Grenze International Journal of Engineering and Technology, June Issue

Driver Safety System using Internet of Things

P Vikranth Reddy[1], Sahil Bavariya[2], Shradhya Rakshit[3], Joshua Dsouza[4] and Priya Badrinath[5]
[1-5]PES University, Dept. of Computer Science & Engineering,Bangalore, 560085, India
Email: padigivikranthreddy2018@gmail.com,
shradhyar4420@gmail.com,Joshuadsouza999@gmail.com,Bavariyasahil@gmail.com,priyab@pes.edu

***Abstract*—Driver drowsiness or drunk driving is a major contributing factor in many road accidents around the world. We're all also aware of cases of drunk driving. We predict an increase in distraction-related collisions as wireless communication, entertainment, and driver assistance systems proliferate in the car industry. concentrating on preventing such accidents and preventing the loss of other lives The strategies for detecting driver inattentiveness and distraction based on behavioral data and deep learning algorithms are surveyed in this research. We were able to extract some helpful algorithms and strategies from the research papers that we might apply in our future work to improve accuracy and navigate challenges. These include yawning, face detection, head movements, entity detection, and blinking. However, because accurate and robust algorithms are required, developing a dependable and effective sleepiness detection system is a challenging task. The identification of driver distraction and sleepiness has been studied using a variety of techniques in the past. To evaluate how successfully these algorithms recognise drowsiness in light of the recent advancement of deep learning, modifications to these algorithms are necessary. We propose an Ensemble Averaging technique using state-of-the-art CNN models like Inception-V3 and VGG-16 which yielded us an accuracy of 97.5% on the validation data. This serves as the backbone to our main IOT system which will conduct a real time driver drowsiness, distraction and alcohol detection in the driver.**

***Index Terms*— Driver Drowsiness, Distraction Detection,Drunk Detection,Eye Aspect Ratio(EAR), Mouth Aspect Ratio(MAR),Deep Learning, Computer Vision, Convolutional Neural Network.**

I. INTRODUCTION

Every year, almost 1.3 million people lose their lives in traffic accidents. Additionally, it's estimated that between 20 and 50 million people experience non-fatal injuries in accidents. According to statistics, driving while intoxicated is most often caused by speeding, fatigue, and inattentive driving. It is an ambitious goal of the UN General Assembly to reduce the number of traffic accidents worldwide by half by 2030. The use of cellular cell phones as driver assistance tools has significantly increased traffic accident rates. The main cause of such mishaps is still human error, therefore establishing a system that would warn people and forbid them from engaging in such activities will greatly reduce accidents. Accidents have always frequently been caused by drunk driving. The driver is still the key factor in preventing such accidents when it comes to control and actions, even though there are a number of alerting systems that have been implemented over the years in the vehicles.
We give this comprehensive overview of pertinent recent research in order to have a better knowledge of driver tiredness and distraction techniques. Each approach employs several deep learning approaches, both from scratch

Grenze ID: 01.GIJET.9.2.88

and utilizing models that have already been trained, and uses them to carry out many categorization tasks. To identify features on the face and the entire body, computer vision techniques are also used.The main input for our detection system would be a live video feed using a camera (Raspberry Pi Camera Module V2) which will subsequently be divided into frames and each individual frame will be analyzed. Using image processing techniques, facial recognition will be implemented on these frames wherein 68 facial landmarks will be identified on the person’s face and will further be used in eye and mouth tracking. Tracking the state of the eye and the mouth, we can determine whether the person is drowsy or not. As far as the distraction detection is concerned, we shall have certain thresholds which will be compared with the input to see if the driver is focused on driving i.e. a time constraint for how long the eyes are focused on the road and if the driver’s eyes are averted for longer than the threshold then he/she shall be alerted to take a break.Testing for the alcohol concentration in the air is also another aspect of our project wherein we will determine whether the driver is driving under the influence. We are utilizing MQ3 sensors for checking the concentration of alcohol in the driver’s breath and/or in the air around him. If alcohol is detected then the driver would be alerted using an alarm.The combined implementation of these three aspects of our project will help us bring forth a system that will ensure a better and safer driving experience for both the driver and the pedestrians involved.

II. RELATED WORKS

There are several techniques and models employed to detect both drowsiness and distraction.Eye blinking rates are a very common method to determine whether the driver is drowsy as mentioned in the papers below.

Muhammed et al. [1] This system takes into account a straightforward real-time drowsiness detection technique that is only dependent on the eye blinking rate calculated from the ocular aspect ratio. HOG and a linear SVM are used for eye detection. The technology sounds an alarm to stop the driver from dozing off if the rate of eye blinking falls below a certain empirical level. In this study, we thoroughly assess the essential conditions for the suggested system. We find that this method performs well when the face is turned towards the camera, but it degrades when the head is significantly inclined. Our assessments' findings serve as the cornerstone for developing our sleepiness detection technology further. Suhaiman et al. [2] The method proposed recognises when a driver is falling asleep and an alarm is used to notify them. Current approaches have significant drawbacks because there are many environmental parameters available. The camera's ability to accurately recognise the face and the eye is impacted by the lighting. Due to either late or no detection, this has a significant impact on image processing techniques and significantly reduces accuracy.

In order to ascertain the driver's condition, this system suggests a real-time detection system that makes use of a camera and image processing libraries that collect and analyze the driver's full face and eye.The main method for image detection is feature extraction from facial landmarks.There are 68 identified landmarks. Oraan et al. [3] the goal of this research was to devise an approach for waking up drowsy drivers while they are on the road. As a result, this study made an experiment to measure the degree of drowsiness in an effort to remedy the problem. The use of a Raspberry Pi Camera and Raspberry Pi 3 module, which could estimate a driver's level of drowsiness, was a necessity for this study. The frequency of eye blinking and head tilting was employed to gauge a driver's level of drowsiness.Using the Raspberry Pi as the hardware component and image processing libraries, a real-time face detection can reliably identify the face and its features.A HAAR Cascade classifier is used for face detection. Vesselenyi et al. [4] utilizes still images taken of the driver to analyze the condition of the driver’s eyes. The main methods used are EEG (Electroencephalography) and EOG (Electrooculography) which show the proportion of the optical i.e. whether the eye is closed or open.

The brain activity is monitored through a sensor placed on the skin and the signals from around the eye muscles are measured.Navya et al. [4] employs an approach which utilizes bioindicators and behaviors of the driver alongside various machine learning techniques such as PERCLOS, HAAR cascade to determine the state of the driver. Deep et al. [5] this paper conducted a study on all the major deep learning techniques used for distraction detection of a driver and compared them based on various parameters. This paper gave us a brief overview of all the algorithms and what would be a favorable approach for our own system. Belal et al. [6] this paper focuses on both drowsiness and fatigue. They proposed an Advanced Driver Assistance System (ADAS) which will help detect drowsiness/fatigue using visual information from a live camera capture coupled with Artificial Intelligence.

Mahek et al.[7] also proposes a similar technique of using Eye Aspect Ratio for determining drowsiness of a driver. Tianchi et al.[8] proposes a semi-supervised extreme learning technique which is an improved version of Extreme Learning Machine (ELM) and it has shown improved performance when it comes to classifying unlabeled images. ELM is a single feed-forward neural network called ELM uses analytical output weights between hidden neurons and output nodes and randomly generated hidden mapping function parameters.Zaeem et al. [9] proposes 2 DCNN architecture models (i.e. Inception V3 & Xception) and did comparison between them. Inception V3 has 44 layers and has 21M learnable parameters. Xception is a 36 layer model with 20M parameters and it is a VGG-esque architecture. Xception model outperforms Inception V3 model by slight difference on ImageNet dataset with 79% accuracy compared to 78.2% accuracy of Inception model. Xception converges faster than Inception V3. More variations can be seen on validation graphs of Xception than on that of Inception. Xception takes more training time as compared to the Inception model with 28 steps/sec and 31 steps/sec respectively. traffic systems.This study focuses only on one specific distraction therefore the scope is limited.

Ankita et al.[10] also proposes a method similar to the aforementioned ones wherein the Region of Interest (ROI) is extracted from the face and bounding boxes are made around the eyes and they are tracked and monitored for changes and the driver is alerted accordingly. Adnan et al.[11] proposes a real time driver distraction detection method using the SqueezeNet architecture. SqueezeNet is suitable for embedded deployment.The method demonstrated here entails making minor adjustments to SqueezeNet so that it may be trained on the AUC Distracted Driver Dataset, producing accuracy levels as high as 93% and detection speeds as high as 11 FPS. Salma et al.[12] proposes a fatigue and distraction detection system using two AlexNet coupled with non-negative matrix factorisation for feature reduction and an SVM classifier.The trials revealed that while the feature extraction SVM-based model performed better, with an accuracy of 99.65%, our suggested transfer learning model only managed to reach an accuracy of 95.7%.

Rupali et al.[13]conducts a very detailed study on various deep learning models and benchmarks them with respect to the distraction detection problem.10 state-of-the-art models are compared using the average cross-entropy loss, accuracy, F1-score and training time.We chose this paper as the base paper for our research, Results show that pre-trained Inception-V3 CNNs alongside stacked Bidirectional Long Short Term Memory performs better than state-of-the-art CNN and RNN models with an average loss and F1-score of 0.292 and 93.1% respectively. Elena et al.[14] also proposed a similar ADAS (Advanced Driver Assistance System) where the driver is alerted if changes in his behavior are noticed i.e if he appears to be drowsy or fatigued. Recurrent and convolutional neural networks are used in the second alternative, which they brought into a system later known as a fuzzy logic-based system. On both systems, accuracy of up to 65% was attained. Also, the accuracy of the test data is 60% and that of the training data is 65%. Furthermore, the fuzzy logic-based system was notable for achieving a specificity of 93% while avoiding false alarms.

III. Proposed Methodology

We briefly discuss the methods employed for our two main modules :Drowsiness Detection and Distraction Detection of the driver. The overall gist of the methodology involves detection of whether the driver is drowsy by tracking key facial features of the driver viz eyes and mouth. The blinking rate of the eye alongside the duration of the yawn is considered. On the other hand, for distraction detection we consider 10 classes of distractions which our model will classify in accordance to the drivers actions as captured by the live video feed. In both cases, a camera captures a live video and we extract frames from said video which are further input into their respective models for classification. An alarm is rung in both cases in order to alert the driver and various pre-recorded messages are also played telling the driver to halt or to take some rest for a while.

A. Drowsiness Detection

Drowsiness Detection of the driver involves tracking the driver’s eyes and mouth and identifying whether the eyes are closed or open, identifying whether the mouth is closed or open. To achieve this goal, we built an inception model with an accuracy of 95.87%. The first dense layer has the output space of 256 and “RELU” activation function is used. The second dense layer has the output space of 64 and “RELU” activation function is used. Then a dropout layer of 0.5 value is added to reduce the contribution of some neurons. The third dense layer has the output space of 2 and “SOFTMAX” activation function is used. But there was one big issue with this model, the size was too large for a 4GB Raspberry Pi to handle. Therefore, the Raspberry Pi started lagging while taking frames and that is not acceptable as the result should be coming in an instant else it is of no use. So we decided to use an in-built library in python. dlib library uses a 68-Landmark points method to detect the face of the person in an image. We have used the Eye Aspect Ratio (EAR) to check whether the eye and mouth are open or closed and the EAR we have come to is 0.2. If eyes are closed for more than the threshold period i.e. if we find the driver's eyes are closed for 18 frames , then the driver is drowsy and we alert him/her by an alarm.. If

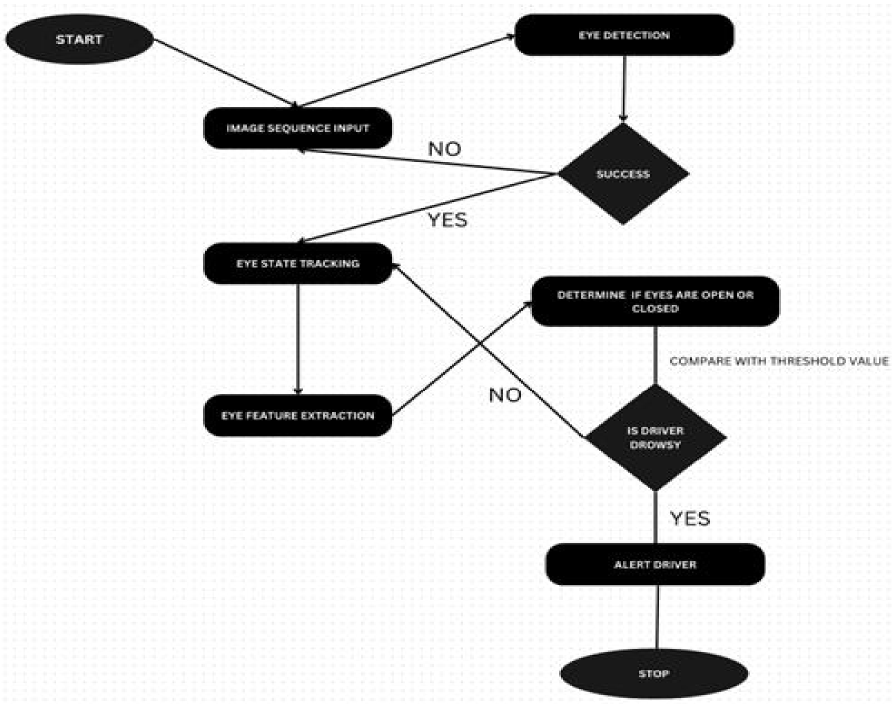

Figure 1. Sequence Diagram

the driver yawns more than the threshold value i.e. if we get that the driver has opened his month for 30 frames in one minute then we the system alerts the driver giving an appropriate message.

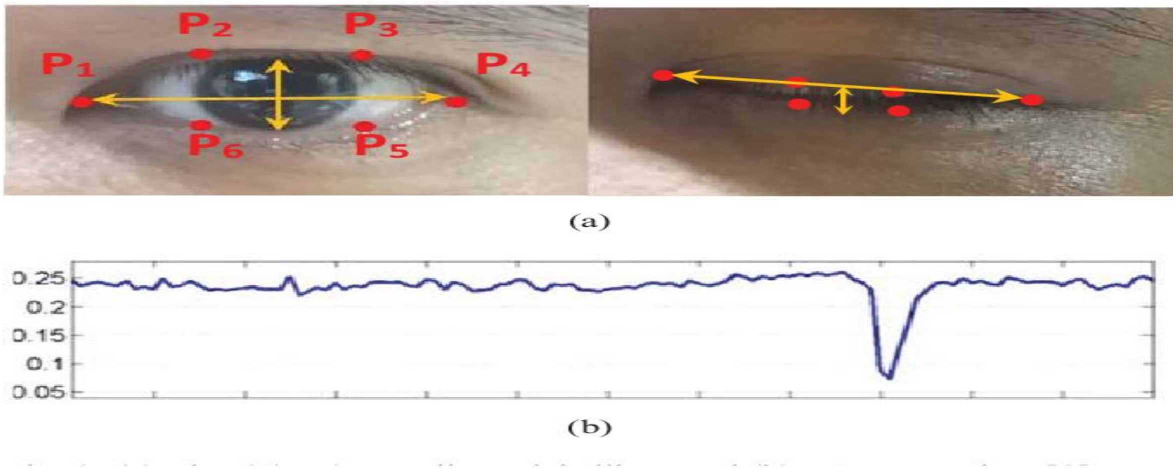

Figure 2.Eye Aspect Ratio(EAR)

B. Distraction Detection

Distraction Detection of the driver involves tracking the driver's action and the position in which they are sitting and identifying whether they fall under the 10 major classes of distraction. As per the State Farm Distracted Driver Dataset (22424 images) , which was created by the American University of Cairo (AUC) has named 10 classes of distractions that a driver can be found exhibiting while driving. Following are the 10 major classes of distraction:

- C0- Safe Driving
- C1- Texting- Left
- C2 - Texting on the phone- Left
- C3 - Texting- Right
- C4 - Talking on the phone- Right

- C5 - Operating the Radio
- C6 - Drinking
- C7 - Reaching Behind
- C8 - Hair and Makeup
- C9 - Talking to passenger

Images of 26 drivers have been taken showing all kinds of distractions exhibited with each driver having an average of 862 images.

C. Data Preprocessing and Loading

We read images from the respective path using python library CV2. We read the images in their original RGB form and we resize the images to 150 x 150 (rows and cols) since deep learning and machine learning models train faster on smaller images. We load the images from the training dataset and we append the images and their respective labels to a list. This will act as our training images and training labels. The next most important step is to normalize our data. It is a process by which we make our data either dimensionless or make sure that they have similar distributions throughout. It is known to improve model performance greatly. Normalizing our data will ensure that every variable is given equal importance or weight and thereby preventing any singular variable from steering the performance of the model. At this stage we reshape our data, turn them into numpy arrays and we conduct a train-test split of 80-20 which will be used later for testing our model performance.
We repeat the same steps for our test dataset and therefore create our validation data for testing of our model performance. We have a variable for mapping of our 10 labels. This constitutes the common data preprocessing that is done prior to training with our models. Initially we implemented a simple scratch CNN followed by two pretrained models i.e VGG-16 and Inception-V3. Further, an average ensemble is conducted on the two pretrained models and the best performing of the two is taken as our final model for multiclass classification.

D. CNN Model

We implemented our scratch CNN model using 6 convolutional layers along with the ReLu activation function which has an advantage over other activation functions in a way that it does not activate all neurons at the same time. There is no problem of vanishing gradient which makes the accuracy more efficient and faster. The batch size is 40 and it trains for 10 epochs. We add batch normalization after every convolutional layer and a dense layer after every second convolutional layer which helps in changing the dimensionality of the outputs of this layer and we add a standard dropout layer to reduce our number of inputs for the next set of convolutional layers. The softmax activation function to create a multinomial probability distribution for multiclass classification. We use categorical loss entropy, which is a combination of a softmax activation function and categorical loss entropy and is mainly used for multiclass classification.

E. VGG-16 Model

We employ the same data preprocessing methods as stated earlier and then we use a pretrained model called VGG-16. This model consists of 16 layers with certain weights (imagenet). It has convolutional layers and max pooling layers. It consists of 14 million total parameters which are all trainable.This architecture contains one input layer of shape 150 x 150 (rows and columns), followed by a functional layer. A flatten layer followed by a dense layer of shape 10 to reduce the output to the final 10 categories.

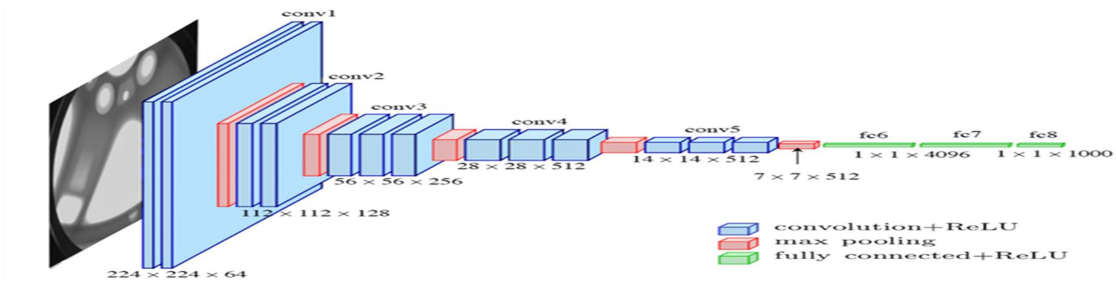

Figure 3. Convolutional Neural Network(CNN)

We employ the same data preprocessing methods as stated earlier and then we use a pretrained model called Inception-V3. This model consists of 21 million total parameters and 34K untrainable parameters. It consists of a total of 42 layers.

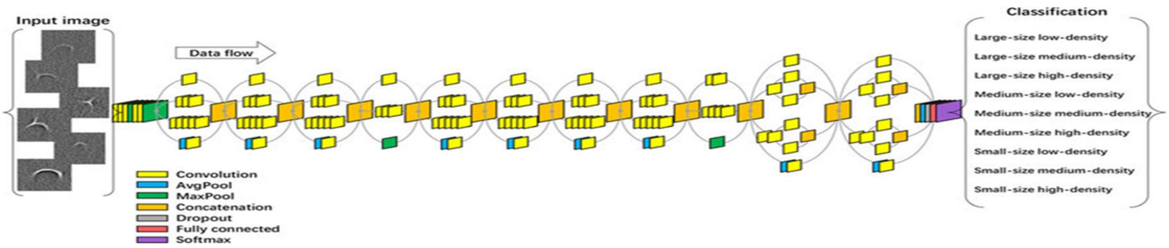

Figure 4. Inner Working Of CNN

We employed data augmentation for both the models to combat issues with data leakage i.e. misclassification of unlabeled images.

F. Ensemble - Averaging

To overcome the overfitting issues, we implemented an ensemble model by using an averaging method. We take the mean of predictions made by both Inception-V3 and VGG-16. We achieve an accuracy of 97% and hence we choose Inception-V3 as the model to make predictions of live video capture. We chose inception v3 since it is the closest to the ensemble average.

G. Alcohol Detection

We have used Arduino to support the MQ3 sensor. The components are:

- Arduino UNO
- MQ3 sensor
- Buzzer

The result is shown by Buzzer, when the sensor detects the presence of the ethanol in the air, it'll send a signal to the buzzer. The buzzer value will change from low to high.

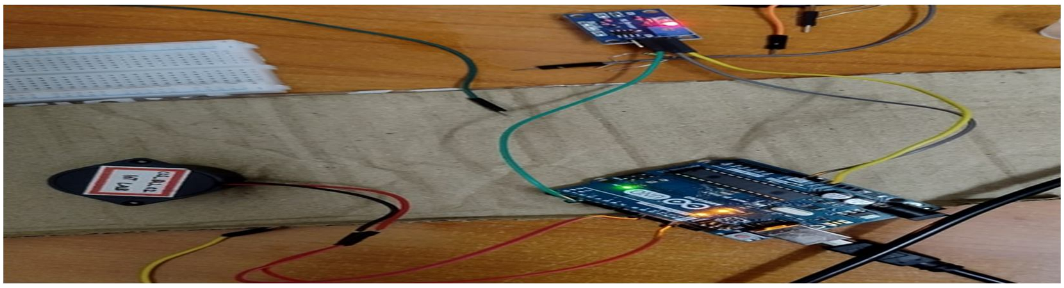

Figure 5. Alcohol Detection using MQ3 Sensor

H. Drowsiness Detection

Based on the threshold values provided, if the eyes remain closed for that given duration the driver is detected to be drowsy and he is subsequently alarmed with a buzzer and recorded messages telling him to halt or take rest. Similarly, based on the threshold values provided, if the driver yawns longer than that then he is detected as drowsy and is alarmed with a buzzer and recorded messages. This works well for when the driver's face is visible or it's in well lit condition and he is not covering his face. Future scope for this part of the project would be to make this system work under low light conditions and for faces with masks.

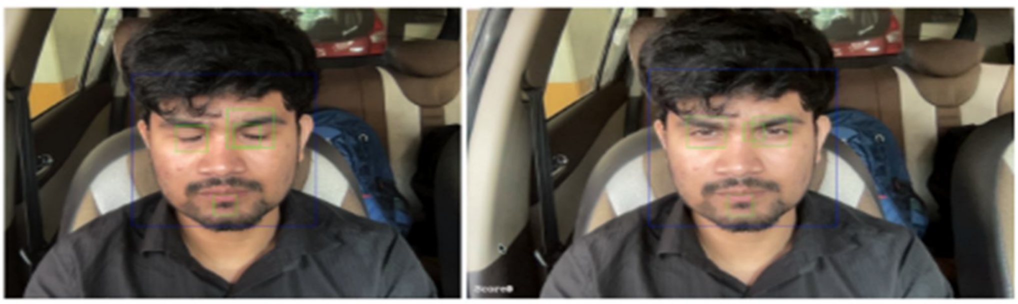

Figure 6.Driver Drowsiness Detection

I. Distraction Detection

CNN

The accuracy achieved is 98% and a loss of 0.04, despite which the model fails to accurately classify the unlabeled images in the testing dataset. This is due to the fact that the training dataset has multiple images of the same person with very slight variations which introduces the problem of data leakage in our model.

With Augmentation:

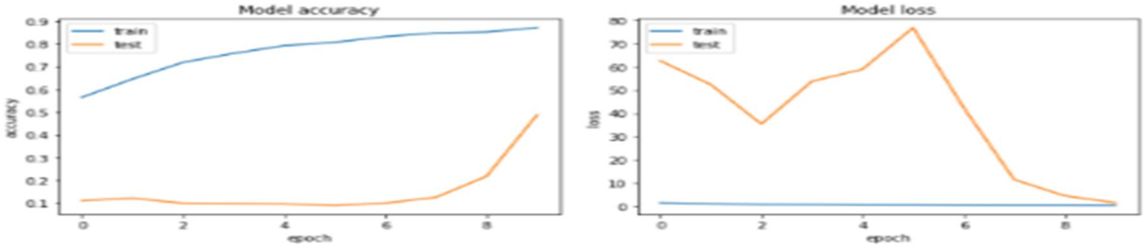

Figure 7. Model accuracy and loss using augmentation

To overcome this we have to use image augmentation and alter the existing images. Image Augmentation is done by rescaling the images and tweaking various parameters of an image to bring about variations in it. After augmentation, the accuracy on the model is 87% and the loss is 15% which means there is overfitting in the model. For such reasons we decided to use pretrained models.

VGG-16

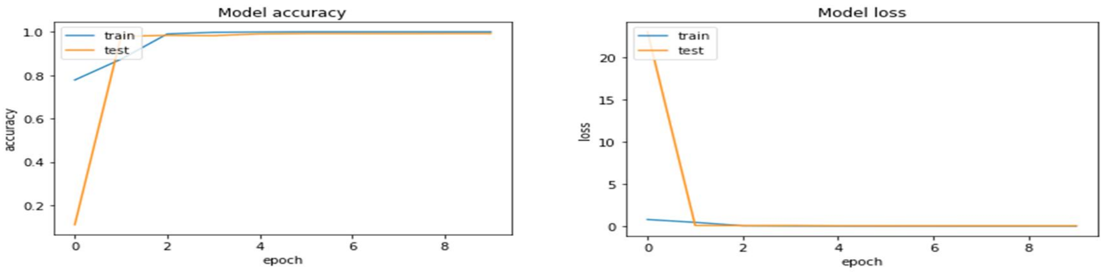

Figure 8. Model accuracy and loss using pre trainedweights (VGG 16)

The accuracy achieved is 99% with a loss of 0.04. This model also suffers from the issue of data leakage therefore image augmentation must be applied to our data.Post augmentation, we get an accuracy of 98% but the validation accuracy is much lower which indicates overfitting of the model.

Inception V3

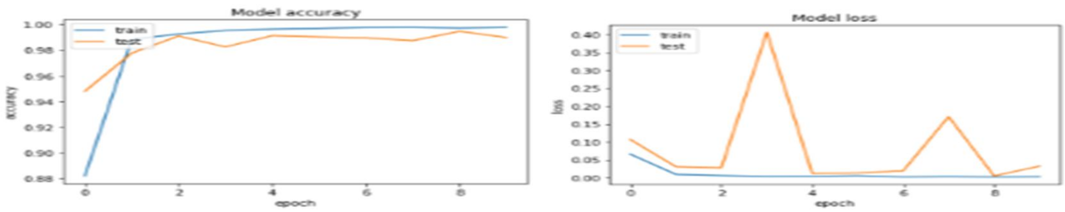

Figure 9. Model accuracy and loss using pre trained weights (Inception V3)

The accuracy achieved with this model is 98% and a loss of 0.04. This model also suffers with data leakage issues and therefore image augmentation is done. Post augmentation we achieve an accuracy of 80% and a loss of 70% which indicates there is a huge overfitting in the model.

Ensemble Averaging

We load our model weights i.e. VGG-16 and Inception-V3 and we make predictions using them. We calculate the mean of the predictions of both models and then we analyze which model performs the best in classifying all 10 categories and its value is closer to the average.The mean for the predictions was 97.5%. Inception-V3 performed better than VGG-16 in identifying all classes and hence we chose that as our final model.

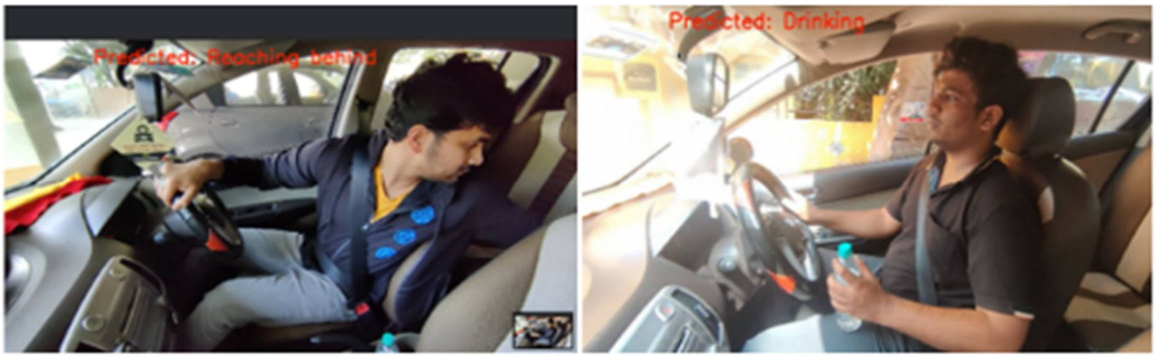

Figure 10. Driver distraction detection

APPENDIX

ACKNOWLEDGMENT

We must express our gratitude to Prof.Priya Badrinath for imparting their knowledge to us during this research. Although any inaccuracies are our own and shouldn't reflect negatively on the reputations of these distinguished professionals, And also very thankful for the comments on earlier draughts of the work.

REFERENCES

[1] Muhammad Fawwaz Yusri, Patrick Mangat, and Oliver Wasenmiller, “Detection of Driver Drowsiness by Calculating the Speed of Eye Blinking”, arxiv.org, 21 October 2021.

[2] Suhaiman, A.A., May, Z., Rahman, N.A.A, “Development of an intelligent drowsiness detection system for drivers using image processing techniques”, IEEE Student Conference on Research and Development, 28 September 2020.

[3] Oran Khunsiputh, Taweechai Chotchinasri, Varakorn Koschakosai, “Driver Drowsiness Detection using Eye-Closeness Detection”,12th International Conference on Signal-Image Technology & Internet-Based Systems, 2016.

[4] V B Navya Kiran, Raksha R, Anisoor Rahman, Varsha K N, Dr.Nagamani N P, “Driver Drowsiness Detection”, International Journal of Engineering Research & Technology (IJERT),Volume 8, Issue 15, 2020.

[5] Deep Ruparel,Abhay Rajde,Sahil Shah,Prof..Manya Gidwani, “Distracted Driver Detection”, Journal of Emerging Technologies and Innovative Research,Volume 7 Issue 10,October-2020.

[6] Belal Alshaqaq, Abdullah Salem Baquhaizel, Mohamed El Amine ouis, Meriem Boumehed, Abdelaziz Ouamri,Mokhtar Keche, “Driver Drowsiness Detection”, 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA),2013.

[7] Mahek Jain,Bhavya Bhagerathi, Sowmyarani C N, “Real-Time Driver Drowsiness Detection using Computer Vision”, International Journal of Engineering and Advanced Technology (IJEAT),Volume-11 Issue-1,October 2021.

[8] Tianchi Liu, Yan Yang,Guang-Bin Huang,Yong Kiang Yeo,Zhiping Lin, “Driver Distraction Detection Using Semi-Supervised Machine Learning”, IEEE Transactions on Intelligent Transportation Systems Volume: 17, Issue: 4, April 2016.

[9] Zaeem Ahmad Varaich, Sidra Khalid, “Recognizing actions of Distracted Drivers using Inception V3 and Xception Convolutional Neural Networks”, 2019 2nd International conference of Advancements in Computational Science (ICACS), 18 Feb 2019.

[10] Prof.Ankita V. Karale, Nikita P. Patil, Prajakta M. Sarvar,Pritam M. Sangle, Shila A. Shinde, “Driver Drowsiness Detection System”, Journal of Emerging Technologies and Innovative Research(JETIR) Volume 8, Issue 5, May 2021.

[11] Adnan Shaou ,BenjaminRoytburd,LuisAlejandroSanchez-Perez, “An Embedded Deep Learning Computer Vision Method for Driver Distraction Detection”,, 22nd International Arab Conference on Information Technology (ACIT),2021.

[12] Salma Anber Wafaa Alsaggaf Wafaa Shalash, “A Hybrid Driver Fatigue and Distraction Detection Model using AlexNet Based on Facial Features”, Multidisciplinary Digital Publishing Institute(MDPI), January 2022.

[13] Rupali Pawar, Saloni Wamburkar, Rutuja Deshmukh, Nikita Awalkar, “Driver Drowsiness Detection Using Deep Learning”, Global Conference for Advancement in Technology, October 2021.

[14] Jimiama Mafeni Mase1, Peter Chapman, Grazziela P. Figueredo, Mercedes Torres Torres, “Benchmarking Deep Learning Models for Driver Distraction Detection”,International Conference on Machine Learning, Optimization, and Data Science, January 2020.

[15] Elena Magán, M. Paz Sesmero, Juan Manuel Alonso-Weber and Araceli Sanchis, “Driver Drowsiness Detection by Applying Deep Learning Techniques to Sequences of Images”, Multidisciplinary Digital Publishing Institute(MDPI), 22 January 2022.