Volume 6 –Issue 2,August 2023
Paper:51

Driver's Drowsiness Detection In Automobiles Using Deep Learning

Avinash.V
Department of Computer Science and Engineering
Vel Tech Multi Tech Dr.Rangarajan and Dr.Sakunthala Engineering College
Chennai, India .
selva2001mani@gmail.com

Edwin Kingsten.A
Department of Computer Science and Engineering
Vel Tech Multi Tech Dr.Rangarajan and Dr.Sakunthala Engineering College
Chennai, India .
edwinkingsten@gmail.com

Gokul.D.G
Department of Computer Science and Engineering
Vel Tech Multi Tech Dr.Rangarajan and Dr.Sakunthala Engineering College
Chennai, India .
dgkgokullatha@gmail.com

Manikiandan.P
Department of Computer Science and Engineering
Vel Tech Multi Tech Dr.Rangarajan and Dr.Sakunthala Engineering College
Chennai, India .
itsmemanimass481@gmail.com

Abstract—Traffic accidents caused by tiredness, exhaustion, and distracted driving are a big concern on a global scale. In this study, we suggest a collective deep learning architecture for automatic driver sleepiness detection. While some earlier works suggested extracting the lips and eyes movements to identify sleepiness, more contemporary computer vision-based systems have performed only somewhat well. This is because they either use extremely large deep learning models with still-poor performance, or they use hand-crafted features with traditional methods like Naive Bayes and SVM. Our suggested ensemble deep learning architecture evaluates the driver's fitness using integrated features from the mouth, eyes, and body-important subsamples in conjunction with a decision structure. Our suggested architecture delivers great accuracy in detecting driver drowsiness by leveraging the strengths of many deep learning models. We conduct comprehensive experiments on real-world datasets to show the efficacy of our methodology. The number of accidents brought on by drowsy driving might be drastically decreased using our suggested approach, which would also increase traffic safety.

Keywords—Driver drowsiness detection, convolutional neural network, ensemble network, multitask cascaded convolutional networks (MTCNN)

INTRODUCTION

Drowsiness is identified when a person is tired enough to be transitioning between 'asleep' and 'awake' cognitive states owing to a variety of spontaneous and long-term health or mental conditions. A driver may be well-aware of rules and safety measures, but it is the eventual fatigue that triggers laxity in the nervous system of the body leading to crashes and mishaps. Some of the known underlying causes [1] of a distracted driver behind the wheels are lifestyle factors, sleep disorders among other mental and health conditions. Substantial evidence [1], [2] pinpoints drowsiness to be a causing factor for road accidents. A survey in the US by the National Highway Traffic Safety Association (2019) [2] reported that road accidents due to driver drowsiness, from the year 2013 to 2017, claimed about 4,111 lives, and about 91000 car crashes and 795 deaths (in the year 2017).It is evident from a CDC report (2014) [3] that the curve depicting fatalities has been on the rise over the past decade. Studies also reveal that a significant fraction of these crashes can be prevented [1], [3] by appropriate measures including, but not limited to, measurement of the problem, public awareness, policy development, and improving vehicular technology. +There are many time-series-based anatomical and physiological features that could be incorporated in drowsiness detection and alarm systems. It is, but, essential to employ methods for feature extraction that are practically possible, fast to process, and are reliable. These EEG and ECG-based features tend to work well as they accurately capture the physiological patterns in the body but these are not an ideal choice due to their partial invasive nature and difficulty in deployment. Many leading automobile industries incorporate vehicle-based sensors for detecting driving patterns in the vehicles but these vehicles were targeted for higher prices in the market. In this scenario, research is essential in cost efficient methods that use non-invasive sensors and can be used by a large-scale population.

LITERATURE SURVEY

[1].Muneeb Ahmed, Sarfaraz Masood, Musheer Ahmad and Ahmed A. An article titled "Intelligent Driver Drowsiness Detection for Traffic Safety Based on Multi CNN Deep Model and Facial Subsampling" was published in October 2022 by Abd El-Latif.The proposed ensemble model consists of only two InceptionV3 modules that help in containing the parameter space of the network. These two modules respectively and exclusively perform feature extraction of eyes and mouth subsamples extracted using the MTCNN from the face images. Their respective output is passed to the ensemble boundary using the weighted average method whose weights are tuned using the ensemble algorithm.

[2].Islam A. Fouad's "A robust and efficient EEG-based drowsiness detection system using different machine learning algorithms" Published July 2022.The proposed SS-CNN provides an accuracy of 98.95%, which outperforms the previous works. In the experimental section, we provide several links in which real-time operations of the proposed system can be observed.

[3].Md Mahmudul Hasanab, Christopher N.Watlingab and Grégoire S.Larue's "Physiological signal-based drowsiness detection using machine learning: Singular and hybrid signal approaches " Published December 2021
This approach considered a range of metrics from three physiological signals – electroencephalography(EEG), electrooculography (EOG), and electrocardiography (ECG) – and used subjective sleepiness indices (assessed via the Karolinska Sleepiness Scale) as ground truth.

PROPOSED SYSTEM METHODOLOGY

PROPOSED SYSTEM ARCHITECTURE

This section describes the proposed ensemble architecture for drowsiness detection systems designed to work in different real-world driving environment scenarios. The developed model has three phases: face region extraction/segmentation, training the CNNs and training the ensemble network. The overview of the proposed framework is shown in Fig. 1.

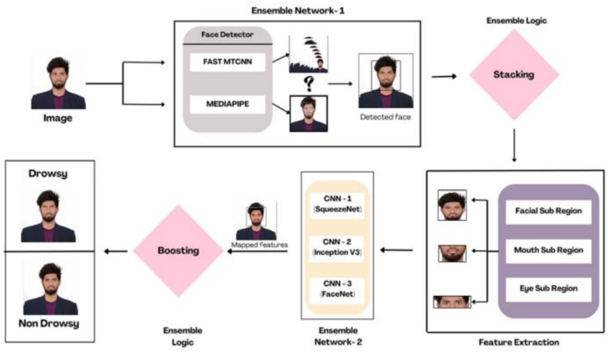

Fig. 1 **Technical Architecture**

HARDWARE REQUIREMENTS

- Processor
- Video card
- Memory
- Webcam

A. Processor

A processor, also known as a central processing unit (CPU), is the component in a computer that carries out instructions and calculations. It is the "brain" of the computer and operates in cycles to retrieve, decode, execute, and store data and instructions. Its speed is measured in hertz (Hz) and it can have multiple cores to perform tasks simultaneously. The processor is responsible for controlling the flow of data between different parts of the computer system and is essential for running software and performing computing tasks.

B.Video Card

A video card, also known as a graphics card or GPU (graphics processing unit), is a component in a computer system that is responsible for rendering and displaying images and video on a monitor or display. It is designed to perform complex calculations and transform data to produce high-quality graphics for gaming, video editing, and other graphics-intensive applications.

The video card has its own dedicated memory and processing power, which offloads graphics processing from the CPU and enhances overall system performance. The performance of a video card is measured in terms of its processing power, memory, and clock speed. It connects to the computer through a PCIe slot or other interface and can support multiple monitors for extended or mirrored displays.

C. Memory

Memory, also known as RAM (Random Access Memory), is a component in a computer system that stores data and instructions temporarily while the computer is running. It provides quick access to frequently used data and instructions for the CPU, which enhances overall system performance. Memory operates at high speeds, allowing for fast read and write operations, and comes in different sizes and speeds. It is typically measured in gigabytes (GB) and its speed is measured in megahertz (MHz) or gigahertz (GHz). The amount and speed of memory in a computer can significantly impact its performance, particularly when running multiple programs or tasks simultaneously.

D. Webcam

A webcam, short for web camera, is a digital camera that captures video and audio and transmits it over the Internet in real-time. Webcams are commonly used for video conferencing, online streaming, and remote monitoring. They are typically connected to a computer via USB or another interface and can be mounted on a monitor or placed on a desk. Webcams can vary in quality and features, but most modern webcams offer high-definition video and built-in microphones for clear audio.

SOFTWARE REQUIREMENTS

- Operating System
- Python
- Tensorflow
- Google colab

A. Operating System

An operating system (OS) is a software program that manages computer hardware and software resources and provides common services for computer programs. The operating system acts as a mediator between computer applications and hardware, controlling and coordinating the use of system resources such as memory, processing power, storage, and input/output devices. Some common operating systems include Windows, macOS, Linux, Android, and iOS. An OS provides a user interface, through which users can interact with the computer and its applications, as well as a set of system utilities that perform tasks such as file management, system maintenance, and security.

B. Python

Python is a runtime environment object-oriented high-level computer program with dynamic semantics. Since there is no compilation phase, the modify cycle is extremely fast. By setting a source level breakpoint, users can analyze both local and global variables, execute custom expressions, define new breakpoints, and navigate through the code step-by-step.Python provides modules and packages, which assists with programming modularity and code reuse.

C . Tensorflow

TensorFlow is an open-source machine learning library developed by Google that allows developers to build and train powerful deep learning models. It provides a range of tools, libraries, and resources for creating and deploying machine learning applications across a variety of platforms, including desktops, mobile devices, and the cloud. TensorFlow uses data flow graphs to represent computation and enables users to define, optimize, and execute complex mathematical computations with ease. With TensorFlow, developers can create a wide range of deep learning models, including convolutional neural networks, recurrent neural networks, and more, for tasks such as image and speech recognition, natural language processing, and time series analysis. TensorFlow also includes a number of high-level APIs that make it easier for developers to get started with machine learning and to build and train models quickly and efficiently.

D. Google Colab

Google Colab is a free cloud-based Jupyter notebook environment provided by Google. It allows users to write and execute Python code through their browser, with access to powerful hardware resources such as GPUs and TPUs. Colab notebooks can be used for various purposes such as machine learning, data analysis, and visualization. Users can easily import and export data, collaborate with others, and even access popular machine learning frameworks such as TensorFlow, Keras, and PyTorch. Colab notebooks are stored in the user's Google Drive and can be shared with others, making it an ideal tool for collaborative projects. Additionally, Colab also provides a range of pre-installed libraries and tools to make it easy for users to get started with their projects.

.

RESULT AND ANALYSIS

By utilizing the NTHU-DDD dataset, which includes video frames of drivers exhibiting both sleepy and non-drowsy states, a deep neural network classifier is trained using the Adam optimizer. The dataset is divided randomly into 80% for training and 20% for testing. After training, the classifier achieves a 95% accuracy on the test set.

A comparative analysis of the results from recent literature and the proposed model is presented in Table I. It is evident from the results presented in this table that the model has significantly outperformed all the models proposed in the recent works that also used the same NTHU dataset for their training and evaluation. This can be easily attributed to the fact the proposed model used an ensemble of two CNNs that individually focused on Eye and Mouth features only, instead of overloading a single CNN to work on the entire face and extract the feature for classification. This helped the model to concentrate on the extraction of fine-grained feature vectors for each of the facial sub-portions that greatly assisted the classifier section of the model to yield such high values of performance metrics and low values of error rate.

Authors	Technique/Type of model(s) applied	Trainable Parameters	Accuracy
Dua et. al [10]	Ensemble of 4 models (AlexNet, ResNet, FlowImageNet and the VGG-FaceNet)	300,527,792	0.85
Huynh et. al [11]	Semi-supervised learning, CNN	32,386,434	0.874
Park et. al [20]	Ensemble of 3 models (VGG-FaceNet, FlowImageNet & AlexNet)	60M+60M+60M = ~180 M	0.731
Yu et al [22]	3D-CNN with Condition Adaptive Representation	8,855,936	0.762
Bakheet and Hamadi [23]	Binary HOG features with Naïve Bayes Classifier	-	0.856
Moujahid et al. [24]	PML-based descriptors with Decision blending	-	0.798
Jamshidi et al. [25]	Hierarchical Deep Drowsiness Detection (HDDD) network	95,902,400	0.871
Proposed Model	**MTCNN with Ensemble of 2 InceptionV3 models**	**47,513,508**	**0.971**

Table.I : COMPARATIVE RESULTS WITH RECENT STUDIES OVER THE SELECTED DATASET

Another important observation that can be drawn from Table I is that the total trainable parameters of the proposed model are also on the lower side, as compared to most of the recent deep learning-based notable works on this problem. The model has around 47.5 million trainable parameters while some of the other proposed models are gauged in a range of 95 million to as high as 300 million. Though the model proposed by Yu *et al.* [22] has around 8.9 million trainable parameters only but has a highly limited performance with an accuracy of only 0.762.

The models suggested in references [23] and [24] were not included in the table since they were not deep learning-based and instead relied on handcrafted features and traditional machine learning algorithms such as Naïve Bayes. Moreover, the performance of the proposed architecture in real-time is further analyzed in Table I, which breaks down the different stages involved.

In this table, the time required by end-to-end prediction (in seconds) has been reported for the proposed architecture.
The processing time utilized by MTCNN is inevitable. However, two strategies can be used to reduce the overall time of prediction: 1) fetching input apriority, and 2) running classification processes concurrently.

CONCLUSION

In this study, a stacking-based ensemble architecture has been proposed that combines the weighted contribution oftwo Inception V3 CNNs which learns eyes and mouth features separately in the samples taken from the NTHU-DDD benchmark dataset. Unlike past studies, this work is plausible as it uses a simple feed from a dashboard camera for drowsiness detection, instead of monitoring the driver's EEG, ECG, or any other physiological signals. In the proposed model, the MTCNN has been employed to extract coordinates of the regions of interest (eyes and mouth) apriori to the processing by the two CNNs. The weights of each ensemble are trained on a perceptron network that employs outputs of the two CNNs as its input. We have demonstrated higher performance using the ensemble architecture against the performance of individual CNNs working on eye and mouth subsamples. This end-to-end model is also robust to variations of pose and illumination conditions as results showed a higher rate of extraction of frames after using MTCNN. The results revealed significantly high detection ability from the proposed model over a large evaluation dataset which also outperforms various recently proposed non-invasive solutions on this dataset.

REFERENCES

[1] Centers for Disease Control and Prevention. (Mar. 21, 2017). *CDC— Drowsy Driving-Sleep and Sleep Disorders*. Accessed: Jun. 28, 2021. [Online]. Available: https://www.cdc.gov/sleep/about_sleep/drowsy_ driving.html

[2] U. Budak, V. Bajaj, Y. Akbulut, O. Atila, and A. Sengur, "An effective hybrid model for EEG-based drowsiness detection," *IEEE Sensors J.*, vol. 19, no. 17, pp. 7624–7631, Sep. 2019.

[3] W. Tipprasert, T. Charoenpong, C. Chianrabutra, and C. Sukjamsri, "This study proposes an approach for detecting driver drowsiness by analyzing eye closure and yawning through the use of an infrared camera." in *Proc. 1st Int. Symp. Instrum., Control, Artif. Intell., Robot. (ICA-SYMP)*, Bangkok, Thailand, Jan. 2019, pp. 61–64.

[4] K. Dwivedi, K. Biswaranjan, and A. Sethi, "Drowsy driver detection using representation learning," in *Proc. IEEE Int. Adv. Comput. Conf. (IACC)*, Gurgaon, India, Feb. 2014, pp. 995–999.

[5] G. Li and W.-Y. Chung, "Detection of driver drowsiness using wavelet analysis of heart rate variability and a support vector machine classifier," *Sensors*, vol. 13, no. 12, pp. 16494–16511, Dec. 2013.

[6] B.-G. Lee and W.-Y. Chung, "A smartphone-based driver safety monitoring system using data fusion," *Sensors*, vol. 12, no. 12, pp. 17536–17552, Dec. 2012.

[7] G. J. Al-Anizy, M. J. Nordin, and M. M. Razooq, "Automatic driver drowsiness detection using Haar algorithm and support vector machine techniques," *Asian J. Appl. Sci.*, vol. 8, no. 2, pp. 149–157, 2015.

[8] M. Awais, N. Badruddin, and M. Drieberg, "A hybrid approach to detect driver drowsiness utilizing physiological signals to improve system performance and wearability," *Sensors*, vol. 17, no. 9, p. 1991, 2017.

[9] P. Cummings, T. D. Koepsell, J. M. Moffat, and F. P. Rivara, "Drowsiness, counter-measures to drowsiness, and the risk of a motor vehicle crash," *Injury Prevention*, vol. 7, no. 3, pp. 194–199, 2001.

[10] M. Dua, R. Singla, S. Raj, and A. Jangra, "Deep CNN models-based ensemble approach to driver drowsiness detection," *Neural Comput. Appl.*, vol. 33, pp. 3155–3168, Jul. 2020.

[11] X.-P. Huynh, S.-M. Park, and Y.-G. Kim, "Detection of driver drowsiness using 3D deep neural network and semi-supervised gradient boosting machine," in *Proc. Comput. Vis.-ACCV Workshops*, in Lecture Notes in Computer Science, 2017, pp. 134–145.

[12] Sleep Education. (Apr. 29, 2021). *Prevent Drowsy SDriving: Stay Awake at the Wheel!* Accessed: Jun. 28, 2021. [Online]. Available: http://sleepeducation.org/sleep-topics/drowsy-driving.