International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 06 | June - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

RESEARCH PAPER ON A DRIVER DROWSINESS DETECTION USING MACHINE LEARNING

Dr. Nilesh T.Gole[1] , Prof. Arvind M. Ganvir[2] , Namo Ganpat Dongre[3], Shweta D.Gawkhare[4], Dhiraj V.Dandekar[5], Yuwani S.Waghmare[6], Yuvraj S. Gatkal[7]

Department of Computer science &Engineering,Vit College,Nagpur

Abstract— **Road accident is one of the major reasons of deaths occurring now-a-days. Many of these disasters occur due to the drowsiness of drivers which cause many severe effects. To prevent these accidents, a system which can find whether the driver is drowsy or not is very helpful and can save many lives. This is an image-based system which uses Computer Vision and Machine Learning techniques (Convolutional Neural Networks) to detect the drowsiness of the driver and alerts the driver if he/she is feeling drowsy. This system uses facial features to detect if a person is feeling drowsy.**

Keywords— Computer Vision, Machine Learning, Convolutional Neural Networks

I. Introduction

An accident may cause severe effects that may even lead to the death of travelers, a severe loss to many families. Drowsiness may be caused due to various situations like driving for long hours, not having rest. The accidents occurring due to this can be prevented if the state of the driver is being predicted. We propose a system developed with the help of the latest technologies that are available today to prevent these accidents.

There are several approaches to this problem. One of the methods to identify drowsiness is by using the facial features. This is an image-based system in which the facial expressions are considered by Feature extraction. Generally, people close their eyes or yawn when they are feeling drowsy. By considering these features, we can predict the state of the driver.

The dataset consists of two sets of images, alert and drowsy. The system design consists of the following 3 steps which will be explained in detail in the further sections. Face Detection is performed on each image in the dataset using Haar Cascades Algorithm[6][7]. The image is converted into grayscale and given to the algorithm and the face region can be extracted from the image. The Region of Interest (ROI) which is the face is then resized to a fixed number of pixels since all the images should be of equal size to be given as input to a neural network.

The CNN Classification model is then built by defining the layers. The feature extraction is performed in the layers by using kernels, also called filters. They convolve across the layer anddot product of the kernel and a window of the image is given outputto the feature map. The performance of model is improved after eachepoch by adjusting the learnable parameters by back propagation. The image frames are continuously captured from the camera, theface region is extracted using Haar Cascades algorithm and given to the previously trained Classifier. It can predict if the face is drowsyand generate the alert.

II. Literature Survey

There are various methods for detecting drowsiness. Some of the approaches which are used in this domain are discussed here.

A. Eye Aspect Ratio (EAR):

The Eye Aspect Ratio, or EAR, is a scalar value that responds, particularly for opening and closing the eyes [43]. During the flashing process, we can see that the EAR value grows rapidly or decreases significantly. Interesting findings in terms of robustness were obtained when EAR was used to detect blinks in , Studies in the past have employed a predetermined EAR threshold to establish when subjects blink (EAR threshold at 0.2). This approach is impractical when dealing with a wide range of individuals, due to inter-subject variation in appearance and features such as natural eye openness, as in this study. Our works used an EAR threshold value to detect a rapid increase or decrease in the EAR value caused by blinking, based on the findings of previous studies.

In this method [1], the eyes are of major concern. The eye aspectratio involves a very simple calculation based on the ratio of distances between facial landmarks of the eyes. Using this, the eye blinks are detected. If the eyes have been continuously closed for a certain threshold of time, the person is considered drowsy.

B. *Physiological Approch*

In this approach[2][13], physiological signals are collected from the driver and analyzed to predict the behavior. Features like electrocardiography (ECG) and electroencephalography (EEG) are included for better performance. The data that is collected from the ECG signal includes heart rate (HR) and heart rate variability (HRV), including low frequency, high frequency and LF/HF ratio. These combined measures are used to determine the state of the driver.

C. *Steering wheel data*

In this method[3], the data is extracted from the movements of the steering wheel of the vehicle. This is another method to find the movement of the vehicle and can be used to assess the driver's state. Steering wheel angle is dependent on various parameters like the road geometry and curvature. The effects of the type of roads are removed from the data and can be used to classify the state of the driver.

D. *Support Vector Machines (SVM)*

Support Vector Machines[4][25] are also based on Machine Learning for classification problems. The eyes or the face can be classified using SVM. There is a margin which acts as a boundary that differentiates various classes. SVM tries to maximize this margin. This is also a widely used classifier but it does not work well when it comes to large datasets.

E. *Artificial Neural Networks (ANN)*

Artificial Neural Networks[5] are Deep Learning techniques which are inspired by the functionality of human brain and neurons. These neural networks have hidden layers which are used for feature extraction. The main disadvantage of ANN is that it can't work very efficiently with 2D data.

III. Proposed System

A. *The Dataset*

The first step is collection of the dataset. We considered some of the datasets for this system.

A dataset called Drowsiness_dataset from Kaggle which contains images with not yawning faces and yawning faces. This dataset contains images of many people captured while driving when they are yawning and not yawning. This dataset also includes images of the people wearing spectacles. Another dataset from Github is considered in which the images of 3 people are recorded while driving. This dataset contains 3 sets of images which are alertimages, images with closed eyes and images with yawning faces. The images of closed eyes and yawning faces are combined as drowsy images.

We combined these images and our dataset has 2 sets of images i.e., alert and drowsy.

B. *System Architecture*

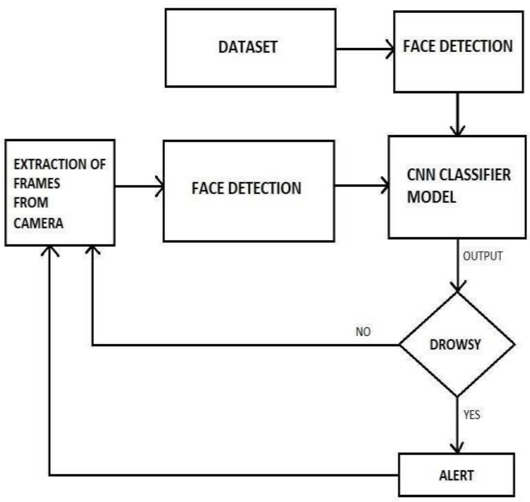


Fig. 1. System Architecture

C. *Face Detection*

The preprocessing of images in the dataset is performed in this section. Instead of giving the entire image to classification model, only the face region is extracted and given to the model since the background and other portion of the image is unnecessary. For this, we use Face detection which is a computer vision technology that locates human faces in digital images. Haar Cascades Algorithm[6], also known Viola-Jones Algorithm is used for Face detection which was proposed by Paul Viola and Michael Jones in their 2001 paper.

Working of the algorithm:
This algorithm uses Haar features to extract objects.

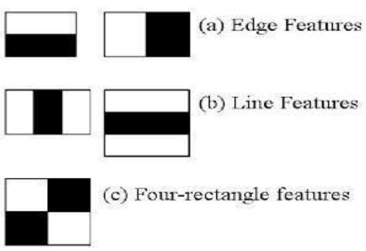


Fig. 2. Haar Features

The algorithm[6][7] applies the features on windows of the image, gradually changing the window size after each turn. If a particular window is not classified as a face, that window is not processed in further steps.
Instead of applying all the features, they are classified into different stages and applied one after the other. Only if a window passes a stage, the next stage is applied to that.
A face region is the one which passes all the stages.
The algorithm returns the starting coordinates, width and height of the detected face, using which the face region is extracted from the corresponding image and resized to a fixed size 100x100.

D. *The Classification model using Convolutional Neural Networks*

This is a Binary Classification problem as we are categorizing the images into two categories (alert and drowsy).

A Deep Learning[8] based model is used in this system. From the literature survey and existing systems[17][18][24], we have observed that the Deep Learning models provide better accuracy and results. Deep Learning algorithms work like the human brain. A Neural Network[9] has neurons arranged in the form of layers. The input passes through the layers starting from the input layer to the output layer and produces the output. The middle layers are known as hidden layers. We use convolutional neural networks[10][11] because they can deal with inputs having 2D format and extract spatial features.

Instead of extracting the features manually as in Traditional Machine Learning techniques, the features are automatically extracted in Neural Networks by the hidden layers. These can also perform well even if the dataset is too large. Neural Networks are generally trained using the Back Propagation algorithm.

The size of the input is defined as 100x100. The images are converted into arrays to be given to the classifier.

The layers in the model are:

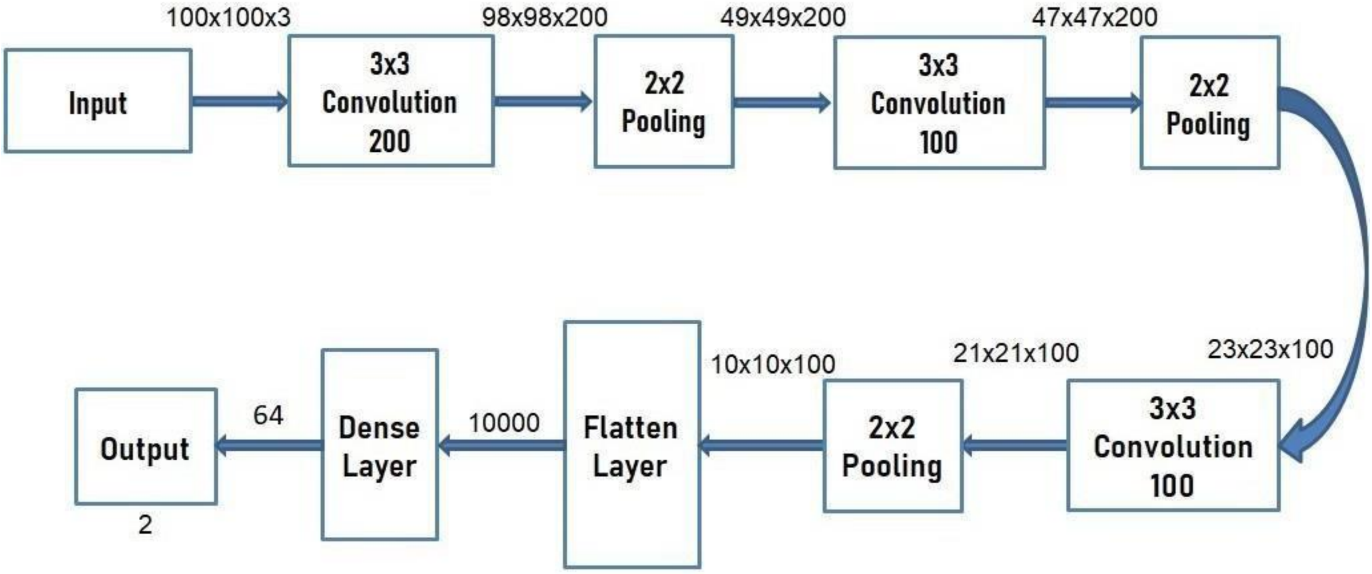


Fig. 3. CNN Architecture

- A 2D convolutional layer with ReLU activation function. The hyperparameters for this layer are 200 (number of kernels) and (3x3) (each of size 3x3). The shape of each input image to this layer is (100x100) and the output shape is (98x98x200). Here, 200 are the number of kernels.
- A Max Pooling layer of pool size (2x2). The output shape of this layer is (49x49x200).
- A 2D convolutional layer with ReLU activation function. The hyperparameters for this layer are 100 (number of kernels) and (3x3) (each of size 3x3). The output shape of this layer is (47x47x100). Here, 100 are the number of kernels.
- A Max Pooling layer of pool size (2x2). The output shape of this layer is (23x23x100).
- A 2D convolutional layer with ReLU activation function. The hyperparameters for this layer are 100 (number of kernels) and (3x3) (each of size 3x3). The output shape of this layer is (21x21x100). Here, 100 are the number of kernels.
- A Max Pooling layer of pool size (2x2). The output shape of this layer is (10x10x100).
- A flatten layer which consists of 10000 neurons. The shape of input to this layer is (10x10) with 100 kernels. So, the output from this layer has 10000 neurons.
- A dense layer with 64 neurons with ReLU activation function.
- The output layer with 2 neurons as this is a binary classification model. It has a sigmoid activation function.

We shuffle the data and split all the available images into train-data (80%) and test-data (20%). We run 20 epochs during training using the train data. While training the model, some part of the training data is used as validation data. We use 10% of the training data for validation. The validation data is used to evaluate the performance of the model at the end of each epoch but not used to train the model. The best model of all the epochs is saved by the end of training.

Testing is used to evaluate the generalizing ability of the model by giving unseen data to it. The test data is given to the model in this phase.

E. Predicting the images captured from the camera

After the model is trained with the given dataset, we can use it to predict the classes of the images captured from the camera.

OpenCV is used to capture the images from the camera. Image frames are continuously captured. The same preprocessing steps which are applied on the dataset are applied on each frame captured i.e., detecting the face from the image frame, extract the Region of Interest and then resizing the Region of Interest to a fixed size (100x100). Then we convert the images into array format to give as input to the model.

Then, we can give a set of images to the trained Classification model to predict the labels for the images. If the driver is feeling drowsy, a voice alert is generated. An audio file is played using the playsound module in python.

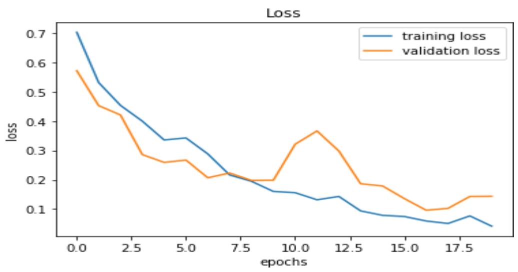
IV. Results

There are different Classification Metrics[15] to evaluate the performance of a Classifier.
Some important terms are:
True Positives (TP): It is the case where predicted output is positive and it is true (actual output is also positive). True Negatives (TN): It is the case where predicted output is negative and it is true (actual output is also negative). False Positives (FP): It is the case where predicted output is positive but it is false (actual output is negative). False Negatives (FN): It is the case where predicted output is negative but it is false (actual output is positive).

A. Loss

Loss is calculated after each epoch to measure the performance of the model. We use Categorical Crossentropy because this is used when there are two or more label classes.

Fig. 4. Training and Validation Loss



B. *Accuracy*

Classification accuracy is the ratio of number of correct predictions to all predictions made.

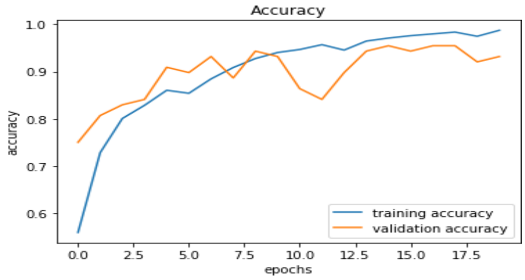


Fig. 5. Training and Validation Accuracy

C. *Precision*

Precision is defined as the ratio of number of TP to the number of TP plus the number of FP.

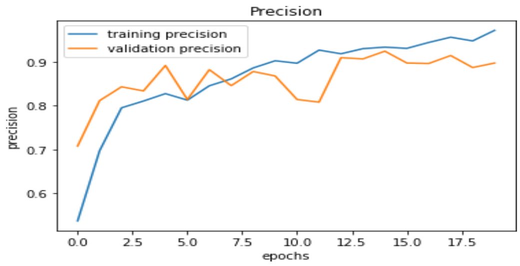


Fig. 6. Training and Validation Precision

D. *Recall*

Recall is defined as the ratio of number of TP to the number of TP plus the number of FN.

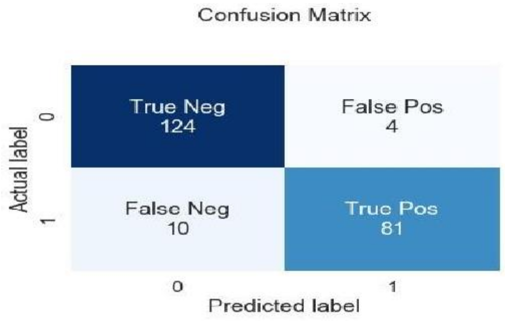


Fig. 8. Confusion Matrix

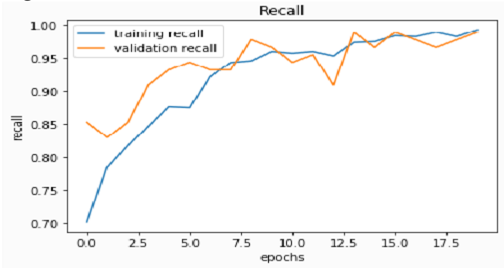


Fig. 7. Training and Validation Recall

E. *Confusion Matrix*

A confusion matrix is a table used to describe the performance of a classification model on test data. This uses the actual labels and predicted labels and how many are correctly and wrongly predicted.

The results on test data:
Test loss: 0.37542635202407837
Test accuracy: 0.9086757898330688
Test precision: 0.6204379796981812 Test recall: 0.77625572681427

V. Conclusion

This is an effective system which determines the drowsy state of the driver using facial features and generates an alarm in the form of audio to alert the driver. This system has three modules Face Detection, Classification and Predicting the captured images. Since this system uses Deep Learning techniques, it can predict the labels with good accuracy and performance.

Reference

[1] Rosebrock, A. (2017, May 8). Retrieved from Pyimagesearch: https://www.pyimagesearch.com/2017/05/08/drowsiness-detectionhttps://www.pyimagesearch.com/2017/05/08/drowsiness-detection-opencv/opencv/

[2] Awais M, Badruddin N, Drieberg M. A Hybrid Approach to Detect Driver Drowsiness Utilizing Physiological Signals to Improve System Performance and Wearability. Sensors (Basel). 2017 Aug 31;17(9):1991. doi: 10.3390/s17091991. PMID: 28858220; PMCID: PMC5620623. https://pubmed.ncbi.nlm.nih.gov/28858220/

[3] Arefnezhad, S., Samiee, S., Eichberger, A., & Nahvi, A. (2019). Driver Drowsiness Detection Based on Steering Wheel Data Applying Adaptive Neuro-Fuzzy Feature Selection. *Sensors (Basel, Switzerland)*, *19*(4), 943. https://doi.org/10.3390/s19040943

[4] B. K. Savaş and Y. Becerikli, "Real Time Driver Fatigue Detection Based on SVM Algorithm," 2018 6th International Conference on Control Engineering & Information Technology (CEIT), 2018, pp. 14, doi: 10.1109/CEIT.2018.8751886. https://ieeexplore.ieee.org/abstract/document/8751886/

[5] Vesselenyi, T., Moca, S., Rus, A., Mitran, T., & Tătaru, B. (2017, October). Driver drowsiness detection using ANN image processing. In *IOP Conference Series: Materials Science and Engineering* (Vol. 252, No. 1, p. 012097). IOP Publishing. https://iopscience.iop.org/article/10.1088/1757-899X/252/1/012097/meta