Investigation on Driver Drowsiness Detection using Deep Learning Approaches

Dakshnakumar G S
Computer Science and Engineering Karunya Institute of Technology and Sciences
Coimbatore, India
dakshnakumar@karunya.edu.in

J. Anitha
Computer Science and Engineering Karunya Institute of Technology and Sciences
Coimbatore, India
anitha_j@karunya.edu

***Abstract*— Driver sleepiness is a major factor in road accidents, necessitating effective detection methods to enhance road safety. Before the emergence of deep learning models, conventional approaches rely on manual observation to identify drowsiness symptoms in drivers. However, these methods are subjective, time-consuming, and prone to errors. This work aims to address the limitations of previous methods and provide a more accurate and efficient solution. To achieve this, a comparative analysis has been conducted using three popular CNN architectures: VGG19, EfficientNetB7, and MobileNetV2. These models are trained and evaluated on a carefully annotated dataset of driver eye images, classifying them as opened or closed. The results demonstrate the effectiveness of deep learning models in accurately detecting driver drowsiness. Among the evaluated architectures, EfficientNetB7 and MobileNetV2 achieve outstanding accuracy rates of 99.87% and 99% respectively, surpassing the traditional approaches. This paper contributes field of driver safety by demonstrating the potential of deep learning models in reducing the risks associated with drowsy driving.**

Keywords—Driver Drowsiness Detection, Computer Vision, Deep Learning, Convolutional Neural Networks, Transfer Learning, VGG19, EfficientNetB7, MobileNetV2

I. INTRODUCTION

Driver drowsiness is a significant issue that poses a serious threat to road safety. Fatigue or drowsiness can impair a driver's ability to react quickly and make sound decisions while operating a vehicle. This can lead to accidents, injuries, and even fatalities. According to statistics, drowsy driving contributes to a significant number of accidents worldwide. Therefore, the development of effective driver drowsiness detection system is crucial in mitigating this risk and ensuring safer roads. Several driver drowsiness detection systems [1] have been developed to address the problem of drowsy driving. These systems employ various techniques such as eye tracking, facial expression analysis, and physiological signal monitoring to detect signs of driver drowsiness. However, many existing systems have limitations in terms of accuracy, real-time performance, and usability. Some systems require specialized hardware or invasive sensors, making them less practical for widespread adoption. Others may struggle with accurately distinguishing between normal driver behavior and actual drowsiness, leading to false alarms or missed detections. The system's contributions lie in its accurate facial landmark tracking, precise ear calculation, and real-time monitoring capabilities.

This work aims to analyze and compare the performance of VGG19, EfficientNetB7, and MobileNetV2 deep learning models for driver drowsiness detection. The objective is to assess the accuracy and effectiveness of these models in classifying driver-eye states as opened or closed, while also exploring the benefits of transfer learning. The findings of this study will provide insights into the suitability of these models for real-world deployment and offer recommendations for the development of practical driver drowsiness detection systems.

II. RELATED WORKS

Driver drowsiness detection has been extensively studied in the literature, driven by the need to enhance road safety and mitigate the risks associated with fatigued drivers. Various approaches have been explored, ranging from traditional machine learning algorithms to more recent advancements in deep learning techniques [2, 3]. This section provides a comprehensive review of the existing literature on driver drowsiness detection, discusses previous studies that have leveraged deep learning models, and highlights the advantages and limitations of those models.

The literature on driver drowsiness detection reveals a diverse range of methodologies and techniques. Traditional approaches often rely on handcrafted features extracted from facial expressions, eye movements, and head poses. However, this method is limited in its ability to capture complex patterns and adapt to different situations. With the advent of deep learning, researchers have turned to convolutional neural networks (CNN) for their incredible ability to automatically learn features from raw data.

Deep learning-based driver drowsiness detection has shown remarkable advancements in recent years. CNNs have been extensively employed to analyze facial features and recognize drowsiness-related cues, such as eye closure, yawning, and head movements. Studies have demonstrated the effectiveness of CNNs in detecting drowsiness with high accuracy and robustness. In addition, using large-scale databases such as Facial Action Coding System (FACS) [4] and Drowsiness Driver Detection (DDD) datasets has facilitated the training of deep learning models.

Transfer learning, [5] a popular technique in deep learning, has been widely applied in driver drowsiness detection. Pre-trained models, such as VGG19, EfficientNetB7, and MobileNetV2, trained on large-scale image datasets (e.g., ImageNet), are fine-tuned on drowsiness-specific datasets. This approach leverages the learned features from the pre-trained models and adapts them to the specific task, allowing for improved performance and reduced training time. Transfer learning has proven effective in handling the limited availability of labeled drowsiness datasets, enabling the development of accurate and robust models.

VGG19 [6], known for its deep architecture, VGG19 has been widely used in computer vision tasks such as imaging and object detection. Its architecture, comprising multiple convolutional layers with small-sized filters (3x3), enables the extraction of rich spatial information. VGG19 has demonstrated exceptional performance in various domains;

979-8-3503-3324-4/23/$31.00 ©2023 IEEE

however, its main drawback lies in its large model size, leading to increased memory and computational requirements during training and inference.

EfficientNetB7 [7] has gained significant attention for its state-of-the-art accuracy and scalability. It employs a compound scaling method that balances model depth, width, and resolution to achieve high efficiency and effectiveness. The advantage of EfficientNetB7 lies in its ability to achieve superior accuracy with fewer parameters, making it well-suited for deployment in resource-constrained environments. However, training and fine-tuning EfficientNetB7 models may require more computational resources due to their larger size.

MobileNetV2 [8, 9] stands out for its lightweight design and efficient inference. This model significantly reduces computational complexity by utilizing depth-wise separable convolutions, while maintaining satisfactory performance. MobileNetV2 strikes a balance between accuracy and model size, making it suitable for real-time applications on devices with limited computational capabilities. However, its simplified architecture may not capture fine-grained details as effectively as deeper models like VGG19 and EfficientNetB7.

The advantages and limitations of these models play a crucial role in determining their applicability in driver drowsiness detection systems. VGG19 excels in capturing intricate details but may demand more computational resources. EfficientNetB7 offers excellent accuracy while requiring careful resource allocation. MobileNetV2 provides efficient inference but sacrifices some level of fine-grained.

III. Methodology

Various deep learning approaches are used in this work to detect driver drowsiness and to analyze the performance of the deep learning models in the identification of driver drowsiness problems. As shown in Fig. 1, a systematic approach is used to develop and evaluate deep learning models for driver drowsiness detection. Firstly, a dataset consisting of 4,000 images representing opened and closed eye states has been obtained from Kaggle. The dataset is carefully curated to capture diverse variations in eye states, including different individuals, lighting conditions, and head poses. This dataset is then processed by standardizing image dimensions, normalizing pixel values, and dividing it into training and test sets in a ratio of 70:30. Three deep learning models, namely VGG19, EfficientNetB7, and MobileNetV2, have been selected for the driver drowsiness detection task. Each model is chosen based on its proven performance in image classification and unique architectural features. Model configurations are tailored specifically for the driver drowsiness detection task. Additional layers such as global average pooling, dropout, and dense layers are added to EfficientNetB7 and MobileNetV2 to adapt the models' outputs to the classification task. For VGG19, transfer learning is applied by fine-tuning the pre-trained model and adding dropout and dense layers. The models are trained using the prepared dataset with specified hyperparameters, including a learning rate 1e-4, 7 epochs, and a batch size 32.

An Adam optimizer [10] and the categorical cross-entropy loss function are used for training. Following the training process, the models are evaluated using the testing set to measure their accuracy in detecting driver drowsiness. The primary performance metric used is accuracy, which measures the proportion of correctly classified instances. The achieved accuracy values are compared among the three models, allowing for a comprehensive analysis of their effectiveness. VGG19 has achieved an accuracy of 96.87%, while EfficientNetB7 and MobileNetV2 have achieved higher accuracies of 99.87% and 99%, respectively. The comparative analysis reveals the strengths and weaknesses of each model.

EfficientNetB7 demonstrates superior accuracy with a more optimized architecture, while MobileNetV2 exhibits a lightweight structure and efficient inference capabilities. The findings and observations from the performance evaluation and comparative analysis contribute to understanding the suitability of deep learning models for driver drowsiness detection. The study provides insights into the strengths and limitations of different models, guiding the development of accurate and efficient driver safety systems. The research outcomes have important implications for practical applications, including selecting the most suitable model based on computational resources, accuracy requirements, and real-time performance constraints.

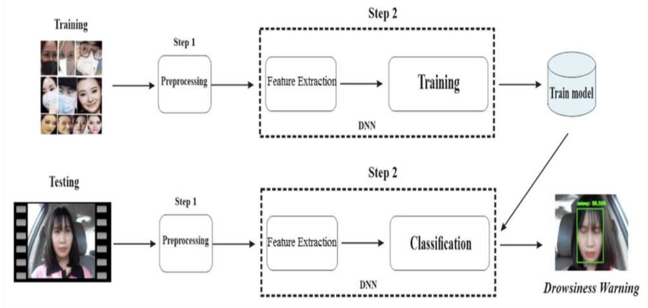

Fig. 1. Framework of drowsiness using deep learning model

A. Dataset Description and Preprocessing

This work utilizes a dataset sourced from Kaggle [14] for the training and evaluation of driver drowsiness detection models. The dataset consists of a total of 4,000 images, categorized into two classes: closed eye and opened eye. Fig. 2 depicts the sample images of the two categories. The closed eye class comprises 2,000 images, while the opened eye class contains an equal number of images, ensuring a balanced representation of both states. To prepare the dataset for model training, a series of pre-processing techniques are performed. Firstly, the image size is standardized to ensure uniformity across the dataset, resizing all images to a common resolution. This step eliminates any potential bias caused by variations in image dimensions and facilitates the compatibility of images with the selected deep learning models. Fig. 3 shows the pre-processed images of the two categories.

Subsequently, data augmentation techniques are applied to augment the dataset and enhance its diversity [11]. Techniques such as random rotations, translations, and flips are employed to introduce variations in pose, viewpoint, and illumination. Data augmentation aids in mitigating overfitting and improves the generalization capabilities of the trained models, enabling them to better handle real-world scenarios. In addition to data augmentation, normalization is done to standardize image pixel values. Normalization ensures that the input data has a consistent scale and distribution, facilitating effective convergence during model training. Finally, the data set is split into training and validation sets. The training set is used to train the deep learning model, while the validation set serves as an independent evaluation set to evaluate the model's performance and prevent overfitting. The dataset split is carefully designed to maintain a representative distribution of closed eye and opened eye images in both sets, preserving the balance of the classes. Employing these pre-processing techniques aims to enhance the quality and diversity of the dataset, enabling more accurate and robust training of the driver drowsiness detection models. The standardized resolution, data augmentation, normalization, and appropriate dataset splitting contribute to a comprehensive evaluation and reliable assessment of the selected models' performance.

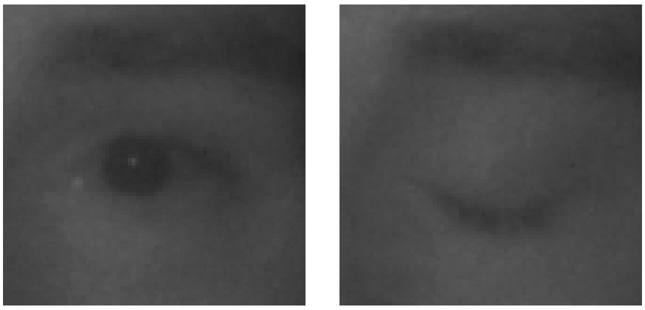

Fig. 2. Sample images of opened and closed eye state

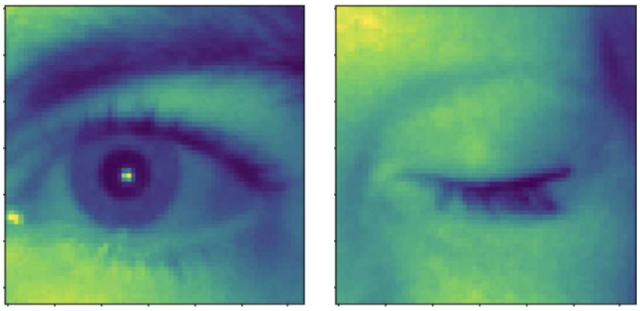

Fig. 3. Pre-processed images of opened and closed eye state

B. Deep Learning Models

In this study, three popular deep learning models are employed for driver drowsiness detection: VGG19, EfficientNetB7, and MobileNetV2. Each of these models possesses distinct architectural features and unique characteristics that contribute to their performance and suitability for the task at hand.

1) VGG19:

Fig. 4 illustrates the architectural design of the VGG19 model used in the study for driver drowsiness detection.

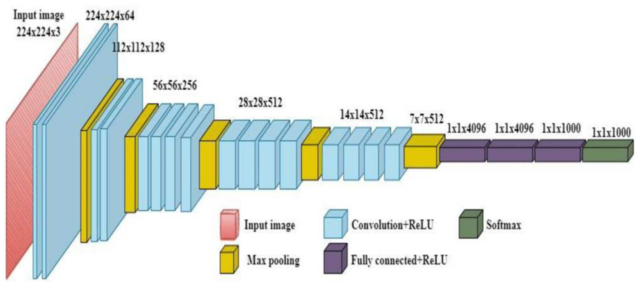

Fig. 4. Aricitecture of VGG19 deep learning model

VGG19 is a deep convolutional neural network (CNN) architecture widely used in various computer vision problems. It is characterized by its deep structure, consisting of 19 layers, including convolutional layers with small-sized filters (3x3) and max-pooling layers. VGG19's primary strength lies in its ability to capture intricate details and spatial features effectively. It can extract rich representations from input images by employing multiple convolutional layers. However, VGG19's main limitation is its large model size, which results in higher memory and computational requirements during both training and inference. This aspect needs to be considered, especially when deploying the model in resource-constrained environments.

2) EfficientNetB7:

Fig. 5 showcases the architectural design of the EfficientNetB7 model utilized in study for driver drowsiness detection.

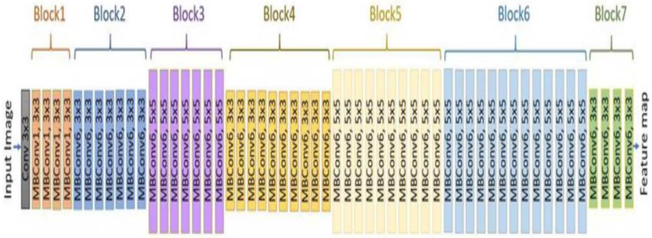

Fig. 5. Aricitecture of EficientNetB7 deep learning model

EfficientNetB7 is part of the EfficientNet family of models, known for their state-of-the-art performance and scalability. These models employ a compound scaling method that balances model depth, width, and resolution to achieve optimal efficiency and effectiveness. EfficientNetB7, specifically, has a larger size and higher complexity compared to earlier variants, resulting in enhanced accuracy. One of the main advantages of EfficientNetB7 is that it achieves high performance with fewer parameters, making it suitable for deployment in resource-limited environments. However, it is important to note that training and fine-tuning EfficientNetB7 models may require additional computational resources due to their larger size.

3) MobileNetV2:

Fig. 6 presents the architectural design of the MobileNetV2 model employed in the study for driver drowsiness detection.

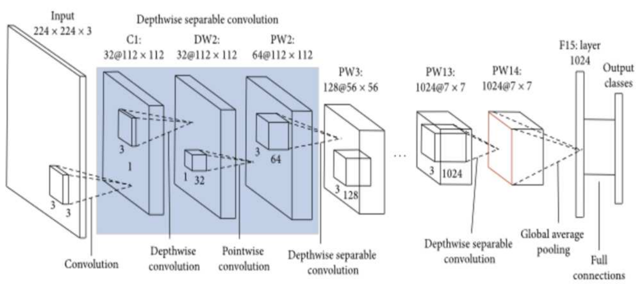

Fig. 6. Aricitecture of MobileNetV2 deep learning model

MobileNetV2 is a CNN architecture specifically designed to efficiently interact with mobile and embedded devices. This convention is achieved by using split-depth convolution, which divides the standard convolution process into parts of different depths and split points. This approach significantly reduces the computational complexity while maintaining reasonable performance. MobileNetV2 strikes a balance between accuracy and model size, making it particularly suitable for real-time applications on devices with limited computational capabilities. However, due to its simplified architecture, MobileNetV2 may not capture fine-grained details as effectively as deeper models like VGG19 and EfficientNetB7.

When comparing the strengths and weaknesses of these models for driver drowsiness detection, several factors come into play. VGG19 excels in capturing intricate details and spatial features, making it suitable for scenarios where fine-grained analysis is critical. However, its large model size may pose challenges in terms of memory and computational requirements. EfficientNetB7 offers superior accuracy due to its optimized scaling approach while utilizing fewer parameters. This makes it a compelling choice for achieving high performance in resource-constrained environments. However, the increased model size may require more computational resources during training and fine-tuning. MobileNetV2 stands out for its lightweight design and efficient inference, making it well-suited for real-time applications on devices with limited computational capabilities. While it may sacrifice some level of fine-grained feature representation, its efficiency and speed make it a valuable choice for deployment in real-world scenarios. Ultimately, the selection of the most appropriate model depends on the specific requirements and constraints of the driver drowsiness detection system. Considering factors such as computational resources, accuracy requirements, and real-time performance can guide the decision-making process and ensure the chosen model aligns with the desired objectives.

IV. Implementation

The experimental setup involves conducting the driver drowsiness detection model training and evaluation on a MacBook M1 Pro device. For the software setup, the training and evaluation procedures are performed using the Visual Studio Code (VSCode) integrated development environment (IDE).

A. *Training and Testing*

To train and evaluate the model, the data set is divided into a training set and a test set in the ratio of 70:30. The training set is used to train the model, while the testing set served as an independent dataset for evaluating the trained models' performance. For the EfficientNet configuration, the base EfficientNetB7 model is utilized. The output from the base model is fed into a global average pooling layer to aggregate spatial information. A dropout layer with a rate of 0.5 is applied to mitigate overfitting, followed by a dense layer with a softmax activation function to generate the final classification output. The EfficientNet configuration is implemented using TensorFlow and the Keras API.

In the VGG19 configuration, the pre-trained VGG19 model is utilized with the weights pre-initialized to the ImageNet weights. All layers of the VGG19 model, except for the last layer, are set as non-trainable. The output from the VGG19 model is extracted using the second-to-last layer. A dropout layer with a rate of 0.2 is added for address overfitting, followed by a dense layer with a softmax activation function for classification.

For the MobileNet configuration, the base MobileNetV2 model is used. The output from the base model is passed through average pooling to obtain spatial information. A flatten layer is applied to convert the output into a 1-dimensional vector. This is followed by a dense layer with a ReLU activation function and a dropout layer with a rate of 0.5 to prevent overfitting. The final dense layer with a sigmoid activation function produces the classification output.

For all three models, VGG19, EfficientNetB7, and MobileNetV2, the training process have been conducted using the following hyperparameters [12, 13]:

Learning Rate (INIT_LR): The initial learning rate is set to 1e-4. It determines the step size taken during model optimization and affects the speed and quality of convergence.

Number of Epochs (EPOCHS): The models are trained for 7 epochs. An epoch represents one complete pass through the entire training dataset during the training process.

Batch Size (BS): The batch size is set to 32. This specifies the number of samples used in each iteration of the training process. Larger sizes can speed up training but may require more memory.

B. *Hyperparameter Optimization*

The specified hyperparameters, INIT_LR, EPOCHS, and BS, are selected based on empirical observations and previous studies in the field. These values are fine-tuned to achieve a balance between model performance and computational efficiency. During the training process, the Adam optimizer has been employed with the specified learning rate. The categorical cross-entropy loss function is used to measure the discrepancy between predicted and actual class labels and guide the optimization process. The selected hyperparameters and optimization techniques ensure that the models are trained effectively and converged to their optimal performance within the given resource constraints. The experimental setup provides a robust foundation for training and evaluating the driver drowsiness detection models. By utilizing the specified hardware and software configurations, as well as implementing appropriate training and validation procedures with the defined hyperparameters, the models are trained effectively and evaluated accurately, enabling reliable analysis and comparison of their performance.

C. *Performance Evaluation*

The performance of driver drowsiness detection models is evaluated using accuracy, which measures the proportion of correctly classified instances in the testing set. Accuracy is a widely adopted metric for classification tasks and provides an overall assessment of the model's effectiveness in distinguishing between opened and closed eye states.

The performance of the VGG19, EfficientNetB7, and MobileNetV2 models is compared based on the achieved accuracy on the testing set. The results of the evaluation are as follows:

VGG19: The VGG19 model achieves 96.87% accuracy in the test set. This architecture, with its deep structure and ability to capture intricate details, demonstrated promising performance in detecting driver drowsiness. However, its larger model size may pose computational challenges in resource-constrained environments.

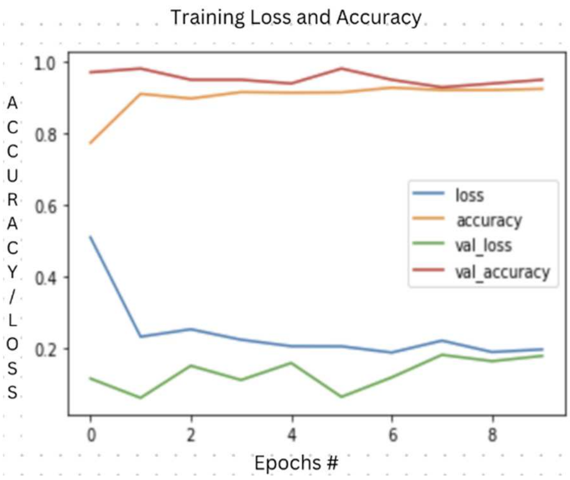

Fig. 7. Training and validation performance of VGG19

Fig. 7 provides insights into the learning progress and generalization capabilities of the VGG19 model for driver drowsiness detection.

EfficientNetB7: The EfficientNetB7 model achieves an accuracy of 99.87% in the test set. The EfficientNetB7's optimization approach, balancing model depth, width, and resolution, results in superior accuracy while utilizing fewer parameters. This model's performance indicates its potential as an effective solution for driver drowsiness detection tasks.

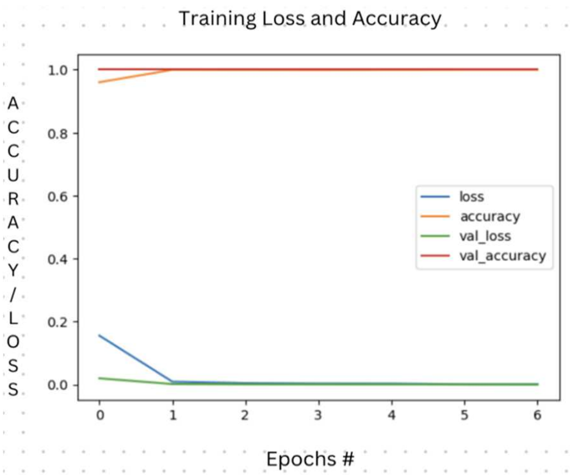

Fig. 8. Training and validation performance of EfficientNetB7

Fig. 8 offers valuable information about the EfficientNetB7 model's learning trajectory and its ability to generalize in the context of driver drowsiness detection.

MobileNetV2: The MobileNetV2 model achieves an accuracy of 99% on the testing set. MobileNetV2's lightweight architecture and efficient inference capabilities make it a suitable choice for real-time applications on devices with limited computational resources. Although it may sacrifice some level of fine-grained feature representation, it still achieves commendable accuracy in detecting driver drowsiness.

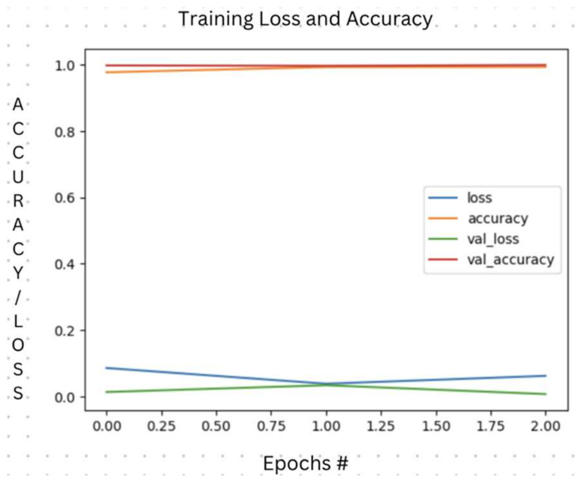

Fig. 9. Training and validation performance of MobileNetV2

The depicted Fig. 9 provides valuable insights into the MobileNetV2 model's learning progression and its capacity for generalization when applied to the detection of driver drowsiness.

Table I shows the comparative analysis of implemented deep learning models. The comparative analysis reveals that all three models, VGG19, EfficientNetB7, and MobileNetV2, demonstrate strong performance in driver drowsiness detection. EfficientNetB7 achieves the highest accuracy of 99.87%, closely followed by MobileNetV2 with an accuracy of 99%. VGG19 achieves a slightly lower accuracy of 96.87%. The results suggest that deeper models like VGG19 and EfficientNetB7, with their ability to capture intricate details, perform well in driver drowsiness detection. However, EfficientNetB7 outperforms VGG19 by utilizing a more optimized architecture, achieving higher accuracy with fewer parameters. MobileNetV2, on the other hand, shows promising results with its lightweight architecture and efficient inference. Despite its simplified structure, it achieves a commendable accuracy of 99%, demonstrating its suitability for real-time applications on devices with limited computational capabilities.

These findings highlight the importance of selecting an appropriate model based on the specific requirements of the driver drowsiness detection system, such as computational resources, accuracy thresholds, and real-time performance constraints. Overall, the experimental results showcase the potential of deep learning models in accurately detecting driver drowsiness, providing valuable insights for the development of effective driver safety systems.

TABLE I. PEFORMANCE ANALYSIS OF THE DEEP LEARNING MODELS

Model	Precision	Recall	F1-Score	Accuracy
VGG19	97	9	95.2	96.52
MobileNetV2	99.2	97	98.5	98
EfficientNetB7	99	98.3	98	99

V. Conclusion

In this work, the performance of three deep learning models, VGG19, EfficientNetB7, and MobileNetV2 are investigated for driver sleepiness detection based on eye images. Findings reveal that all three models achieve high accuracy, with EfficientNetB7 and MobileNetV2 outperforming VGG19. These results demonstrate the effectiveness of deep learning models in accurately classifying driver drowsiness states. While this study showed promising results, there are several avenues for future research. First, expanding the dataset to include a wider range of driving conditions and driver demographics would enhance the models' robustness and generalizability. Additionally, integrating other modalities such as head pose, facial expressions, and physiological signals could further improve the accuracy and reliability of drowsiness detection systems. It is crucial to emphasize the importance of selecting appropriate models for driver drowsiness detection. The comparative analysis highlights the strengths and weaknesses of the VGG19, EfficientNetB7, and MobileNetV2 models. The choice of model should consider factors such as computational complexity, real-time performance, and scalability to ensure practical feasibility in deployment scenarios.

Driver drowsiness detection holds significant importance in ensuring road safety. By employing accurate and reliable models and developing effective systems to detect drowsiness in drivers and mitigate potential accidents. Future research in this field has the potential to refine and optimize existing models, leading to the development of robust and efficient driver drowsiness detection systems. In conclusion, this study highlights the potential of deep learning models for driver drowsiness detection. Through further research and advancements, can contribute to the development of innovative solutions that enhance road safety and save lives.

References

1. Chieh, T.C., Mustafa, M.M., Hussain, A., Hendi, S.F., and Majlis, B.Y.: 'Development of vehicle driver drowsiness detection system using electrooculogram (EOG)', in Editor (Ed.)^(Eds.): 'Book Development of vehicle driver drowsiness detection system using electrooculogram (EOG)' (IEEE, 2005, edn.), pp. 165-168
2. Kamti, M.K., and Iqbal, R.J.T.r.r.: 'Evolution of Driver Fatigue Detection Techniques—A Review From 2007 to 2021', 2022, 2676, (12), pp. 485-507
3. Magán, E., Sesmero, M.P., Alonso-Weber, J.M., and Sanchis, A.J.A.S.: 'Driver drowsiness detection by applying deep learning techniques to sequences of images', 2022, 12, (3), pp. 1145
4. Clark, E.A., Kessinger, J.N., Duncan, S.E., Bell, M.A., Lahne, J., Gallagher, D.L., and O'Keefe, S.F.J.F.i.p.: 'The facial action coding system for characterization of human affective response to consumer product-based stimuli: a systematic review', 2020, 11, pp. 920
5. Deshpande, A.V.J.A.a.S.: 'Transfer Learning-Based Drowsiness Detection System for Driver Assistance'
6. Hashemi, M., Mirrashid, A., and Beheshti Shirazi, A.J.S.C.S.: 'Driver safety development: Real-time driver drowsiness detection system based on convolutional neural network', 2020, 1, pp. 1-10
7. Albasrawi, R., Fadhil, F.F., Ghazal, M.T.J.B.o.E.E., and Informatics: 'Driver drowsiness monitoring system based on facial Landmark detection with convolutional neural network for prediction', 2022, 11, (5), pp. 2637-2644
8. Phan, A.-C., Nguyen, N.-H.-Q., Trieu, T.-N., and Phan, T.-C.J.A.S.: 'An efficient approach for detecting driver drowsiness based on deep learning', 2021, 11, (18), pp. 8441
9. Bharti, S.S., Mahajan, R., Bhoriya, S., and Bhat, A.: 'Driver's Drowsiness Detection using MobileNet on Web', in Editor (Ed.)^(Eds.): 'Book Driver's Drowsiness Detection using MobileNet on Web' (IEEE, 2023, edn.), pp. 1229-1233
10. Kingma, D.P., and Ba, J.J.a.p.a.: 'Adam: A method for stochastic optimization', 2014
11. Viswanadhapalli, Y., Srinivas, C.V., Basha, G., Dasari, H.P., Langodan, S., Venkat Ratnam, M., and Hoteit, I.J.A.S.L.: 'A diagnostic study of extreme precipitation over Kerala during August 2018', 2019, 20, (12), pp. e941
12. Vo, H.-T., Hoang, T.N., Quach, L.-D.J.I.J.o.A.C.S., and Applications: 'An Approach to Hyperparameter Tuning in Transfer Learning for Driver Drowsiness Detection Based on Bayesian Optimization and Random Search', 2023, 14, (4)
13. Yang, Y., Li, C., Hou, X., Wang, C., and Zhang, W.J.A.A.: 'Nozzle atomizing performance evaluation in complex environment using residual atrous spatial pyramid network', 2022, 12, (3), pp. 035205
14. https://www.kaggle.com/datasets/prasadvpatil/mrl-dataset (accessed on 29 June 2023).