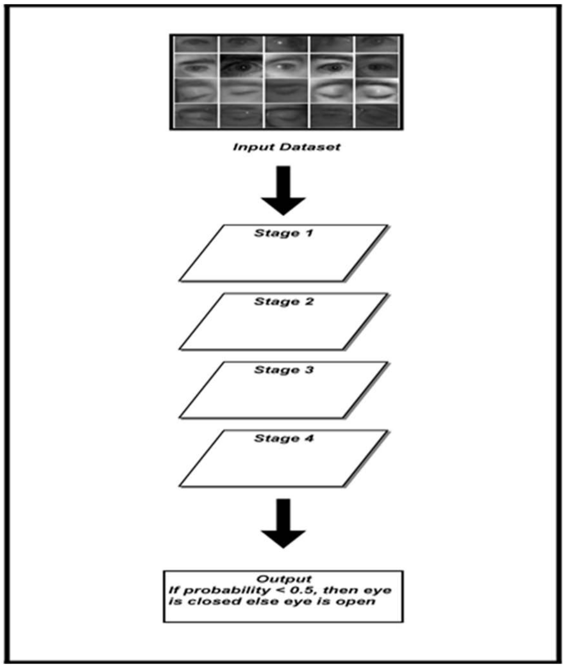

Figure 1: Custom CNN Model

Stage 1,2,3: Each stage consists of two Convolutional Neural Network (CNN), one Batch Normalization, and one Max Pooling layer. These layers are part of the feature ex- traction process in the convolutional neural network, which includes applying multiple convolutional filters, introducing non-linearity through ReLU activation, and downsampling using max pooling. The Batch Normalization layer con- tributes to training stability. These layers collectively con- tribute to the network's ability to learn and extract features from the input data effectively. The hyper-parameters for these layers are:

Stage 4: This is the output stage containing Flatten Layer and Dense Layers. The flattened layer is responsible for converting the 2D or 3D feature maps generated by the convolutional layers into a 1D vector for input to the fully connected layers. Dense layers, sometimes referred to as completely linked layers, give the network non-linearity and carry out linear transformations. The last dense layer in a binary classification makes use of a sigmoid activation

Units	Activation
256	relu
128	relu
1	sigmoid

Table 2

Parameters for output layer

Validation Split	0.2
epochs	100
Batch size	32

Table 3

Parameters for training and testing

function. The layer output for binary classification usually consists of a single neuron with a sigmoid activation func- tion. The hyperparameters are:

These convolutional layers are used to extract features from input images. They scan the feature maps to look for patterns and structures in the data. By using multiple convolutional layers with various filters, the model can cap- ture both high-level and low-level characteristics which will help to identify difficult patterns and structures. All things considered, these convolutional layers are essential building blocks of deep learning models for image processing appli- cations, allowing the model to identify intricate patterns and characteristics in the input.

float

3.1.4. Training and Testing Model

The dataset is divided into two parts: 67% for training and 33% for testing with 100 epochs. In training data, 20% of the data is used for validation. After training of model testing dataset is passed through a model that records its predictions and compares them to the actual values to determine predic- tion accuracy. Parameters for training and testing the model are:

3.2. Detecting Drowsiness

Using Open-CV video can be captured from the camera. After taking a frame from the video it is converted into a grayscale, and then face and eye detection is performed by Haar cascade classifiers. Haar cascade classifiers identify the coordinates of the face and eyes. By passing these coordinates of eyes into our created CNN model it gives a prediction. If it is greater than 0.5 then it is considered as an “open eye” otherwise a “closed eye”.

The code keeps track of “closed eye”. If the eye is closed then the counter and timer both start. For other frames, the eye remains closed, and then the counter increases by 1. If the counter passes its threshold value i.e. 4 and the timer is greater than 3 sec then the state is tired and the alarm is triggered. During this monitoring, If an “open eye” is detected for more than 3 sec, then the alarm is halted, and the drowsiness date, time, and duration are recorded. Fig 2 shows the flow chart of drowsiness detection and score calculation.

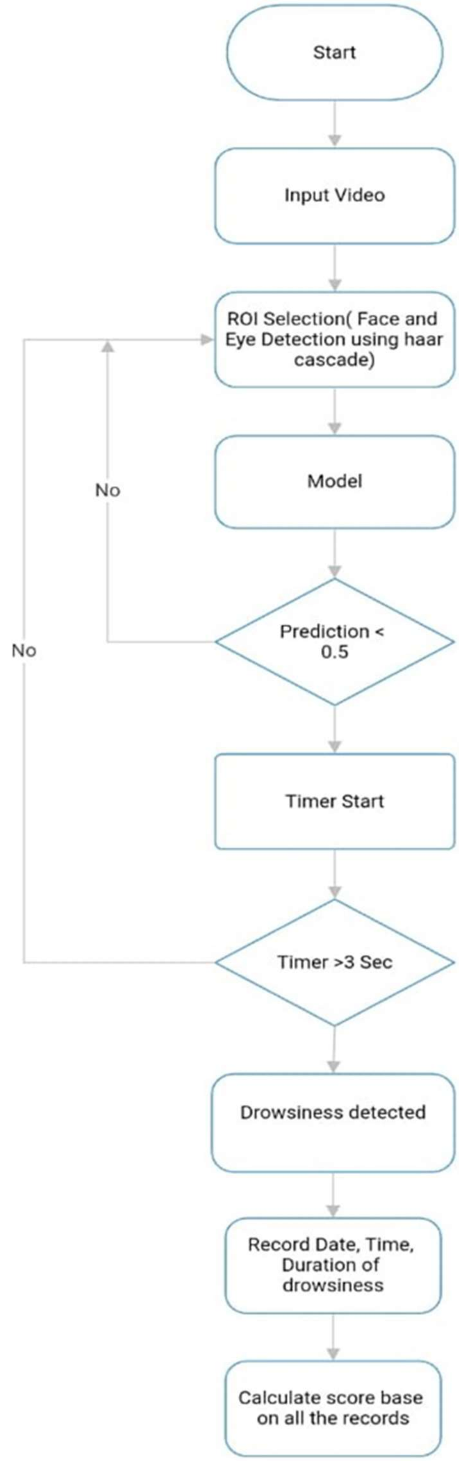

Figure 2: Flow Chart

3.3. Score Calculation

The score is the probability of a driver experiencing drowsiness in overall sessions. This probability is calculated based on past events performed by a driver and based on that we can make a decision to avoid future accidents. Initially, the drowsiness probability for each session is calculated by the formula:

Session Score = (Total Drowsiness Duration in a session/ Total Session Duration)*100

Following this calculation the driver's age is considered. If the driver's age is over 50 then the session score is increased by 10%. This adjustment is done based on the study's finding, that aging has an impact on car handling, observation, signaling, positioning, and traffic flow assess- ment:

Age>50: Session Score = Session Score + (Session Score*0.1)

The driver's diabetic condition is also considered safe driving. If the driver is diabetic then the score is increased by 20% because when individuals with diabetes take insulin or specific pills, their blood sugar can drop too low, affecting their ability to think clearly, crucial for driving. For the Diabetic patient: Session Score = Session Score + (Session Score*0.2) To calculate the final score, the performance is measured across all the sessions completed by the driver. It provides an overall evaluation of the driver's performance. The total score is calculated by the formula:

Total Score = (All Sessions Score)/ (Number of ses- sions)

4. Results

4.1. CNN Model

Custom CNN model is trained on 83,059 image dataset. For testing model 3,223 images were used. The confusion matrix is used for calculating accuracy using 3. Based on accuracy, precision, and recall the performance of a model can be decided.

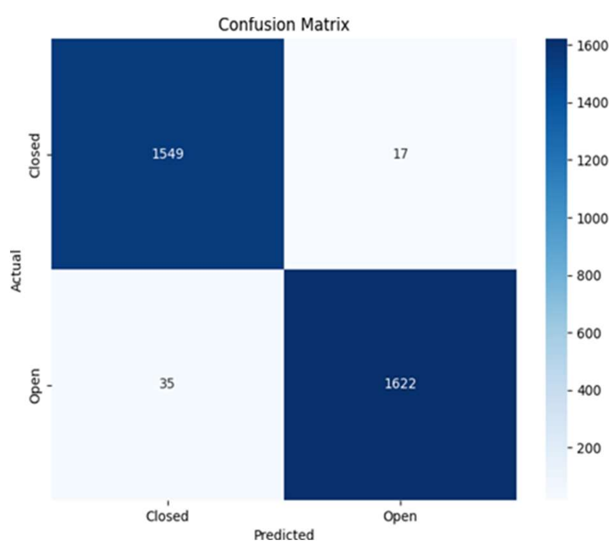

Figure 3: Confusion Matrix

Accuracy = (TP + TN)/(TP + FP + FN + TN)

Accuracy = (1549+1622)/(17+35+1549+1622)

Accuracy = 0.98387

Accuracy % = 0.98387 X 100 Accuracy % = 98.387 %

After testing the custom CNN model accuracy is 98.387% on 3,223 testing images.

Precision = (TP)/(TP + FP) Precision = (1549)/(1549+17) Precision = 0.98914

Recall = (TP)/(TP + FN) Recall = (1549/1549+35)

Recall = 0.97791

F1-score =2 X (precision X recall)/(precison+recall) F1-score = 2(0.97791 X 0.98914)/(0.97791+0.98914)

F1-score = 0.98313

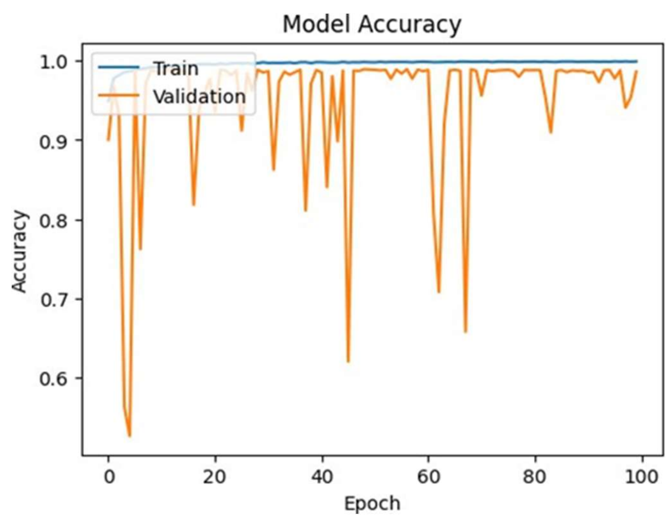

Figure 4: Epoch vs Accuracy

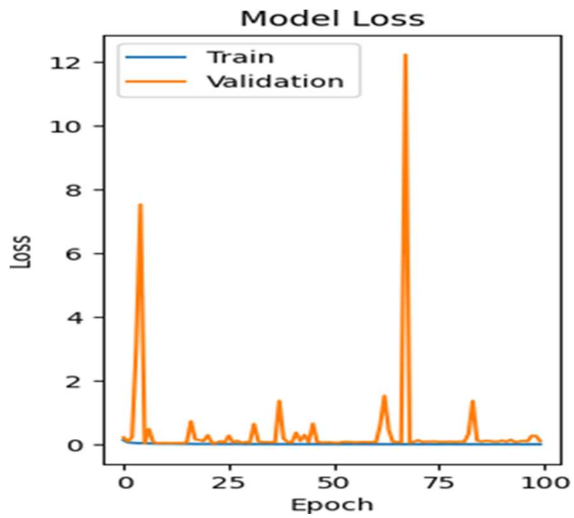

Figure 5: Model Loss vs Epoch

Fig 4 shows the graph of how model learning is done while training with respect to the epoch. The increase in epoch increases the accuracy of the model and reduces the loss. Fig 5 is the loss vs epoch graph this shows the model loss is decreasing as epoch is increasing and at 100 model loss is near to 0 i.e the accuracy of model is high.

4.2. Drowsiness Detection and Score Calculation

Drowsiness is detected from passing each frame into the model. The timer starts after detecting “closed eye” in a frame. The timer will stop after the model detects “open eye” and between this, if it exceeds time 3 sec then this is a drowsy state and an alarm is raised. If the eye is closed after 3 sec then till the eye is open the duration is called drowsiness duration. This drowsiness duration gets recorded into the MYSQL database to calculate the driving score.

Fig 6 shows the drowsiness detected and "Count" is the timer to detect the drowsiness

Fig 7 shows the calculated score of each session and "Average Score" is the final score calculated on all the past session scores.

```
Driver: Shubham
Session ID: 1, Score: 28.47%
Session ID: 3, Score: 28.68%
Session ID: 4, Score: 9.70%
Session ID: 5, Score: 10.19%
Session ID: 6, Score: 3.36%
Session ID: 7, Score: 29.72%
Session ID: 9, Score: 20.11%
Average Score: 18.60%
```

Figure 6: Drowsiness Detection

Figure 7: Score Calculation

5. Discussion

The proposed method of the custom CNN model in this paper has an of accuracy 0.98387, f1 score of 0.98313, recall of 0.97791 and precision is 0.98914 over 3,223 testing images. Considering these values it is the most efficient

Model	Precision	Recall	F1 Score	Accuracy	Training Images	Testing Images
InceptionV3	0.9945	0.9967	0.9956	0.9956	6,800	1,020
VGG16	0.9928	0.9951	0.9934	0.9934	6,800	1,020
ResNet50V2	0.9982	0.9996	0.9989	0.9989	6,800	1,020
EfficientNetB0	Na	Na	Na	0.578	Na	Na
RF	0.9771	Na	0.9582	0.959	14,400	3,600
Proposed CNN Model	0.98914	0.97791	0.98313	0.98387	83,059	3,223

Table 4

Data of other Models

CNN model to detect drowsiness also the model is trained on a large dataset containing 83,059 images. from table 4 all other models like InceptionV3, VGG16, resNet50V2 are trained and tested on less number of images. The accuracy of these models is high because they use only 1,020 images for testing and 6,800 images for training which is very low to use this model in real-world applications. In EfficientNetB0 model the accuracy is too low i.e. 0.578, because of its low accuracy it can not be used in drowsiness detection where accuracy is considered. In the RF model accuracy is 0.959 but again they use less number of images for training. So to make a model feasible for real-world applications model has to be trained on a large dataset that contains eye images in various lightning conditions and that is done by using the MRL(Media Research Lab) eye database. Based on the data from table no. 4, the custom CNN model demonstrates superior accuracy compared to InceptionV3, VGG16, and ResNet50V2. Although these models exhibit slightly higher accuracy, they employ fewer training and testing images to evaluate efficiency. The custom CNN model, on the other hand, uses a large database for training and achieves the highest accuracy across 3,223 images during training.

6. Conclusions and Future Scope

In this paper, with the help of a custom CNN model that can accurately detect drowsiness. If the eye remains closed for more than 3 seconds, then it is considered a drowsy state. If the drowsy state is detected, an alarm can be raised to alert the driver, and the data can be stored in the MYSQL database. Based on this database the score is calculated. The score represents the probability of a driver being drowsy while driving. A high score shows a high probability of drowsiness. This score informs decisions regarding driver safety, prompting actions for drivers having high scores. Future accidents can be minimized by taking action on these drivers. In addition, to calculate a score precisely that describes the driving ability of a driver, yawn detection, lane detection, and speed detection can be added. After this implementation, It can be used in driving tests to calculate the driving score, and based on this score, the issuance of a driving license can be determined. This project is useful for the cab services industry. By integrating this project into the cab, passengers can access real-time information about the safety of their cab and while booking of cab passengers can see the safety level of the booked cab. This initiative not only benefits for passengers but also contributes in raising overall safety standards. It helps in improving safety measures, driving behaviors, following traffic rules, and reducing accidents by adopting this project in private and commercial vehicles. Implementing this project in India not only improves driving performance but also increases safety and from this number of accidents also reduces.

References

[1] Florez, Ruben, et al. "A CNN-Based Approach for Driver Drowsi- ness Detection by Real-Time Eye State Identification." Applied Sciences 13.13 (2023): 7849.

[2] Magán, Elena, et al. "Driver drowsiness detection by applying deep learning techniques to sequences of images." Applied Sciences 12.3 (2022): 1145.

[3] Wang, Hui, Lei Zhang, and Longxu Yao. "Application of genetic algorithm based support vector machine in selection of new EEG rhythms for drowsiness detection." Expert Systems with Applica- tions 171 (2021): 114634.

[4] Alharbey, Riad, et al. "Fatigue state detection for tired persons in presence of driving periods." IEEE Access 10 (2022): 79403-79418.

[5] Scarpelli, Serena, et al. "Age-related effect of sleepiness on driv- ing performance: a systematic-review." Brain sciences 11.8 (2021): 1090.

[6] Falkenstein, Michael, Melanie Karthaus, and Ute Brüne-Cohrs. "Age-related diseases and driving safety." Geriatrics 5.4 (2020): 80.

[7] Ratcliff, Roger, and Yonatan Vanunu. "The effect of aging on decision-making while driving: A diffusion model analysis." Psy- chology and aging 37.4 (2022): 441.

[8] Robertsen, Rolf, et al. "Aging and driving: A comparison of driv- ing performance between older and younger drivers in an on-road driving test." SAGE Open 12.2 (2022): 21582440221096133.

[9] Selvaraju, Ramprasaath R., et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization." Proceedings of the IEEE international conference on computer vision. 2017.

[10] Selvaraju, Ramprasaath R., et al. "Grad-CAM: Why did you say that?." arXiv preprint arXiv:1611.07450 (2016).

[11] Chirra, Venkata Rami Reddy, Srinivasulu Reddy Uyyala, and Venkata Krishna Kishore Kolli. "Deep CNN: A Machine Learning Approach for Driver Drowsiness Detection Based on Eye State." Rev. d'Intelligence Artif. 33.6 (2019): 461-466.

[12] Park, Sanghyuk, et al. "Driver drowsiness detection system based on feature representation learning using various deep networks." Asian Conference on Computer Vision. Cham: Springer International Pub- lishing, 2016.

[13] Ramzan, Muhammad, et al. "A survey on state-of-the-art drowsiness detection techniques." IEEE Access 7 (2019): 61904-61919.

[14] Stancin, Igor, Mario Cifrek, and Alan Jovic. "A review of EEG signal features and their application in driver drowsiness detection systems." Sensors 21.11 (2021): 3786.

[15] Ramzan, Muhammad, et al. "A survey on state-of-the-art drowsiness detection techniques." IEEE Access 7 (2019): 61904-61919.

[16] Ngxande, Mkhuseli, Jules-Raymond Tapamo, and Michael Burke. "Driver drowsiness detection using behavioral measures and ma- chine learning techniques: A review of state-of-art techniques." 2017 pattern recognition Association of South Africa and Robotics and mechatronics (PRASA-RobMech) (2017): 156-161.

[17] Chowdhury, Anuva, et al. "Sensor applications and physiological features in drivers' drowsiness detection: A review." IEEE sensors Journal 18.8 (2018): 3055-3067.

[18] Deng, Wanghua, and Ruoxue Wu. "Real-time driver-drowsiness de- tection system using facial features." Ieee Access 7 (2019): 118727- 118738.

[19] Reddy, Bhargava, et al. "Real-time driver drowsiness detection for embedded system using model compression of deep neural net- works." Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 2017.

[20] Albadawi, Yaman, Maen Takruri, and Mohammed Awad. "A review of recent developments in driver drowsiness detection systems." Sensors 22.5 (2022): 2069.

[21] Ghoddoosian, Reza, Marnim Galib, and Vassilis Athitsos. "A re- alistic dataset and baseline temporal model for early drowsiness detection." Proceedings of the ieee/cvf conference on computer vision and pattern recognition workshops. 2019.

[22] Jabbar, Rateb, et al. "Real-time driver drowsiness detection for an- droid application using deep neural networks techniques." Procedia computer science 130 (2018): 400-407.

[23] Mohanty, Shruti, et al. "Design of real-time drowsiness detec- tion system using dlib." 2019 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE). IEEE, 2019.

[24] McDonald, Anthony D., et al. "A contextual and temporal algorithm for driver drowsiness detection." Accident Analysis Prevention 113 (2018): 25-37.