International Journal of Artificial Intelligence Trends (IJAIT)
Vol. 4, Issue 2; No. 55, February 2025, pp. 576-584

A HYBRID DEEP LEARNING APPROACH FOR THE EARLY DETECTION OF DRIVER DROWSINESS

Engr. Dr. Harbor M.C.(Ph.D.)
Nigerian Defence Academy
Email: [1*] *dumbormart@gmail.com*

Article Info
Received: 4/2/ 2025
Revised: 30/2/2025
Accepted 2/3/2025
Corresponding Authors Email:
dumbormart@gmail.com
Corresponding Author's Tel:
+2348068215797

ABSTRACT

Since driver fatigue is a major contributor to traffic accidents globally, identifying it early on is essential to improving road safety. Considering the condition of the driver's eyes, this study presents a deep learning-based method for identifying driver drowsiness using face cues. The study uses publicly available dataset from Kaggle, a hybrid model that combines Convolutional Neural Networks (CNN) for feature extraction and VGG16 for transfer learning was developed and trained. Four categories such as open, closed, yawning, and no yawning, based on the level of tiredness are annotated on the dataset's images. To enhance model performance and generalization, data preparation methods such as image scaling, normalization, and data augmentation were used. The proposed hybrid model produced impressive results, with 97.1% accuracy, 96.5% precision, 97.3% recall, and 96.9% F1-score. These results were validated using 10-fold cross-validation, which showed the model's robustness and generalizability across various dataset subsets. Based on the findings, the developed system is highly suitable for real-time driver tiredness detection and might be included into Advanced Driver-Assistance Systems (ADAS) to improve road safety. Future research might look at other input characteristics like head position or body gestures to further increase the recognition accuracy.

Keywords: Driver Drowsiness Detection; Deep Learning; CNN; VGG16; Transfer Learning, Kaggle

1. INTRODUCTION

Approximately 1.3 million people are killed in auto accidents annually, making them the top cause of mortality. The majority of these collisions are brought on by inattentive or distracted drivers. In addition to impairing focus, activity, and alertness, drowsiness also makes a driver less likely to make judgements or to make them at all. According to Kumar et al. (2024), drowsiness impairs mental awareness, decreases a driver's capacity to operate a vehicle safely, and raises the possibility of human mistake, which can result in fatalities and serious injuries. The driver's mistake rate has dropped. Day and night, countless numbers of people travel great distances by car. Accidents can be caused by sleep deprivation or distractions like chatting on the phone or interacting with a passenger. We propose a method that alerts the motorist whether they are drowsy or inattentive in order to prevent these collisions (Alvarez, 2012; Titare et al., 2021).

According to a data published by Nigeria's Federal Road Safety Corp (FRSC), for example, more than 12,077 traffic accidents were reported in the last quarter of 2015, resulting in 75400 deaths. According to a more recent yearly report from the same agency, during the first quarter of 2020, there were 3,947 traffic accidents involving 6448 cars, resulting in 1758 fatalities and 11250 injuries. All investigations pointed to reckless driving and tiredness (Precious et al., 2022).Therefore, intelligently based, cost-effective drowsy driver monitoring and detection systems are required in order to track drivers' actions while driving and provide precise results. The symptoms of drowsiness, which is characterised as a sensation of weariness, include decreased reaction time, sporadic unconsciousness, or microsleeps (blinks that last longer than 500 milliseconds). Every day, hundreds of drivers—including cab drivers, truck drivers, and long-distance travellers drive on roads when sleep deprived (Khan et al., 2022). Additionally, tiredness lowers a driver's level of attention, which leads to dangerous situations. This is one of the main causes of traffic accidents and greatly raises the likelihood that drivers would miss exits or road signs, veer into other lanes, or possibly get into a collision (Ahmad et al., 2023).

Every year, more people are killed and injured worldwide as a result of drowsy drivers. These days, artificial intelligence (AI) plays a big role in tackling a lot of problems throughout the world. For example, safety driver drowsiness detection technology can assist avoid accidents caused by drivers who fall asleep while driving, reducing the amount of incidents on the road that are caused by tiredness. Sleep disorders have been linked to a wide range of behavioural and general health problems, including poor driving ability (Tavakoli-Kashani et al., 2022). Lack of sleep, fatigue, poor road conditions, and tiredness are the main causes of thousands of accidents globally (Khan et al., 2022). The public health administration is worried about the possible role of poor driving, accidents involving people who are asleep at the wheel, and the rising number of fatalities and injuries brought on by these problems (Azam et al., 2014).

Early on in the drowsiness process, it is possible to detect driver weariness and notify the driver.

Numerous techniques have been put forth over time to assess driver weariness, including biological signal measurements related to tracking the driver's behaviour and vehicle movement, as well as analysis of the driver's motions and facial expressions (Marouf et al., 2017).Techniques for assessing lip motions, eye shutting, eye blinking, and head position are all part of the analysis of the driver's facial expressions and movements (Nagy et al., 2017; Vesselenyi et al., 2017a; Vesselenyi et al., 2017b; Vesselenyi et al., 2009). Physiological signals like electrooculography (EOG), electromyogram (EMG), electroencephalogram (EEG), and electrocardiogram (ECG) are among the biologically based techniques that are related to driver bio-signals (Nagy et al., 2018; Cardone et al., 2022; Ebrahimian et al., 2022; Arefnezhad et al., 2022; Huang 2020).

Driving behaviour, such as speed, steering wheel angle, lane position, duration in lane crossing, etc., is one of the techniques based on behavioural monitoring and vehicle motion (Nagy et al., 2017; Purnamasari and Hazmi, 2018). A number of these approaches have been thoroughly examined, and other articles have discussed the benefits and drawbacks of each. An effective drowsiness detection system would result from combining many techniques into a hybrid system. This study's primary contribution is the creation of a drowsiness detection system that uses computer vision techniques to recognise a driver's face in images and then applies deep learning techniques to determine whether or not the driver is tired or sleeping based on the image of their face in a real-time setting. This distinguishes the suggested model from the others. Finally, the proposed study primarily integrates two models, one of which is the designed CNN model and the other a pre-trained model (VGG16)and contrasts their effectiveness for the detection of driver drowsiness.

2. DESIGN METHODOLOGY

In this work, a deep learning model is built and trained using a dataset from Kaggle, a web-based data science platform where academics studying data and machine learning can access and share datasets for analysis and model construction. The deep learning model employed in this study is based on the CNN architecture. The model is first trained using the dataset that was acquired from Kaggle following pre-processing. The model gains the ability to recognize the telltale symptoms of fatigue in a driver's face through training. The system uses facial expressions and eye movements to categories the driver's state. The model is further evaluated, analyzed, and validated by considering accuracy, precision, recall, and F-score.

2.1 Data Collection

The model was developed following appropriate preparation of the dataset, which was sourced from open data sources such as Kaggle. While demographic variables such as gender and age group are included in the dataset, they were not specifically utilized in the analysis, such as gender-based or age-group-based studies. To train and evaluate the algorithm, the driver downiness dataset is made publicly available on Kaggle (Perumandla, 2020). The 2800 images in the collection are categorized into four groups according to the level of drowsiness: open, closed, yawning, and no yawning. The dataset makes it easy to comprehend the many eye disorders that are represented in it. The dataset contains a number of significant variables that improve the analysis of driver drowsiness in addition to the eye condition labels. The investigation of possible differences in drowsiness patterns across genders is made possible by the gender feature, which indicates the gender of the driver in the images. By classifying the drivers into several age groups, the age feature makes it easier to look at any significant variations in drowsiness patterns across different age ranges. The driver's head orientation in the images is described by the head position feature. It provides important information about the relationship between head posture and tiredness and if certain head positions are more common among drowsy drivers.

2.2 Data Pre-processing

One of the most crucial phases in increasing the effectiveness of any classification challenge is data pre-processing, which helps to clean and convert the data into more ideal and intelligible formats. The models were developed and the dataset pre-processed using

Google Colab (Google Collab, 2023). Since the backdrop of the image was distracting and unnecessary, we first just removed the face. The images were then downsized to the dimensions specified by VGG, which were 145 × 145 for the CNN model and 64 × 64 for the VGG16 model. Additionally, a NumPy array was created from the dataset. The images in the dataset are labeled as either open, closed, yawn, or no-yawn. To change the category target labels into 0, 1, 2, and 3, we applied one-hot encoding techniques to the labels. Using the sklearn library's LabelBinarizer() method, it was executed with the mappings 0 for yawn, 1 for no yawn, 2 for closed, and 3 for open. Furthermore, the dataset was separated into training and testing datasets. Thirty percent of the dataset was used for testing and assessing the performance of the final model, while the remaining seventy percent was used for training the model using CNN and VGG16 to predict the kind of drowsiness. Lastly, to extend the dataset and guarantee that the model receives various image variants by rotating the image to different angles, the data augmentation technique ImageDataGenerator was used to strengthen the model's resilience.

3. DESCRIPTION AND TRAINING OF THE PROPOSED CNN MODEL

A sequential CNN is a kind of artificial neural network used in deep learning that uses three different kinds of layers to filter inputs into useful information. First, the number of neurons in the input layer, which is where we feed images to the model, is mirrored by the total number of pixels in an image. Second, it makes use of hidden layers, which receive the input layer's output. The model and the volume of data are used to calculate the number of hidden layers. The number of neurons in each hidden layer might vary and is typically greater than the number of pixels in the image. Third, the output layer, which converts each class's output into its probability score by passing the output from the hidden layer into a logistic function. A CNN's feature map captures the outcome of applying filters to each layer image. Visualizing a feature map for a specific collection of images aims to increase understanding of the features identified by the suggested CNN. CNN has been used in the study to determine a driver's degree of tiredness by analyzing the condition of their eyes. The quantity of images in the dataset has a significant impact on the model's performance, and 2800 images were enough to train the suggested model well. The CNN model layers used were Conv2D, MaxPooling2D, Flatten, Dropout and Dense respectively (Pytorch, 2022). Each layer is briefly described subsequently.

Keras Conv2D is a two-dimensional convolution layer that winds a convolution kernel with the layer input to produce a tensor of outputs. As an additional example, the kernel is a convolution matrix or mask that may be used to execute a convolution between a kernel and an image in order to blur, sharpen, emboss, detect edges, and more. The kernel in this layer navigates across two-dimensional input and performs element-wise multiplication, adding the results to produce a single output. Figure 1 illustrates how the two-dimensional convolution process is carried out independently for each of the three color channels (Red, Green, And Blue (RGB)) in a colored image. The final output is then merged (MLK, 2022).

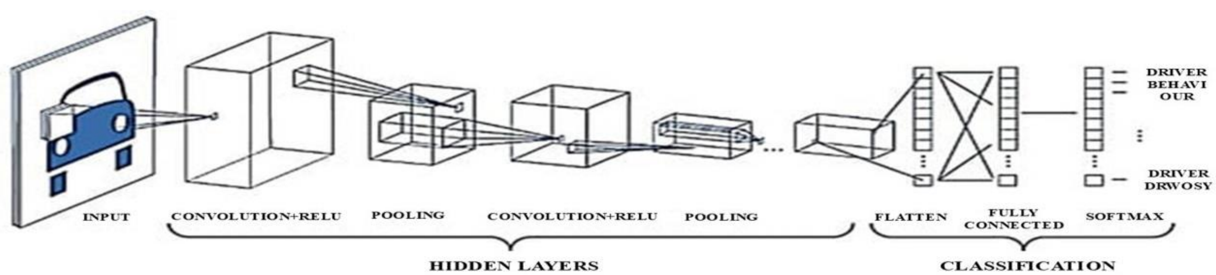

Figure 1: The General Flow of CNN Algorithm

The pooling procedure comprises epitomizing the features that fall within the coverage zone of a two-dimensional filter that is applied to each channel of the feature map. A pooling procedure called max pooling selects the peak element from the feature map region that the filter has contained. Consequently, the max-pooling layer's output would be a feature map that includes the most noticeable elements from the previous feature map (CNN, 2022). The process of flattening involves preserving batch size while transforming a matrix produced by convolutional and pooling layers into a single

International Journal of Artificial Intelligence Trends (IJAIT)
Vol. 4, Issue 2; No. 55, February 2025, pp. 576-584

features vector. Given that ANNs receive a one-dimensional array as input, this layer is crucial (Tensorflow.js, 2022). A dropout layer is one that keeps the remaining neurons unchanged while blocking some from contributing to the following layer. Overfitting-prone data are trained using this layer. The first set of training data may disproportionately harm subsequent samples if this layer is absent, preventing them from learning characteristics that are exclusive to later samples. The dropout layer creates a better implicit mode and significantly improves the fundamental architecture (Pelt and Sethian, 2018). The unassuming layer with neurons is called the dense layer. In this layer, every neuron sends inputs to every other neuron. The dense layer classifies the image using the output from the convolutional layers. This procedure produces a structure that, for a single set of operations, uses a small number of components and parameters to get precise results.

3.1 Model Development

Because of its ability to accurately recognize objects, the Haar cascade classifier is a popular and extensively used method in computer vision. The main goal of this study's machine learning-based object detection technique, the Haar cascade classifier, was to identify faces. Following a successful face identification procedure, the image processing pipeline resized and cropped the identified face areas before classifying and storing them. Additional processing and analysis, including feature extraction and classification, were made easier by the standardized format of cropped and scaled face areas. Furthermore, the Haar cascade classifier was selected due to its high degree of accuracy in face detection, which was essential for producing trustworthy study findings. The CNN architecture was created to train the model to recognize a driver's eye and mouth conditions in order to determine how drowsy they are. The design includes MaxPooling2D, Flatten, and Dropout layers, as well as a Conv2D layer with a "ReLU" activation function and "he_normal" kernel initialisers. Lastly, because the classification output is multilabel, a dense layer with a "SoftMax" activation function is used. The transfer learning model VGG16 was trained on the large ImageNet dataset using the CNN architecture (ImageNet, 2023). Three dense (completely linked) layers and thirteen convolutional layers make up the model's sixteen layers. The information that the model learnt during training could serve as a good foundation for determining the degree of tiredness. The model and its weights were loaded using the Keras framework. Additionally, a flattened layer was incorporated to flatten the VGG16 model's output so that it could be used as an input for the dense, fully linked layer. Lastly, a "SoftMax" activation mechanism was added to the dense output layer.

3.2 Model Training

Finding the ideal collection of parameters for generalizing incoming data while avoiding overfitting and under fitting is the aim of the model training process. The model was trained using a variety of regularization and optimization techniques throughout this stage. Given that this is a multi classification issue, optimization techniques like Adam were able to modify the model parameters during training in order to provide the least possible loss by using the "categorical_crossentropy" loss function. Additionally, a suitable learning rate was established to help "Adam" update the model's parameters. To avoid overfitting, regularization techniques were applied. In order to improve the model's performance, the VGG16 model employed L2 regularization. Furthermore, both models used an early stopping strategy that enables the specification of several training epochs while tracking the model's performance. In order to prevent overfitting and improve the model's generalization, this method will halt training as soon as the model's performance stops getting better. Figure 2 displays the model of the suggested intelligent model.

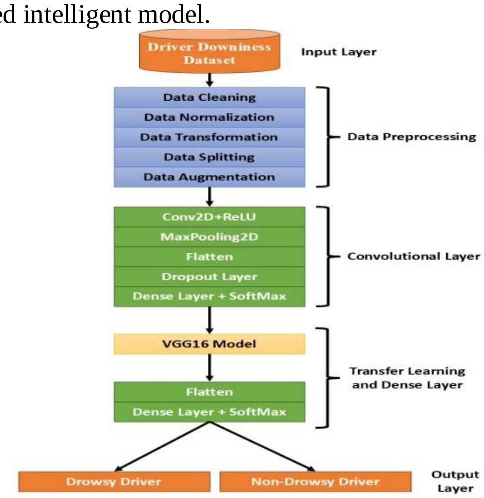

Figure 2: Model of the Proposed System

Based on face data taken from images, the driver drowsiness detection model shown in Figure 2 uses a deep learning technique to categorize drivers as either drowsy or not. To ensure consistent convergence during training, the dataset is subjected to a thorough data preparation step. During this phase, pixel values are normalized using Z-score normalization, which centers them on a mean of zero with a unit variance. Gaussian noise addition is a data augmentation approach that simulates real-world variables such as illumination changes and sensor noise in order to improve model generalization. In order to reduce dimensionality and preserve the most important characteristics while lowering computing cost and avoiding overfitting, Principal Component Analysis (PCA) is also used.

To extract hierarchical visual characteristics, the processed data is then run through a Convolutional Neural Network (CNN), which consists of MaxPooling2D layers, Dropout layers, Flatten layers, and Conv2D layers with ReLU activation. A pre-trained VGG16 model is integrated via transfer learning to further improve the feature extraction process. Fully linked Dense layers with SoftMax activation are then used for final classification. Driver drowsiness is detected with greater accuracy and resilience thanks to the model's training on enhanced and optimized data. The method is quite useful for real-time applications in road safety since the final classification layer decides if a driver is drowsy or not.

4. SYSTEM IMPLEMENTATION

On Google Colab, the Driver Drowsiness Detection System was put into practice by utilizing VGG16 Transfer Learning and Convolutional Neural Networks (CNN). The dataset was pre-processed by scaling the images to a fixed dimension (145x145), normalizing the pixel values, and dividing the images into training and testing sets. The dataset included images of drivers with open, closed, yawning, and non-yawning faces. Multiple convolutional layers, max-pooling, dropout, ReLU activation, and a fully connected layer with SoftMax activation for classification were all included in the CNN model. Furthermore, a pre-trained model called VGG16 was employed as a feature extractor to increase classification accuracy while maintaining learnt representations by freezing its convolutional layers. The main assessment metrics for both models were accuracy, categorical cross-entropy loss, and the Adam optimizer.

To enhance generalization, data augmentation techniques like as rotation, width/height shifting, and flipping were used during training. Accuracy and loss plots were used to assess the models' performance after they were trained on the enhanced dataset for 20 epochs. Because of its deeper architecture and pre-learned characteristics, the VGG16 model usually performed better than the custom CNN. After training, the models were stored as HDF5 files for further use. By combining the trained model with OpenCV to take and categorizes real-time driving images, the system may be expanded for real-time drowsiness detection, improving road safety by warning drowsy drivers before collisions happen.

5. RESULTS AND DISCUSSION

The Hybrid CNN-VGG16 Model was evaluated using multiple performance metrics, including accuracy, precision, recall, and F1-score, to measure its effectiveness in detecting driver drowsiness.

5.1 Evaluation of the CNN-VGG16 Training Results

The developed CNN-VGG16 model achieved an accuracy of 96.8% in the detection of the driver’s level of drowsiness. Table 1 shows the training report, which displays the precision, recall, F1, and support scores for the CNN-VGG16 model.

Table 1: Training report of the CNN-VGG16 Model

Accuracy (%)	**Precision (%)**	**Recall (%)**	**F1-Score (%)**
96.8	96.1	97.2	96.6

Figure 3 shows the history of fitting the CNN model, which displays the accuracy and the loss plots on the training and validation datasets throughout training epochs, respectively. It is apparent that after 20 epochs, algorithms start converging in both the training and validation phases. Similarly, the loss begins to taper off towards the zero line, as shown in the later part of the figure

6. Validation Results of the CNN-VGG16 Model

The hybrid method used VGG16's pre-trained layers for deep hierarchical feature learning and CNN for initial feature extraction. 70% of the dataset was used to train the model, while the remaining 30% was used for testing. To increase robustness, data augmentation was used. According to the results, the hybrid model performed better than the solo CNN and VGG16 architectures, with 97.1% accuracy, 96.5% precision, 97.3% recall, and 96.9% F1-score.

Ten-fold cross-validation was used to guarantee the model's dependability. The dataset was split into ten equal subsets, and the model was iteratively evaluated on the remaining fold after being trained on nine folds. This method offered a generalized performance evaluation while lowering bias. The Table2 presents the 10-fold cross-validation results for the hybrid model.

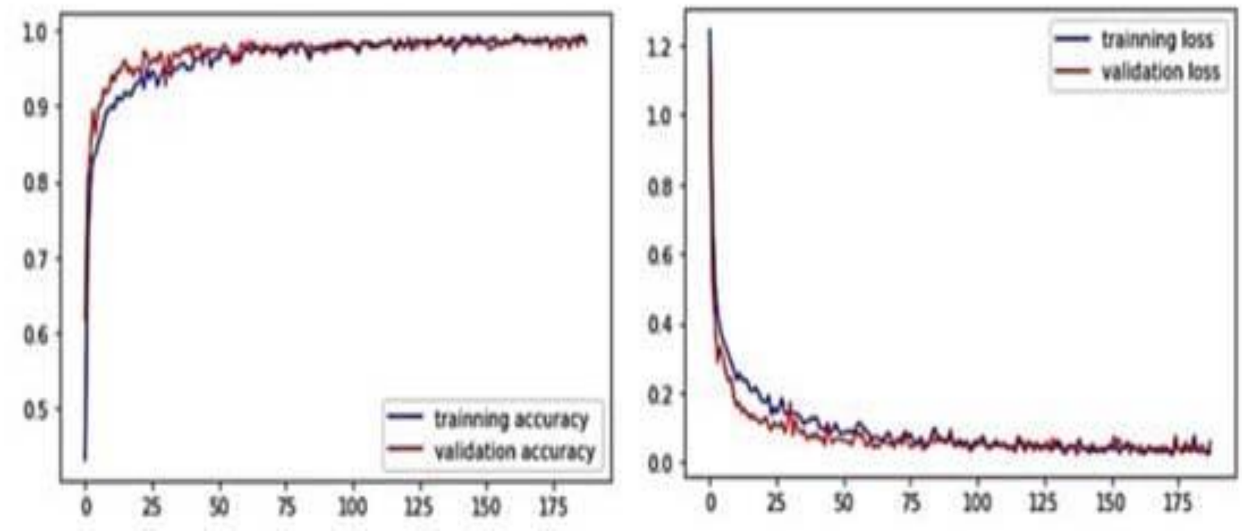

Figure 3: CNN training history plots

Table 2: Validation Results for Hybrid CNNVGG16 Model

Fold	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
1	96.8	96.1	97.2	96.6
2	97.0	96.3	97.5	96.9
3	97.2	96.6	97.1	96.8
4	96.9	96.4	97.4	96.9
5	97.3	96.8	97.2	97.0
6	96.7	96.2	97.3	96.7
7	97.4	96.9	97.5	97.2
8	97.1	96.5	97.2	96.8
9	97.0	96.4	97.3	96.8
10	97.2	96.7	97.1	96.9
Average	97.1	96.5	97.3	96.9

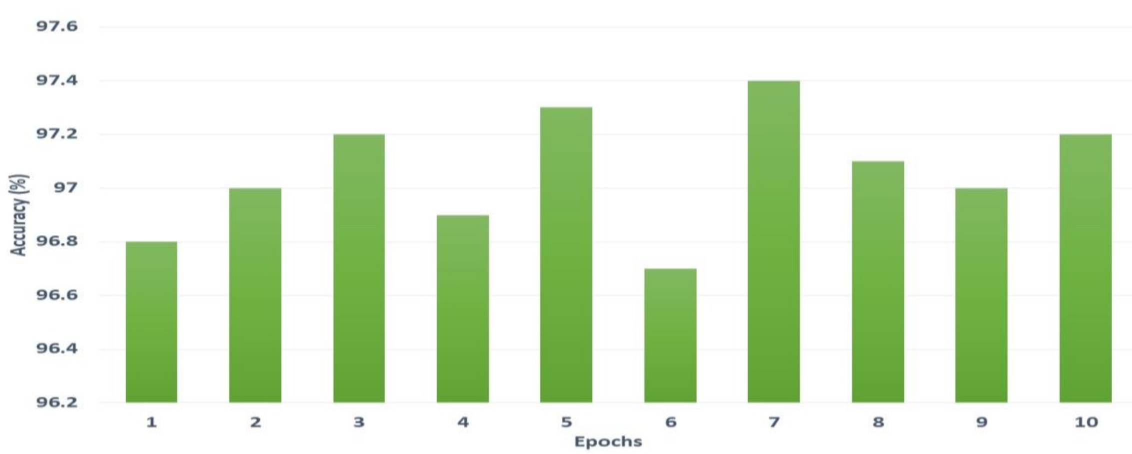

Figure 4: Accuracy Performance of the Hybrid Model

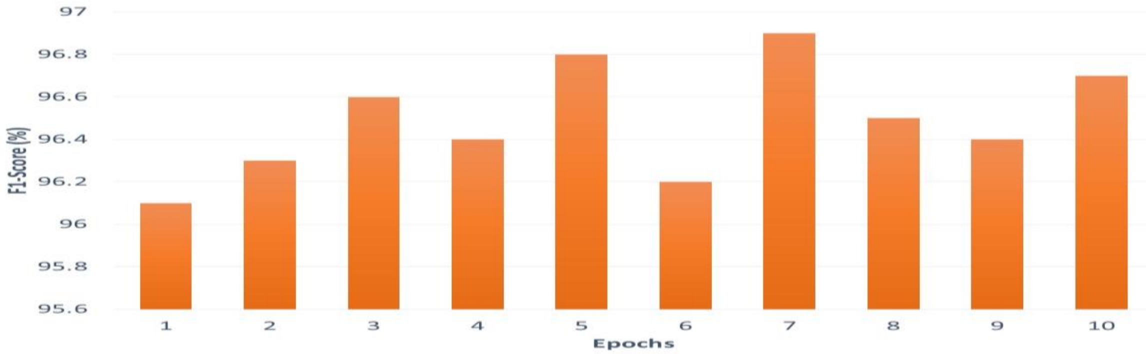

Figure 5: Precision Performance of the Hybrid Model

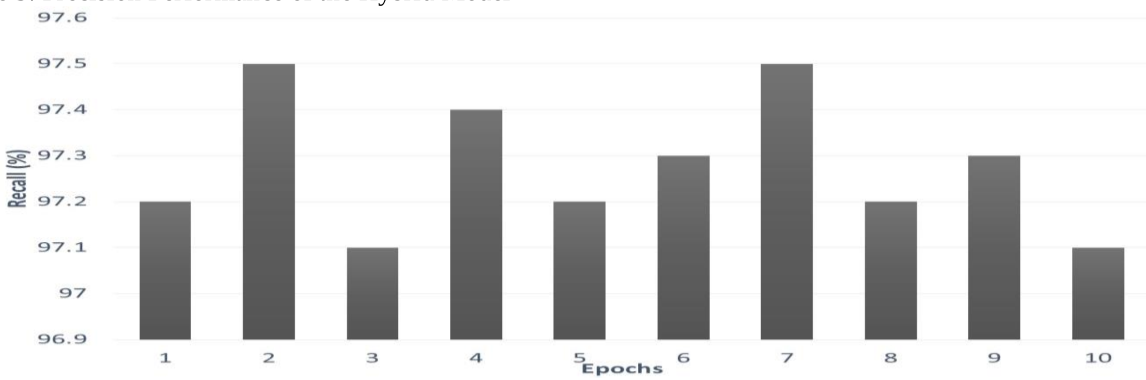

Figure 6: Recall Performance of the Hybrid Model

The result presented in Table 1 is further illustrated from Figures 4-7 where a diagrammatic representation of the system performance is reported as a chart. The chart presents how the hybrid model performed across each iteration of implementation to maintain a consistent performance result which validated the performance of the model.

In Figure 4, the performance accuracy of the hybrid model is reported, from the chart, it can be observed that the model achieved the highest accuracy of 97.4% on the 7th epoch, followed by the 5th, where it attained an accuracy of 97.0%. However, the first implementation of the model reported the poorest accuracy of 96.6%. Figure 4 reports the chart for the precision performance of the model. The precision result of the system is presented in Figure 5. In the figure, it can be seen that the highest precision was attained on the 7th epoch with the precision of 96.9%. The outcome of the recall validation is presented in Figure 6. The hybrid model performance in terms of recall has been presented in Figure 6, the recall attained 2 peak performances at the 2nd and 7th epoch with recall values of 97.5% each and 97.4% on the 4th epoch of the validation. The chart in Figure 6 reports the performance of the system in terms of F1-score.

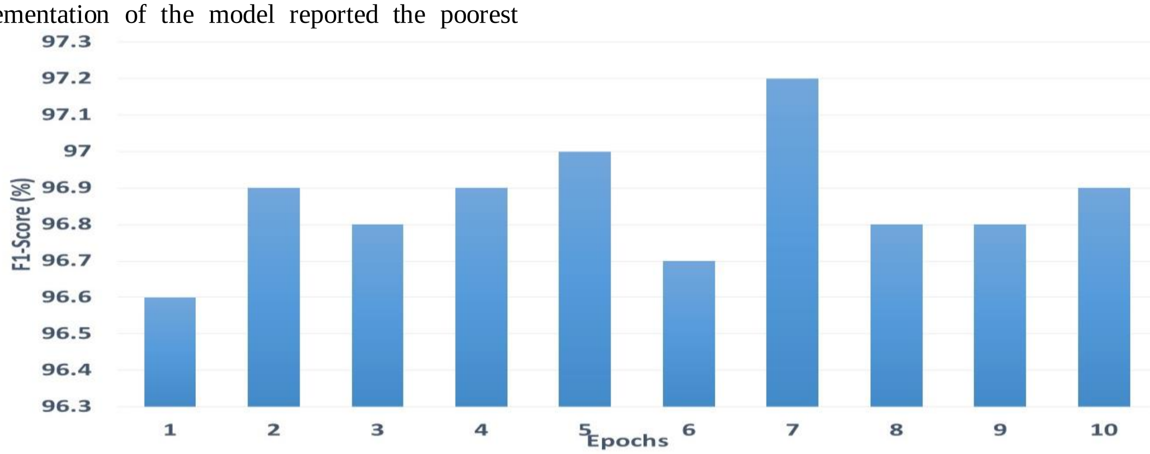

Figure 7: F1-Score Performance of the Hybrid Model

From the chart in Figure 7, the F-Score performance of the system reports that the hybrid model achieved the least score on the first epoch with a score of 96.6% and the best score on the 7th epoch with a score of 97.2%. The average accuracy of 97.1% confirms that the hybrid CNN-VGG16 model outperforms both standalone CNN (90.5%) and VGG16 (95.6%). Additionally, the precision (96.5%), recall (97.3%), and F1-score (96.9%) indicate a well-balanced model with minimal false positives and false negatives. The low variance across different folds further validates the model's stability and robustness. These findings demonstrate that the hybrid approach is highly effective for driver drowsiness detection, making it suitable for real-time applications in road safety monitoring systems.

6. CONCLUSION

This work used a hybrid CNN-VGG16 model to effectively design and assess a driver drowsiness detection system. A publicly accessible dataset of images classified according to the driver's eye conditions such as open, closed, yawning, and no yawning was used to train the algorithm. In order to identify tiredness, the suggested system uses deep learning algorithms to track face traits, especially eye states. The hybrid technique, which combined the advantages of VGG16 for transfer learning and CNN for feature extraction, performed better than the others, obtaining an F1-score of 96.9%, accuracy of 97.1%, precision of 96.5%, and recall of 97.3%. These outcomes confirm the model's efficacy in practical settings where identifying driver drowsiness is essential to averting traffic accidents. The model's generalizability and robustness were further validated by the 10-fold cross-validation, which showed no variance in performance between folds. The system improved its applicability in dynamic driving settings by handling changes in image quality and illumination through the use of data augmentation techniques and the potent features learnt from VGG16. To sum up, this work shows how deep learning-based systems for detecting driver tiredness may improve road safety and offers a dependable and effective way to incorporate it into Advanced Driver-Assistance Systems (ADAS). To further increase detection accuracy, future research can concentrate on refining the model for real-time deployment and investigating further characteristics like head posture or body motions.

7. REFERENCES

Ahmed, M. I. B., Alabdulkarem, H., Alomair, F., Aldossary, D., Alahmari, M., Alhumaidan, M., Alrassan, S., Rahman, A., Youldash, M., Zaman, G. (2023). A deep-learning approach to driver drowsiness detection. Safety, 9(3), 65. https://doi.org/10.3390/safety9030065

Alvarez, J. M., et al. (2016). Real-time drowsiness detection system for improving road safety. IEEE Transactions on Intelligent Transportation Systems, 17(6), 1633-1642.

Arefnezhad, S., Eichberger, A., Frühwirth, M., Kaufmann, C., Moser, M., Koglbauer, I. V. (2022). Driver monitoring of automated vehicles by classification of driver drowsiness using a deep convolutional neural network trained by scalograms of ECG signals. Energies, 15, 480. https://doi.org/10.3390/en15020480

Azam, K., Shakoor, A., Shah, R. A., Khan, A., Shah, S. A., Khalil, M. S. (2014). Comparison of fatigue-related road traffic crashes on the national highways and motorways in Pakistan. Journal of Engineering and Applied Sciences, 33, 47–54.

Cardone, D., Perpetuini, D., Filippini, C., Mancini, L., Nocco, S., Tritto, M., Rinella, S., Giacobbe, A., Fallica, G., Ricci, F., et al. (2022). Classification of drivers' mental workload levels: Comparison of machine learning methods based on ECG and infrared thermal signals. Sensors, 22, 7300. https://doi.org/10.3390/s22073000

CNN. Introduction to Pooling Layer—GeeksforGeeks. (2022). Available online: https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/ (accessed on 16 April 2022).

Ebrahimian, S., Nahvi, A., Tashakori, M., Salmanzadeh, H., Mohseni, O., & Leppänen, T. (2022). Multi-level classification of driver drowsiness by simultaneous analysis of ECG and respiration signals using deep neural networks. International Journal of Environmental Research and Public Health, 19, 10736. https://doi.org/10.3390/ijerph191710736

Google Colab: Welcome to Colaboratory—Colaboratory. Available online: https://colab.research.google.com/?utm_source=scs_index (accessed on 20 April 2023).

Huang, J., Luo, X., & Peng, X. (2020). A novel classification method for a driver's cognitive stress level by transferring interbeat intervals of the ECG signal to images. Sensors, 20, 1340. https://doi.org/10.3390/s20051340

ImageNet. Available online: https://image-net.org (accessed on 1 May 2023).

Introduction to Convolution Neural Network—GeeksforGeeks. (2022). Available online: https://www.geeksforgeeks.org/introduction-convolution-neural-network/ (accessed on 16 April 2022).

Khan, M. A. A., Alsawwaf, M., Arab, B., Alhashim, M., Almashharawi, F., Hakami, O., Olatunji, S. O., Farooqui, M., Rahman, A. (2022). Road damage detection and classification using deep learning and UAVs. In Proceedings of the Asian Conference on Innovation in Technology (ASIANCON) (pp. 1–6), Ravet, India, 26–28 August 2022.

Kumar, S., Devi, P., Singh, M., & Rani, M. (2024). Driver drowsiness detection system. International Journal of Science and Research Archive, 12(01), 1017–1022. https://doi.org/10.30574/ijsra.2024.12.1.0935

Marouf, M., Saranovac, L., &Vukomanovic, G. (2017). Algorithm for EMG noise level approximation in ECG signals. Biomedical Signal Processing and Control, 34, 158–165.

Nagy, R. B., Vesselenyi, T., &Popentiu-Vladicescu, F. (2017). An analysis of electro-oculogram signals processing using an artificial neural network. In Proceedings of the 13th International Conference eLearning and Software for Education (Vol. III, pp. 560–567), Bucharest, Romania, 27–28 April 2017.

Nagy, R. B., Vesselényi, T., &Popentiu-Vladicescu, F. (2018). Results regarding an EOG-based assistive wireless robot control system with visual feedback. In Proceedings of the International Scientific Conference eLearning and Software for Education, Bucharest, Romania, 19–20 April 2018 (Vol. III, pp. 453–460).

Pelt, D. M., & Sethian, J. A. (2018). A mixed-scale dense convolutional neural network for image analysis. Proceedings of the National Academy of Sciences of the United States of America, 115, 254–259. https://doi.org/10.1073/pnas.1718431115

Perumandla, D. (2020). Drowsiness_Dataset, Kaggle. Available online: https://www.kaggle.com/datasets/dheerajperumandla/drowsiness-dataset (accessed on 10 February 2023).

Precious, O. C., Kingsley, C. I., Chukwuemeka, E. E., & Ebere, U. C. (2022). Development of a real-time drowsy driver detection system. International Journal of Research and Innovation in Applied Science, VII(I), 1-9. ISSN 2454-6194

PyTorch Conv2D Explained with Examples—MLK—Machine Learning Knowledge. (2022). Available online: https://machinelearningknowledge.ai/pytorch-conv2d-explained-with-examples (accessed on 16 April 2022).

Tavakoli Kashani, A., Rakhshani Moghadam, M., &Amirifar, S. (2022). Factors affecting driver injury severity in fatigue and drowsiness accidents: A data mining framework. Journal of Injury and Violence Research, 14, 75–88.

Tensorflow.js tf.layers.flatten() function—GeeksforGeeks. (2022). Available online: https://www.geeksforgeeks.org/tensorflow-js-tf-layers-flatten-function/ (accessed on 16 April 2022).

Titare, S., Chinchghare, S., & Hande, K. N. (2021). Driver drowsiness detection and alert system. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 7(3), 583-588. https://doi.org/10.32628/CSEIT2173171

Vesselenyi, T., Dzitac, I., Dzitac, S., Hora, C., & Porumb, C. (2009). Preliminary issues on brain-machine contextual communication structure development. In Proceedings of the IEEE Conference, 3rd International Workshop on Soft Computing Applications (pp. 35–40), Szeged, Hungary, 29 July–1 August 2009.

Vesselenyi, T., Moca, S., Rus, A., Mitran, T., & Tataru, B. (2017). Driver drowsiness detection using ANN image processing. Proceedings of the 11th Edition of The International Congress of Automotive and Transport Engineering, Pitesti, Romania. IOP Conference Series: Materials Science and Engineering, 252, 012097. https://doi.org/10.1088/1757-899X/252/1/012097

Vesselenyi, T., Rus, A., Mitran, T., Tataru, B., & Moldovan, O. (2017). Vehicle driver drowsiness monitoring and warning system. In Proceedings of the 13th Edition of the International Congress of Automotive and Transport Engineering, CONAT 2016 (pp. 873–880), Brasov, Romania, 26–29 October 2016. Springer International Publishing