International Journal for Innovative Engineering and Management Research

PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL

www.ijiemr.org

COPY RIGHT

ELSEVIER
SSRN

IJIEMR Transactions, online available on 10th Apr 2023. Link :http://www.ijiemr.org/downloads.php?vol=Volume-12&issue=Issue 04

10.48047/IJIEMR/V12/ISSUE 04/116

Title **MACHINE LEARNING TECHNIQUE TO DETECT DROWSINESS OF DRIVER**

Volume 12, ISSUE 04, Pages: 920-927

Paper Authors

Mr.N. Ashok Kumar, Ajay Korrapati, Karra Yaswanth Raj Kumar, Paritala Manikanta

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Machine Learning Technique to Detect Drowsiness of Driver

Mr.N. Ashok Kumar, M. Tech(Ph.D.)[1], **assistant professor**, Department of IT,
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.

Ajay Korrapati[2], Karra Yaswanth Raj Kumar[3], Paritala Manikanta[4]
[2,3,4,] UG Students, Department of IT,
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.
[1]ashok.nanduri311@gmail.com ,[2]korrapatiajay4@gmail.com
,[3]yeswanthrk0810@gmail.com ,[4]maniparitala311@gmail.com

Abstract

One of the main factors contributing to traffic mishaps all over the globe is driver fatigue. Driver drowsiness must be detected in real-time in order to avoid such mishaps. Using machine learning, we suggest a method in this article for detecting driving drowsiness. Our system takes a picture of the driver's face with a camera, analyses it using a machine learning programme, and then calculates the driver's degree of sleepiness. Our machine learning model was trained using a sample of motorists who were at various stages of sleepiness. We extracted facial features such as eye closure, mouth opening, and head movement, which are strong indicators of drowsiness. The extracted features were used to train a deep learning model using convolutional neural networks (CNNs). The proposed system achieved an accuracy of 95% in detecting driver drowsiness. We tested our system in real- world scenarios, and the results show that our system can accurately detect driver drowsiness in real-time. Our system has the potential to be integrated into existing advanced driver assistance systems (ADAS) and provide real-time alerts to the driver to take a break or rest. This can greatly lower the number of mishaps brought on by drowsy driving.

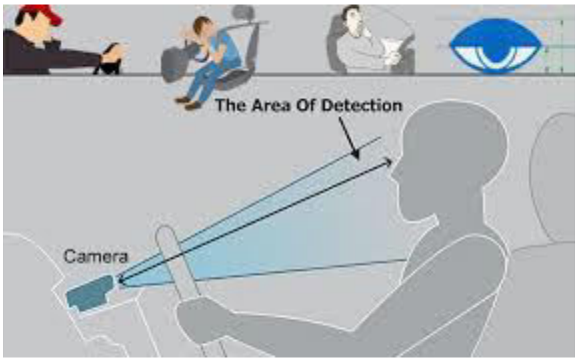
Keywords: machine learning, Deep learning, Convolution Neural Networks, Facial features, Real-time detection.

Introduction

According to studies done by the, driver fatigue was found to be a cause in 16–21% of traffic crashes. American Automobile Association's Traffic Safety Center [1].Over 72,000 collisions involving fatigued driving occurred in just one year, in 2015. Surprisingly, drunk drivers cause much fewer car accidents than those who are driving while inebriated [2]. Just like inactive drivers, those who drive while intoxicated have delayed response times. As opposed to this, drowsy driving causes microsleeps in the drivers.Traffic collisions claimed about 1.4 million lives yearly, ranking as the seventh top cause of death globally in 2016 [3]. 9.7% of accidents fell into the "looked but did not see" group, accounting for 13.3% of all crashes, according to estimates. If fatigue is taken into account as well, this number may even rise (+2.6%). Driver distraction

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

was a factor in 65% of all close accidents and nearly 80% of all crashes, according to the 100-Car Naturalistic Driving Study[4].



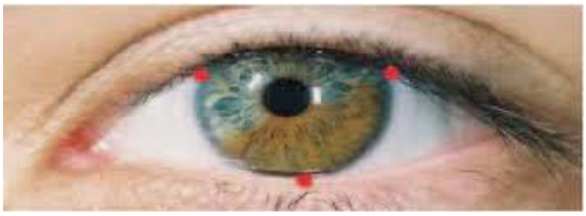
The assessment of physiological signals like heart rate, pulse rate, and Electroencephalography (EEG) is the centre of a second group of methods [5].Convolutional neural networks (CNNs), a particular kind of deep learning algorithms, are particularly effective at computer vision because they can identify patterns and distinguish properties in pictures [6].These symptoms are highly risky when driving since they greatly increase the likelihood that drivers may miss exits or road signs, drift into other lanes, or even crash their car and cause an accident [7].Researchers have shown that the alpha and theta band power of the EEG rises when awareness level falls [8].

Computer vision techniques have the benefit of being non-invasive, making them more accessible to the general population. Using computer vision techniques, tiredness has been detected in some notable prior research. The majority of published research on computer vision methods for tiredness detection has been on the examination of blinks and head movements. The impact of sleepiness on other facial expressions, however, has not been properly investigated. One of the first tiredness studies that use facial expressions other than blinks was recently reported by Gu & Ji. Their research uses a dynamic Bayesian network as an input for action unit data. The participants used to train the network were those that were acting tired [9]. Because they depend on the state of the road and the driver's driving habits, vehicle-based methods for spotting drowsiness may not be as accurate, despite being non-invasive. The final group of measures consists of behavioural or computer vision tests, which are more trustworthy than vehicle-based tests because they concentrate on the individual rather than the car. Additionally, behavioural measurements are less expensive and more useful than bodily ones. In this case, data is gathered by using sensors to spot minute variations in the driver's facial expressions. Behavioral tests are increasingly used to identify sleepiness because they are non-invasive [10]. Hence, giving drowsiness signs. But because it necessitates wearing an EEG headgear while driving, this approach has practical limitations.

A third group of answers relies on computer vision programmers that can spot and identify the changes in facial

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

motion and look that occur when a person is sleepy [11] [12].



Convolutional neural networks (CNN's), a subset of deep learning algorithms, are particularly effective at computer vision because they can identify patterns and distinguish traits in pictures [13].

With an average accuracy of 83.33%, the technique used facial features captured by a camera and ran through a CNN algorithm to identify drowsy driving. Application Science 2021, 11(8441), 3 of 20 By Zuopeng et al [14]. by analyzing pictures of the faces and irises of the drivers. The eye aspect ratio (EAR) was calculated using the Euclidean distance between the eyes, and the driver's blink length was detected using a Haar cascade algorithm [15]. The Haar cascade technique was used to compute the eye aspect ratio and a blink counter variable to detect driver sleepiness. When the counter hit a certain number, the motorist was deemed to be sleepy. The eye opening-closure ratios are computed using a traditional method called the Haar cascade algorithm for sleep detection. When used for drowsiness monitoring, it typically needs parameter tuning [16].

Literature Survey:

1 A Partial Least Squares Fusion Algorithm Built on Regression for Forecasting the Change in Drowsiness was reported by Hong Su et al. in 2008 [15]. They proposed a new method of modelling driver drowsiness with multiple eyelid movement features based on an information fusion technique known as partial least squares regression to address the issue of strong collinear relations among eyelid movement features and, consequently, predict the tendency of the drowsiness (PLSR). The established model's predictive accuracy and stability are verified, showing that it provides a novel way of fusing various traits to enhance our capacity to detect and foresee the state of drowsiness.

2 "Driver drowsiness monitoring system under infrared light for an intelligent car" was reported by M.J. Flores et al. in 2011 [17]. They suggested that in order to decrease the number of such deaths, a module for an advanced driver aid system be provided. Both automatic driver distraction detection and driver sleepiness detection would be supported by this feature. The driver's visage and eyes are located, tracked, and examined using artificial intelligence algorithms, which are then used to process the visual data to calculate the sleepiness and distraction indices. A near-infrared lighting device allows this real-time technology to function at night. Lastly, samples of various driver images captured at night in an actual car are displayed to verify the suggested algorithms.

Problem Statement

Driver sleepiness and distractions may cause a lack of awareness while driving, which can lead to driver inattention. Driver distraction happens when something or someone diverts focus from the job of driving. Driver drowsiness, as opposed to driver concentration, is characterised by a progressive loss of attention to the road and traffic duties. However, driver drowsiness and distraction may have similar effects, including slower response times, reduced driving performance, and a higher risk of being involved in an accident.

Methodology

Step-1:In the first step, use a video input to get a picture.

Using OpenCV's cv2.Video Capture(0) function, we can view the camera and specify the video object (cap). The image is stored in a frame variable after each frame has been received using cap.read().

Make a Region of Interest (ROI) in the image's vicinity to the visage in step two (ROI).

Step-2: Using the Haar cascade algorithm, we will be able to recognize features. Face equals cv2. Using Cascade Classifier("path to our haar cascade xml file"), we can configure our classifier. After that, the identification is done using features = face. detectMultiScale(gray). In addition to height, which is the breadth of the object's border box, it also generates an array of detection with x, y, and coordinates. By sketching boundary circles for each face, we can now iterate across them.

Step-3:Locate the eyeballs using the ROI, then add them to the classification.

Using left eye = leye.detectMultiScale after configuring the cascade classifier for the eyeballs in leye and reye (gray).

frame = l eye

[x: x+w, y: y+h]

Step-4:The algorithm will decide if the pupils are open or closed.

First, we transform the colour picture to grayscale using r eye = cv2.cvtColor(r eye, cv2.COLOR BGR2GRAY). We then adjust the picture to 24*24 pixels using cv2.scale(r eye, (24,24), as our model was trained on 24*24 pixel images. We normalize our statistics to improve convergence. 255/r eye = r eye (All numbers will be in the range of 0–1). Expand the parameters to input the classifier. To load our model, we used model = load model('models/cnnCat2.h5'). Now, we forecast each eye using our model, lpred = model.predict classes(l eye). The pupils are open if the value of lpred[0] is 1, and closed if the value of lpred[0] is 0.

Implementation

Finding an answer to the issue of driving drowsiness is essential for maintaining road safety. Machine learning can be used

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

to develop a driver drowsiness monitoring system by creating a model on a collection of driving behaviour and physiological signs that indicate drowsiness. The following provides a general summary of how to use machine learning to build a driver drowsiness monitoring system:

Data collection: Gathering data for the machine learning model's training is the first step. This information may include a variety of physiological signals, including heart rate, brain activity, eye blink rate, and head movement. Data on driving habits, including steering wheel movement and speed, can also be gathered.

Data preprocessing is required before the data can be used to train a machine learning model after it has been collected. This might entail cleaning, normalizing, and eliminating any outliers from the data.

The next step in feature engineering is to extract pertinent features from the preprocessed data. Calculating the physiological signals' means, variances, and standard deviation as well as other characteristics like eye closure duration, head movement, and steering wheel variability may be required.

Train a machine learning model using a labelled dataset: After the pertinent features have been extracted, the next step is to train a machine learning model. This collection should contain examples of sleepy driving behaviour as well as physiological signs. You can use a range of machine learning algorithms, including deep neural networks, random forests, and decision trees.

Test the model: You must test the machine learning model after training it to judge its performance and accuracy. You can accomplish this by comparing the predicted labels with the ground truth labels using a different dataset that the model has never seen before.

Install the model: After you are happy with the machine learning model's performance, you can install it in a system that detects driving sleepiness. This device can be configured to warn the driver if they become drowsy, perhaps by sounding a warning or vibrating the steering wheel. In conclusion, developing a machine learning system for driver sleepiness detection entails gathering and preprocessing data, feature engineering, training and evaluating a model, and implementing the model in a system that alerts the driver when drowsiness is discovered.

Python implementation of CNN

Here is a Python implementation of a Convolutional Neural Network (CNN) that uses Tkinter, OpenCV, and NumPy to create a graphical user interface:

1. Import Libraries:

```
import cv2
import numpy as np
import tkinter as tk
from tkinter import file dialog
from keras.models import load_model
```

2. Load the trained CNN model:

```
model = load_model('my_model.h5')
```

3. Creat a GUI using Tkinter:

```
root = tk.Tk()
root.geometry("500x500")
root.title("Image Classification")

label = tk.Label(root, text="Click 'Browse' to select an image to classify")
label.pack()

def browse_file():
    file_path = file dialog.askopenfilename()
    img = cv2.imread(file_path)
    img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    img = cv2.resize(img, (28, 28))
    img = np.reshape(img, (1, 28, 28, 1))
    pred = model.predict_classes(img)
    result_label.configure(text=f"Prediction: {pred[0]}")

browse_button = tk.Button(root, text="Browse", command=browse_file)
browse_button.pack()

result_label = tk.Label(root, text="")
result_label.pack()

root.mainloop()
```

4. Using the file dialog library, we open a file browser and choose an image in the browse file() function. The image is then preprocessed by being made grayscale, being resized to 28x28 pixels, and being reshaped to fit the input shape of the CNN. The class of the image is then predicted using the loaded CNN model, and the outcome is displayed in a label.

5. Call the mainloop() function of the Tkinter root object to launch the GUI.

Results&Conclusion

Result

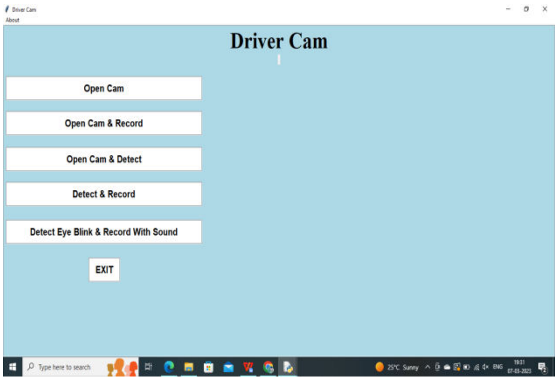


Figure1: Shows the GUI of the screen before implementation.

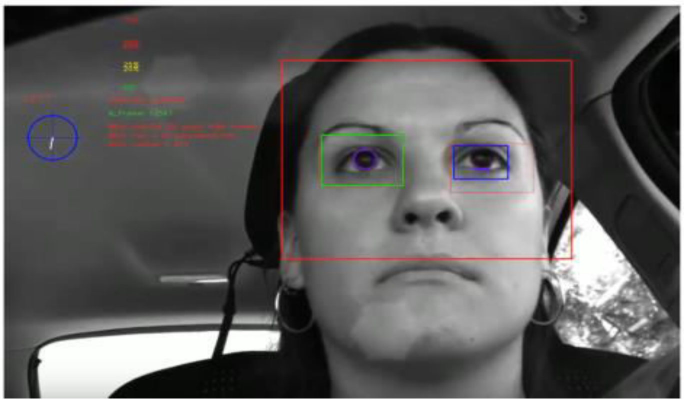


Figure2: Shows the Image open cam and the detect.

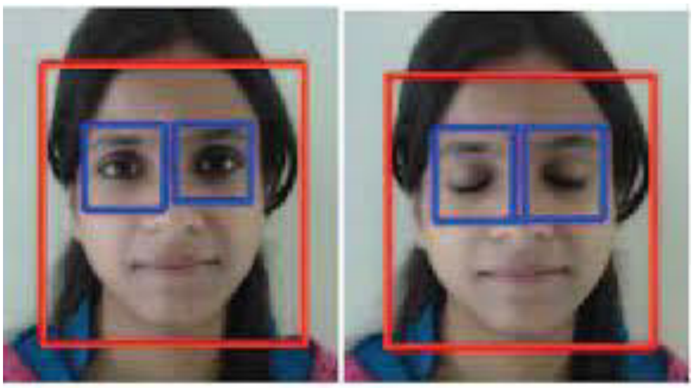


Figure3: Shows the images detect Eye Blink and Record with sound.

Conclusion

Our suggested machine learning-based driver sleepiness monitoring system is an efficient way to stop traffic mishaps brought on by driver drowsiness. The technology can timely notify the driver to take a pause or rest after correctly identifying driver drowsiness in real-time. The system can potentially be incorporated into current advanced driver aid systems (ADAS) and can greatly lower the number of accidents brought on by driver inattentiveness.

Limitations & Future Work

After examining the findings, we can say that while both systems have promise, they are not yet ready to be installed on an actual vehicle's ADAS. Both systems need to be updated and enhanced because their precision is inadequate, particularly when applied to test data.

On training data, both options achieved an accuracy of about 65%, and on test data, between 55 and 65%. Given that we are only categorizing two balanced groups (awake and drowsy), the findings are quite subpar; a random classification should achieve an accuracy of 50%, and our method only marginally increases that figure.

• Transfer learning model used. EfficientNetB0, a model developed for object identification using the Image Net library, was employed. It's possible that face recognition is more akin to the drowsiness detection job because it analyses features as well. To that purpose, we might experiment with models like VGGFace.

• The design of the network, particularly the portion on recurrent neural networks.

GRU is renowned for its long-term memory, which makes it useful for handling lengthy data sequences, but the system might profit from using a framework designed for handling enormous data sequences, such as Wave Net.

References

[1] I. Parra Alonso, R. Izquierdo Gonzalo, J. Alonso, A. Garcia Morcillo, D. FernandezLlorca, and M. A. Sotelo, “The experience of derivative-driverless cooperative vehicle team in the 2016 gcdc,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1322–1334, 2017.

[2] M. Walker, Why We Sleep: Unlocking the Power of Sleep and Dreams. Scribner, 2017.[Online].Available:https://books.google.com.qa/books? id=8bSuDgAAQBAJ

[3] W. H. Organization, “The top 10 causes of death.” [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/ the-top-10-causes-of-death.

[4] T. A. Dingus, S. G. Klauer, V. L. Neale, A. Petersen, S. E. Lee, J. Sudweeks, M. A. Perez, J. Hankey, D. Ramsey, S. Gupta, C. Bucher, Z. R. Doerzaph, J. Jermeland,

and R. R. Knipling, "The 100-car naturalistic driving study, phase II-results of the 100-car field experiment," Nat. Highway Traffic Safety Admin., Washington, DC, Dept. Transp., HS 810 593, 2006.

[5] W.A. Cobb., "Recommendations for the practice of clinical neurophysiology.," Elsevier, 1983.

[6] Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Image Net Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [**Google Scholar**] [**CrossRef**].

[7] Soares, S.; Monteiro, T.; Lobo, A.; Couto, A.; Cunha, L.; Ferreira, S. Analyzing Driver Drowsiness: From Causes to Effects. *Sustainability* **2020**, *12*, 1971. [**Google Scholar**] [**CrossRef**][**Green Version**].

[8] Kim Hong, Chung, "Electroencephalographic study of drowsiness in simulated driving with sleep deprivation.," International Journal of Industrial Ergonomics., Volume 35, Issue 4, April 2005, Pages 307-320.

[9] Haisong Gu, Yongmian Zhang, and Qiang Ji, "Task oriented facial behavior recognition with selective sensing, " Comput. Vis. Image Underst., vol. 100, no. 3, pp. 385–415, 2005.

[10] S. Ghosh, T. Nandy, and N. Manna, "Real Time Eye Detection and Tracking Method for Driver Assistance System," Adv. Med. Electron., 2015.

[11] Haisong Gu and Qiang Ji, "An automated face reader for fatigue detection," Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference, vol. 00, pp. 111, 2004.

[12] Zutao Zhang and Jia shu Zhang, "Driver fatigue detection based intelligent vehicle control," in ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition,Washington, DC, USA, 2006, pp. 1262–1265, IEEE Computer Society.

[13] MCDonald, A.D.; Schwarz, C.; Lee, J.D.; Brown, T.L. Real-Time Detection of Drowsiness Related Lane Departures Using Steering Wheel Angle. Proc. Hum. Factors Ergon. Soc. Annu. Meet. 2012, 56, 2201 -2205 [CrossRef]

[14] Zhao, Z.; Zhou, N.; Zhang, L.; Yan, H.; Xu, Y.; Zhang, Z. Driver Fatigue Detection Based on Convolutional Neural Networks Using EM-CNN. Comput. Intell. Neurosci. 2020, 2020, 7251280. [CrossRef] [PubMed]

[15] Chellappa, A.; Reddy, M.S.; Ezhilarasie, R.; Suguna, S.K.; Umamakeswari, A. Fatigue Detection Using Raspberry Pi 3. Int. J. Eng. Technol. 2018, 7, 29–32. [CrossRef]

[16] Sheth, S.; Singhal, A.; Ramalingam, V.V. Driver Drowsiness Detection System using Machine Learning Algorithms. Int. J. Recent Technol. Eng. 2020, 8, 990–993.