Enhanced Identification of Driver Fatigue System Using Deep Learning

Ms. Swathika T, Ms. Gokila Devi R , Mrs. V. Bakyalakshmi

II M. Sc Computer Science, Dr. N.G.P Arts and Science College,

Coimbatore-48, Tamil Nadu, India.

Assistant Professor, Department of Computer Science,

Dr. N.G.P Arts and Science College, Coimbatore-48, Tamil Nadu, India.

Abstract

Driver fatigue is a critical factor contributing to road accidents, emphasizing the need for robust fatigue detection systems. This research presents a novel approach to enhance the identification of driver fatigue using deep learning techniques. Leveraging state-of-the-art neural network architectures, our model integrates diverse data sources, including physiological signals and driving behavior patterns. The system's effectiveness is evaluated through comprehensive training and testing procedures, showcasing its potential to outperform existing methods. The study not only addresses the limitations of current fatigue detection systems but also offers insights into the integration of deep learning for advanced real-time monitoring. The findings pave the way for improved road safety measures and open avenues for the research on intelligent transportation systems in the future.

Index Terms: *Deep Learning, Fatigue Detection System, Neural Network Architectures, Road Safety, Accident Prevention.*

1. Introduction

Ensuring safety on the roads is of supreme importance, and one significant challenge we face is addressing driver fatigue- an often under estimated threat to road users. In response to this concern, we embark on a forward-looking initiative: the Enhanced Identification of Driver Fatigue System using Deep Learning. Imagine a sophisticated system that surpasses conventional methods, employing advanced computer techniques (known as neural networks) to analyze signals from the driver's body and observe driving behaviors. This research not only seeks to elevate the standards of road safety but also offers a glimpse into the potential of intelligent transportation.

2. Related Work

The existing system is composed of two sequential subsystems; the input subsystem performs face detection and preprocessing of real-time input video stream data utilizing Haar cascade algorithm, and the output subsystem performs feature extraction and image classification using 5 layers CNN **LeNet** architecture. The evaluation of the model using stratified five-fold cross validation on UTA-RLDD showed that the model achieved high values of average accuracy, precision, recall and F1-score, which are 0.918, 0.928, 0.920, and 0.920, respectively. The system employs deep learning algorithms to analyze the real- time facial expressions of the driver captured through a camera installed in the vehicle. By detecting and interpreting facial cues, such as eye closure, yawning, or facial muscle movements, the system can determine the driver's level of alertness or drowsiness.

3. Proposed System

In the Proposed system, 6 layers of CNN algorithm is used to improve the accuracy and it is non-intrusive. The car has a camera attached to continuously monitor the motions of the driver's eyes. The camera takes pictures of the driver's eyes, and image processing techniques are used to identify variations in the frequency and distance of eye closure. If the system detects that the distance between the driver's eyelids is smaller than a certain threshold for a continuous period of 4 to 5 seconds, it will trigger an alert. This alert can take the form of a beep sound to alert the driver and draw their attention to the fact that they may be drowsy. In addition to the beep sound, the system can also take further action to ensure the driver's safety. By monitoring the driver's eye movements continuously and taking appropriate action when drowsiness is detected, this system can help keep drivers safe and prevent accidents on the road.

4. Software Description

Python IDLE offers a full- fledged file editor, which gives you the ability to write and execute Python programs from within this program. The built-in file editor also includes several features, like code completion and automatic indentation that will speed up your coding workflow. The Packages used are follows as,**CV2**

You can carry out image processing and computer vision activities with the help of the Python package cv2 (opencv-python). It has many features, such as tracking, face recognition, and object detection.

□ **KERAS**

Based on the tensor flow machine learning platform, Keras is a Python deep learning API. It was created with the intention of facilitating quick experimentation and offering a pleasant development experience.

□ **TENSORFLOW**

An end-to-end, open-source machine learning platform is called Tensor flow. With its extensive and adaptable ecosystem of tools, libraries, and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML-powered applications.

□ **NUMPY**

Numpy is a Python library used for working with arrays. Numpy stands for Numerical Python.

□ **MATPLOTLIB**

A complete Python visualization toolkit for static, animated, and interactive graphics is called Matplotlib.

□ **WEB CAMERA VIDEO INPUT**

To detect and recognize eyes in real-time, you will need to have a webcam or video input device connected to your computer.

5. Ethodology

-
- Data Collection
- Building and training a CNN Model
- Face and Eye Detection
- Fatigue Detection
- Alert System

5.1. Data Collection:

An image dataset of the eyes of persons in both open and closed states are used. After constructing our dataset, it must be preprocessed to extract valuable information for deep learning models. Preprocessing data for neural networks primarily aims to accomplish three things: first, it must be cleaned; second, it must handle text and categorical characteristics; and third, it must scale real-valued features using normalization**.**

5.2. Building and training a CNN Model

Once the pre-processing of the datasets is done, CNN need to be built and trained by splitting and balancing the dataset into trained data and test data.

5.3. Face Detection and Eye Detection

Face detection is the process of detecting and locating human faces within an image or video stream. Two popular techniques for face detection are using a CNN model and using a Haar Cascade classifier.

We are detecting eyes in the frame by using the same technique as face detection. Using the closed and open eyes image dataset, we are training the model and detecting the eyes in the detected face using haar cascade eye files.

5.4 Fatigue detection

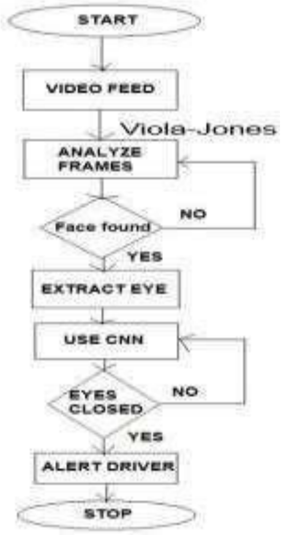
Using a machine learning technique, like a convolutional neural network (CNN), to examine video or picture data of the driver's face and eyes is a popular method for developing a drowsiness detection module. When the algorithm notices that the driver is starting to get sleepy, it can inform them with a sound alarm, vibration, or visual indication. In order to prevent dozing off while driving, the warning may advise the driver to take a rest or change drivers.

5.5. Alert System

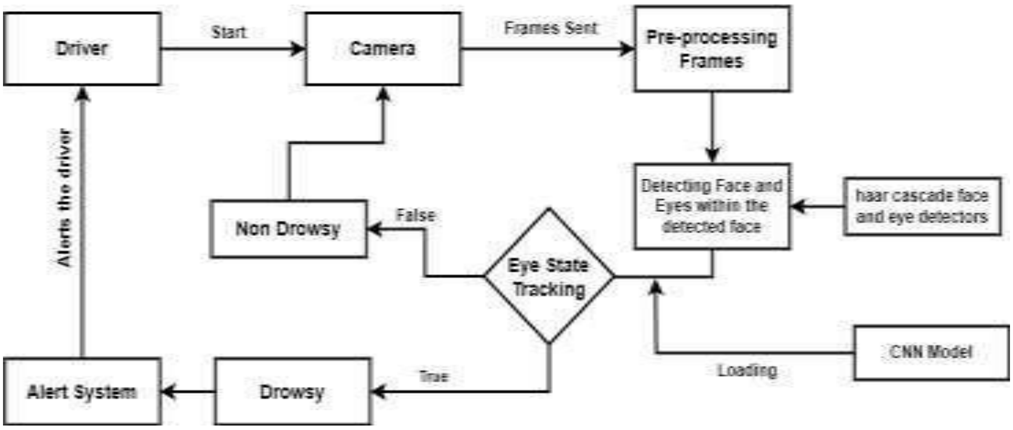
An alert system is a crucial component of a driver fatigue detection, as it can help prevent accidents caused by drowsy driving. When the drowsiness detection module detects that the driver is becoming drowsy or falling asleep at the wheel, the alert system can trigger an alert to the driver, prompting them to take action. The several types of alert systems are audible alarms, voice alerts, Visual cues and so on.

6. Flow Chart

In order to train and validate CNN, the input videos are fed into the code. Once training is finished, the Viola-Jones algorithm is utilized to assess frame-points in the eyes and mouth from a live video feed. Following that, the color images are transformed into grayscale using CNN, and thanks to training, the spots are correctly identified. At last, the appropriate alert messages notify the driver.



i. Viola-jones Algorithm Flow Chart



ii. Proposed CNN Model

7. Viola-Jones algorithm

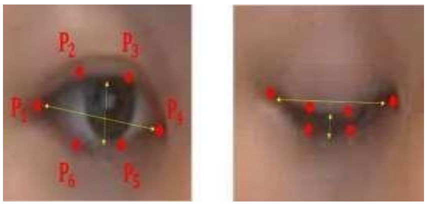
The way that the Viola-Jones algorithm actually works is through the execution of four main steps:

- Haar-like Features
- Integral Image
- Adaboost Training
- Attentional cascade

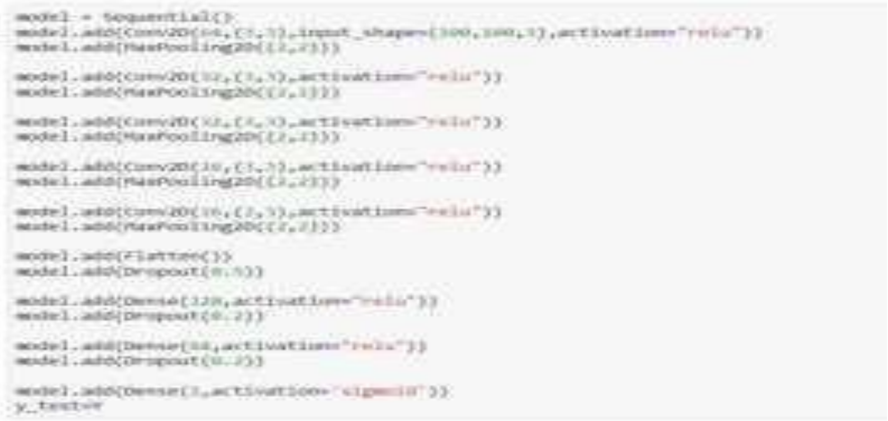
The working of Proposed System Architecture of CNN shown in the figure ii. In this project, in order to reduce the accidents that are caused by driver's drowsiness or sleepiness, a driver fatigue detection system is developed with the help of deep learning techniques. Initially the Dataset contains images of both the closed and open eyes of several persons collected to train the model. Before the training of the model takes place, the images are preprocessed so that the dataset can be directly applied to the deep learning algorithm for further process. After preprocessing, our model has been trained. In particular, activities involving the processing of pixel data and picture identification are performed by CNN, a type of network architecture designed for deep learning algorithms.

The system alerts the driver with a beep sound as its primary response if the driver is drowsy. The Alarm goes on increasing when the driver doesn't wakeup. Then our system is sending mail or making a phone call to the driver's emergency contact person to inform that the person feels drowsy or sleeping while driving along with the driver's coordinates. Thus, this system effectively reduces the risk of road accidents that are caused by driver's carelessness.

8. Results

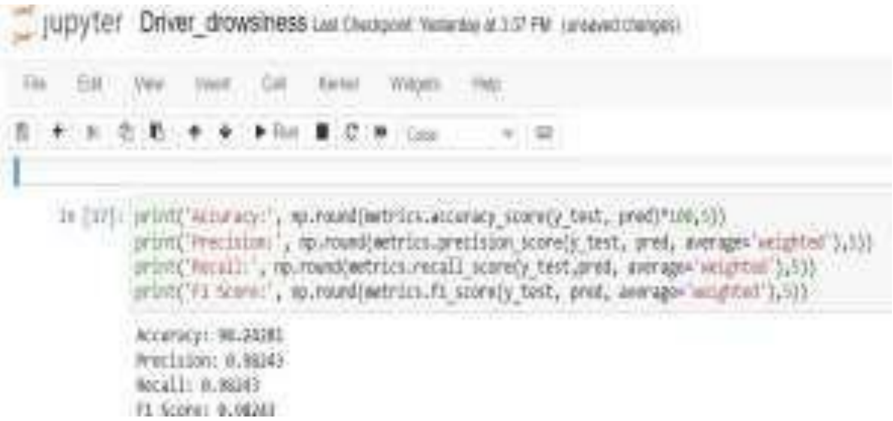


Eye pattern recognition



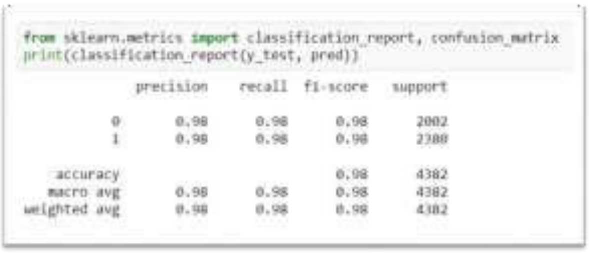
```
model = Sequential()
model.add(Conv2D(64,(3,3),input_shape=(100,100,3),activation="relu"))
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(32,(3,3),activation="relu"))
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(32,(3,3),activation="relu"))
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(16,(3,3),activation="relu"))
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(16,(3,3),activation="relu"))
model.add(MaxPooling2D((2,2)))

model.add(Flatten())
model.add(Dropout(0.5))

model.add(Dense(128,activation="relu"))
model.add(Dropout(0.2))

model.add(Dense(64,activation="relu"))
model.add(Dropout(0.2))

model.add(Dense(1,activation='sigmoid'))
y_test=y
```

Model Structure


jupyter
Driver_drowsiness Last Checkpoint: Yesterday at 3:57 PM (unsaved changes)
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Run
Code

```
In [37]: print('Accuracy:', np.round(metrics.accuracy_score(y_test, pred)*100,5))
         print('Precision: ', np.round(metrics.precision_score(y_test, pred, average='weighted'),5))
         print('Recall: ', np.round(metrics.recall_score(y_test,pred, average='weighted'),5))
         print('F1 Score:', np.round(metrics.f1_score(y_test, pred, average='weighted'),5))

         Accuracy: 98.24381
         Precision: 0.98243
         Recall: 0.98243
         F1 Score: 0.98243
```

Performance analysis



```
from sklearn.metrics import classification_report, confusion_matrix
print(classification_report(y_test, pred))

              precision    recall  f1-score   support

           0       0.98      0.98      0.98      2002
           1       0.98      0.98      0.98      2380

    accuracy                           0.98      4382
   macro avg       0.98      0.98      0.98      4382
weighted avg       0.98      0.98      0.98      4382
```

Classification report

9. Conclusion

The driver drowsiness detection system using Viola jones which requires minimal memory when compared to Convolutional neural network, CNN and OpenCV is an effective solution to improve road safety by alerting drivers who are at risk of falling asleep while driving. The system uses advanced deep learning techniques to accurately detect drowsiness levels in real-time and alert the driver accordingly. By leveraging the power of computer vision and machine learning, we have developed a solution that is more accurate and robust than traditional rule-based systems.

References

[1] Cong Yang, Zhenyu Yang, Weiyu Li, “ FatigueView: A Multi-Camera Video Dataset for Vision-Based Drowsiness Detection ” in IEEE Transactions on Intelligent Transportation Systems, Vol. 24, No. 1, Jan

2023, pp 233 – 246, doi = 10.1109/TITS.2022.3216017.

[2] Lew, M.; Sebe, N.; Huang, T.; Bakker, E.; Vural, E.; Cetin, M.; Ercil, A.; Littlewort, G.; Bartlett, M.; Movellan, J. Drowsy driver detection through facial movement analysis.In Human-Computer Interaction; Springer: Berlin, Germany, 2007; Volume 4796, pp. 6–18.

[3] Mohamed Hedi Baccour, Franke Driewer, Tim Schack and Enkelejda Kasneci, "Comparative Analysis of vechicle and driverbased features for driver drowsiness Monitoring by Support Vector Machines" in IEEE Transactions on Intelligent Transport Systems.

[4] Phillip Ian Wilsom, Dr John Fernandez, Facial feature detection using Haar classifiers. Journal of Computing Sciences in Colleges, vol.

[5] 21, researchgate, 2006. Pushkal Pandey, Monil Sharma, Prakhar Saxena and Rajendra Kumar Dwivedi, " Driver Drowsiness Monitoring and Detection using Machine Learning" in IDCIoT 2023, pp 421 - 426, doi=10.1109/IDCIoT56793.2023.10053497.

[6] Yin, B.-C.; Fan, X.; Sun, Y.-F. Multiscale dynamic features based driver fatigue detection.Int. J. Pattern Recogn. Artif. Intell. 2009, 23, 575–589.

[7] Duy-Linh Nguyen, Muhamad Dwisnanto Putro and Kang-Hyun Jo, "Driver Behaviours Recognizer Based on Light-Weight Convolutional Neural Network Architecture and Attention Mechanism " in IEEE Access, July 11 2022,pp71019,doi= 10.1109/ACCESS.2022.3187185.