Drowsiness detection in real-time via convolutional neural networks and transfer learning

Dina Salem[1]* and Mohamed Waleed[1]

*Correspondence:
Dena.salem@mail.com

[1] Department of Computer and Systems, Faculty of Engineering, MUST University, 6th of October City, Egypt

Abstract

Drowsiness detection is a critical aspect of ensuring safety in various domains, including transportation, online learning, and multimedia consumption. This research paper presents a comprehensive investigation into drowsiness detection methods, with a specific focus on utilizing convolutional neural networks (CNN) and transfer learning. Notably, the proposed study extends beyond theoretical exploration to practical application, as we have developed a user-friendly mobile application incorporating these advanced techniques. Diverse datasets are integrated to systematically evaluate the implemented model, and the results showcase its remarkable effectiveness. For both multi-class and binary classification scenarios, our drowsiness detection system achieves impressive accuracy rates ranging from 90 to 99.86%. This research not only contributes to the academic understanding of drowsiness detection but also highlights the successful implementation of such methodologies in real-world scenarios through the development of our application.

Keywords: Artificial intelligence, Convolutional neural network, Deep learning, Drowsiness detection, Transfer learning

Introduction

Drowsiness, characterized by reduced alertness and cognitive impairment, poses a formidable risk across various facets of human life, with potentially grave consequences. Its repercussions span diverse domains, including transportation, education, and entertainment, with the potential for life-threatening outcomes [1]. In the context of road safety, driver drowsiness stands as a leading cause of accidents, resulting in global injuries and fatalities [2]. The impact extends to education, where drowsiness can impede learning and productivity, particularly in the prevalent realm of online learning environments [3]. This paper delves into the proactive exploration of cutting-edge technologies, specifically convolutional neural networks (CNNs) and transfer learning, to address the intricate challenge of drowsiness detection. These methodologies, renowned for their success in image analysis tasks, present as suitable candidates for discerning subtle facial cues indicative of drowsiness. Through a meticulous analysis, we present the results derived from extensive experiments conducted on diverse datasets.

The research at hand works in two interleaved objectives. The first is to illuminate the effectiveness of the proposed approach, highlighting the accuracy rates across both multiclass and binary classification scenarios. The second is to extend the work beyond mere theoretical exploration and implementations to delve into the practical applications through a drowsiness detection system, emphasizing its potential to significantly support safety and engagement in critical domains. These objectives are to be satisfied by carrying excessive implementations to different learning models on a variety of datasets, some of which are specifically curated for the service of increasing the accuracy of the proposed model.

The rest of this paper is organized as follows; "Related work" section emphasizes the most important work done in the drowsiness detection area. "Datasets" section delves into the datasets utilized for training and evaluating the proposed drowsiness detection model. In "Methods" section, the methodology employed for developing the drowsiness detection models using transfer learning and convolutional neural networks is detailed. "Experimental results" section presents the experimental results, showcasing the performance of the proposed model on the selected datasets. Following this, "Results and discussion" section discusses the developed application, highlighting its usability, features, and potential real-world applications. Finally, in "Conclusions" section, the paper concludes by summarizing the key findings, discussing limitations, and suggesting avenues for future research in the field of drowsiness detection.

Related work

Drowsiness detection holds paramount significance in ensuring road safety, prompting substantial research efforts over the years due to its critical role in road crash fatalities and injuries. In recent years, considerable progress has been achieved in the area of research and technology to protect the lives of drivers, especially with the use of image-based technologies.

One method used convolutional neural networks (CNN) for driver fatigue detection and integrating emotion analysis through 2D-CNN trained on driver facial patterns. This study by H. V. Chand and J. Karthikeyan [4] presents a novel multilevel distribution model for driver fatigue detection using CNN followed by Emotion analysis. Emotion analysis analyzes a driver's mental state and identifies motivating factors for different driving patterns. These driving patterns were analyzed using the acceleration system, vehicle speed, revolutions per minute (RPM), and driver facial recognition. The driver's facial patterns are processed with a 2D convolutional neural network (CNN) to recognize the driver's actions and emotions. The proposed model is implemented using OpenCV. The experimental results prove that the proposed models are well aligned with the trained data and the error rate comparing the trained and test data and reducing with minimal marginal difference. The experimental analysis and comparative statements generate an accuracy level of 93% in detecting both the behavior and emotion of the driver.

Another study by R. S. Duggal [5] reviewed the research on developing technologies that are intended to identify and forecast driver drowsiness while focusing on what occurs while driving. This study then introduced a system that predicts driver drowsiness by analyzing behaviors such as frequent yawning, eye closure, facial expressions,

voice, and steering patterns using a vehicle-mounted camera, aiming to alert drivers if distracted or exhibiting irregular steering patterns indicative of fatigue.

The research utilized a customized InceptionV3 model, pretrained with Imagenet weights, for binomial classification to discern whether subjects' eyes were open or closed. The training was conducted on "MRL Eye Dataset," The output of the input pipeline was fed into the customized InceptionV3 model. The first model achieved an accuracy of 94.78% on the training data but dropped to 82.32% on the test data, indicating an overfitting of 12.46%. Similarly, the second model attained a categorical accuracy of 94.98% on the training data, but on the test data, it decreased to 81.92%.

Moreover, a group of researchers Younes Ed-Doughmi, Najlae Idrissi, and Youssef Hbali [6], in their study, propose a novel approach for analyzing and predicting driver drowsiness by utilizing a curated dataset to refine and validate their model, which incorporates a multi-layer architecture of 3D convolutional networks (CNNs) specifically designed for detecting driver drowsiness. Through training, the researchers achieved an impressive accuracy rate of approximately 92%, demonstrating the efficacy of their approach.

M. Gomaa, R. Mahmoud and. A. Sarhan [7] use a novel deep learning-based model for predicting driver drowsiness by combining CNN and long short-term memory (LSTM) networks, they achieved superior results compared to state-of-the-art methods, accurately predicting driver drowsiness from video footage captured during driving.

Majeed et al. [8], focused on classifying "Yawning" and "No Yawning" classes for drowsiness detection. Their proposed CNN model achieves an impressive average accuracy of 96.69%, tested on both the original and augmented datasets.

Many research teams used different types of information to help make driving safer while exploring ways to detect when drivers get tired. Some look at how the eyes and mouth move to spot fatigue based on their geometric features [9]. Others categorized data into three types: vehicular, physiological, and behavioral to survey the optimal method for detecting driver fatigue [10]. And there are those who used frontal facial images captured from real-time video recorded with a webcam. Frames are extracted, and Haar-Adaboost facial recognition is used to identify faces in the images along with a CNN model [11]. Table 1 summarizes some of the recent research carried out to help improve the detection of drivers' drowsiness.

The significance of integrating multiple techniques and diverse datasets to enhance the reliability of driver fatigue detection systems was the main idea behind the research at hand. Our study remarkably advances drowsiness detection using CNN and Transfer Learning across six datasets, including a newly merged dataset (YAWDD, CEW, Glasses) achieving impressive accuracy in multi-class and binary classification. We have also developed a user-friendly app for real-world driver drowsiness detection, enhancing safety with additional features.

Datasets

In the domain of drowsiness detection, the quality and diversity of datasets play a crucial role in the development and evaluation of effective models and systems. The selection of appropriate datasets is paramount in ensuring the accuracy and reliability of drowsiness detection systems. This research dug into a variety of datasets, each

Table 1 Summary of some recent research with the highest accuracy

Article name	Year	Dataset	Method	Accuracy
[8]	2023	YAWDD	Proposes a deep neural network architecture for drowsiness detection employing a convolutional neural network (CNN)	96.69%
[5]	2022	MRL	Haar-cascade technology was used for detecting open and closed eyes. Convolutional neural network has been used here for detecting driver drowsiness particularly eyes blink, open and closure rate, and the face position	94%
[7]	2022	MRL and NTHU-DDD	A novel deep learning-based model for predicting driver drowsiness using a combination of convolutional neural networks (CNN) and long short-term memory (LSTM)	98%
[4]	2022	DDD and CK+	Model of multi-level distribution of detecting the driver drowsiness using the convolution neural networks (CNN) followed by the emotion analysis.	93%
[9]	2021	iBUG 300w	EAR (eye aspect ratio)	98%
[6]	2020	NTHU-DDD	Convolution neural networks (CNN)	92%
[10]	2020	DROZY, ZJU, NTHU, and RLDD	Hybrid model uses AI-based multi-task cascaded convolutional neural networks (MTCNN) as a behavioral measure, and the galvanic skin response (GSR) sensor as a physiological measure	91%

Table 2 Details of utilized datasets

Dataset name	Classes	Images	Size
YAWD Dataset	4	2904	175.84 MB
Driver Drowsiness Dataset (DDD)	2	41,790	2.76 GB
Glasses Dataset	2	164	1.1 MB
MRL Eye Dataset	2	84,898	328.2 MB
CEW Dataset (Closed Eyes in the Wild)	1	1192	20 MB
NTHU-DDD	2	66,520	3.02 GB
OC-Dataset (Open and Closed Eyes Dataset)	2	4103	212 MB

offering unique characteristics and challenges, to assess the robustness and generalizability of the proposed models. Table 2 explains the details of the deployed datasets.

The YAWDD dataset, comprising four classes (closed, yawn, no-yawn, and open) sourced from videos recorded by in-car cameras, features drivers engaged in a range of activities such as talking, singing, remaining silent, and yawning. The variety of images played a crucial role in the training phase.

To further enrich used datasets, this research proposes a new custom dataset with four classes (closed, yawn, no-yawn, open), consolidating images from the CEW, YAWDD, and Glasses and no-glasses datasets. This amalgamation resulted in a dataset of 2771 images, encompassing a wide range of facial expressions and activities. This curated dataset has been made publicly available on Kaggle with the name "YAWDD, CEW, GLASSES".

For binary classification tasks (closed and open eyes), we utilized the OC-Dataset, which consists of 4103 images that were generated using the UnityEyes simulator. Additionally, the MRL dataset, a large-scale collection of human eye images, provided valuable insights. We selected 20,000 images for each class from the MRL dataset, encompassing infrared images captured in diverse lighting conditions and resolutions. This experiment allowed us to evaluate our model's performance across different dataset types and sizes.

In addressing drowsiness detection (drowsy and not drowsy), we turned to the Driver Drowsiness Dataset (DDD). This dataset comprises extracted and cropped faces from the Real-Life Drowsiness Dataset videos, using VLC software. A subset of 4179 images 10% of the dataset was used for training our model.

Lastly, the NTHU-DDD dataset, transformed from a real dataset, served as another valuable resource for our project. We experimented with 6000 images from this dataset to evaluate the adaptability of our model to different data sources. Together these datasets enabled us to create a comprehensive and diverse training environment for our eye-related classification tasks. The datasets are balanced, and each has been divided into 2 parts which are used separately for training (70% of the dataset) and testing (30% of the dataset).
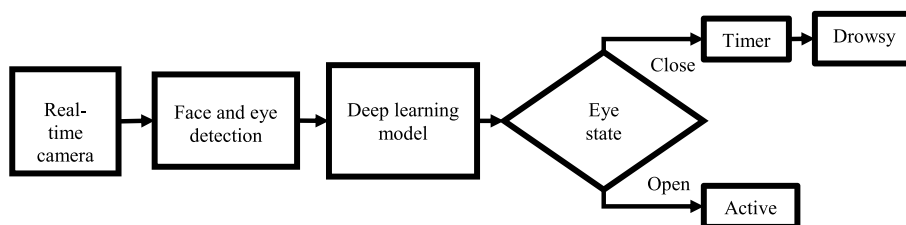
Methods

The proposed model works on the simple principle of a counter, where a time limit is set, and once it is exceeded, an audio/alarm signal is played when the user becomes sleepy.

A counter increases depending on how long the driver's eyes are closed. In our project, we want to compare the performance of a convolutional neural network (CNN) trained from scratch with CNN that leverages transfer learning (TL) to determine the most effective approach is more effective at detecting driver drowsiness. This comparative analysis provides valuable insight into the suitability of CNNs for this safety-critical application and helps make an informed choice between CNN-based and TL-based models. The driver drowsiness schema and the general flow of the proposed methodology are illustrated in Figs. 1 and 2 respectively.
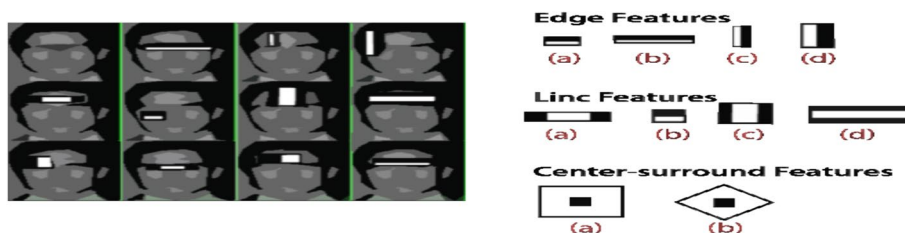
The Haar Cascade Algorithm

Feature extraction is a process to locate significant features or attributes of the data. Particularly when dealing with high-dimensional data. This technique would extract relevant data and bring out more accurate and concise descriptions. The dataset we are using contains a lot of complex information. Feature extraction is a step that

Fig. 1 Driver drowsiness schema
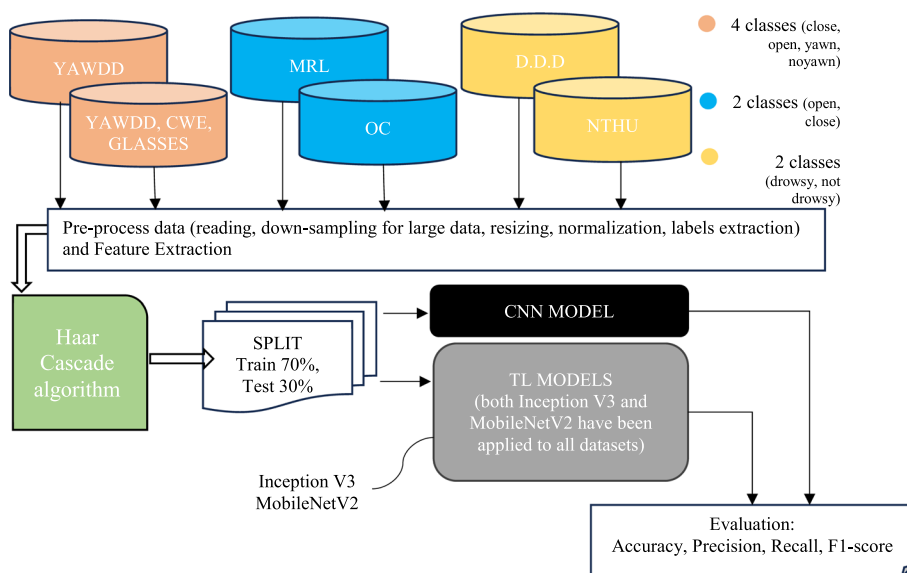
Fig. 2 Proposed methodology flowchart

Fig. 3 Illustration of the Haar Cascade Algorithm

helps to break complex information into much simpler forms, making it much easier to extract relevant information from the data. The features extracted could prove to be useful as it helps in mitigating the computational complexity that would otherwise have occurred. It also increases the accuracy of the model.

The Haar Cascade shown in Fig. 3 is a machine-learning object detection algorithm used for identifying objects or features within images or video frames. It is named after the Haar-like features used in the algorithm, which are simple rectangular filters. The Haar Cascade works by training positive and negative images of the object or feature to be detected. It creates a cascade of classifiers, where each classifier is trained to rule out negative instances quickly. The algorithm progressively applies these classifiers in a cascade fashion, with each stage reducing false positives. It is commonly used for face detection, eye detection, pedestrian detection, and various other object detection tasks [12]. In our project, Haar Cascade classifiers are employed to identify and locate the regions of interest (ROIs) (detect faces and eyes) in the video frames to detect whether the user is drowsy or not.
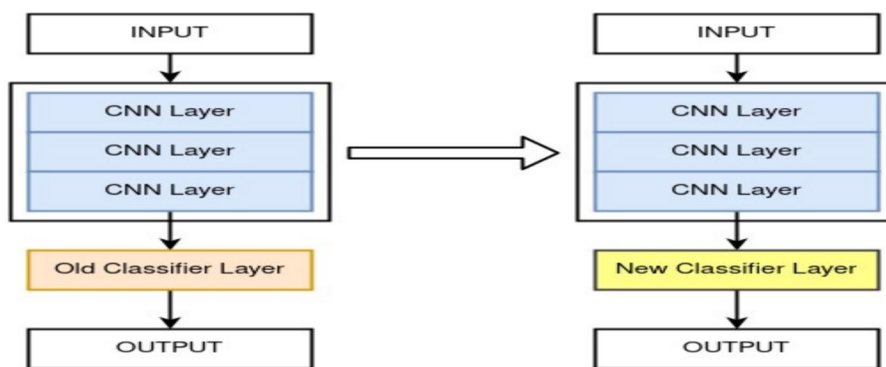
The transfer learning model

Transfer learning, using popular pre-trained models such as InceptionV3, MobilenetV2, was employed to exploit existing knowledge and adapt it to the drowsiness detection task. A general framework of the transfer learning model is shown in Fig. 4. In a transfer learning scenario with two models, both encompassing input layers, several convolutional neural network (CNN) layers, and output layers, the distinction lies in the classifier layer.

The first model, equipped with an "old" classifier layer, has been previously trained on a large dataset for a different task. This classifier layer has learned general features relevant to the original task. The second model, with a "new" classifier layer, is intended for a related but distinct task and has not been trained yet. However, this model inherits the same architecture as the first, including the same CNN layers and input/output structure. During transfer learning, the "old" classifier layer from the first model is detached, and the CNN layers, retaining their learned features, are connected to the "new" classifier layer. This new classifier layer is then trained or fine-tuned using a smaller dataset specific to the new task.

The key advantage lies in making the most of the learned representations from the initial task (via the pre-trained CNN layers) while adapting the final classifier to the new task. This process enables the efficient adaptation of knowledge gained from the original task to improve performance on the new task, especially when data for the new task is limited [13]. Different models are employed in the proposed methodology according to the classification task.

Four-class classification (close, open, yawn, no yawn)

For this classification task, the InceptionV3 [14] model is deployed. InceptionV3 a powerful convolutional neural network (CNN) architecture pre-trained on ImageNet. The model is tailored for image classification tasks with its deep and intricate structure. The last layer is adapted to output four classes corresponding to closed eyes, open eyes, yawning, and non-yawning expressions. A Global Average Pooling layer, dense layers, and dropout are added to fine-tune the model for the specific facial expression classification task.

Fig. 4 General framework for transfer learning model

Two-class classification (close, open)

Utilizing Haar Cascade for facial feature extraction and the InceptionV3 TL model, this binary classification model employs a final dense layer with sigmoid activation. Sigmoid activation is suitable for binary classification tasks, providing probability outputs for closed and open eyes. The InceptionV3 TL model enhances the learning of intricate features, making it adept at discerning between these foundational eye states.

Two-class classification (drowsy, not drowsy)

The binary classification task is facilitated by a final dense layer with sigmoid activation, enabling the model to predict the likelihood of drowsiness. MobileNetV2's [15] inherent ability to capture complex features in images, combined with transfer learning, enhances its performance in discerning between drowsy and non-drowsy states, providing valuable insights for safety-critical applications.

In all cases, the transfer learning technique is applied, utilizing the knowledge gained by the pre-trained models on large datasets. This allows the models to extract relevant features from facial images effectively. The inclusion of global pooling layers, dense layers, and dropout aids in refining the models for the specific nuances of each classification task, ensuring optimal performance in facial expression and drowsiness analysis.

Convolutional neural networks (CNNs)

A convolutional neural network (CNN) is a specialized deep learning architecture designed for processing structured grid-like data, such as images or sequences. It comprises layers that learn hierarchical representations by applying convolutions, pooling, and nonlinear activation functions. These networks excel in capturing spatial hierarchies and patterns within data, enabling tasks like image classification, object detection, and more [16]. Details of constructing the layers of CNN cordially with tuning its optimization parameters form our proposed model that is summarized in Fig. 5 and explained below.

Constructing the CNN layers

The basic architecture of a convolutional neural network (CNN) extracts its construction from the datasets. The main differences lie in the input dimensions, which are adapted based on the characteristics of the datasets and the tasks.
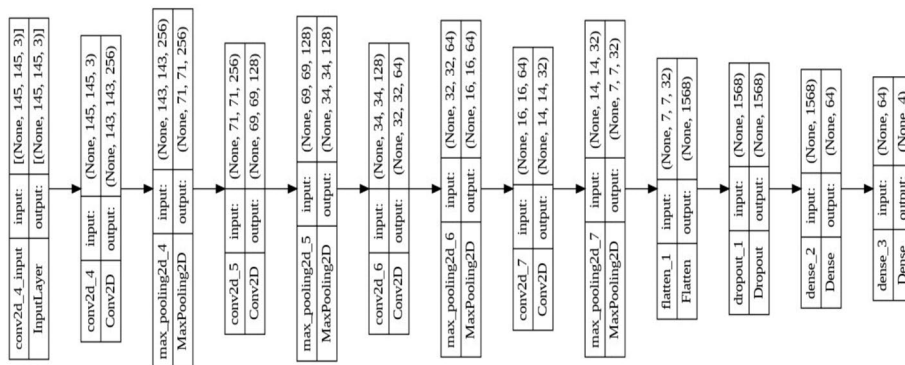
Fig. 5 CNN Architecture of the proposed model

All parts of the proposed module use convolutional layers to capture hierarchical features in the input images. Convolutional layers are designed to detect patterns and spatial hierarchies. In CNN, a pooling layer is a type of layer used to reduce the spatial dimensions (width and height) of the input volume (also known as the feature map) while preserving important information. This research proposed model uses Max-Pooling2D layers to down-sample the spatial dimensions of the feature maps, reducing computational complexity and extracting dominant features.

Then a Flatten Layer is used after convolutional and pooling layers. The importance of the Flatten layer is to transform the 2D feature maps into a 1D vector. This flattened vector is then fed into dense layers. Dense (fully connected) Layers are used for classification. The final dense layer has neurons equal to the number of classes in our specified task (2 or 4 based on the dataset used), with a softmax activation function for multi-class classification or sigmoid activation for binary classification.

One important layer in CNN is the Dropout layer which helps prevent overfitting by randomly setting a fraction of input units to 0 at each update during training. This introduces a form of redundancy, making the network more robust. In the proposed models a Dropout = 0.5 is employed with a 50% ratio of the input units.

Model optimization techniques

Batch normalization is employed in the proposed model to reduce the internal covariate shift by normalizing the layer's inputs, which in turn accelerates training and enables the utilization of higher learning rates. Conversely, data augmentation, while not a layer itself, served in our model as a vital technique for enhancing model generalization by artificially augmenting the diversity of the training dataset through the application of random transformations; rotation, zoom, and flip to input images. Additionally, activation functions play a crucial role in introducing non-linearity to the model, with ReLU commonly used in hidden layers to resolve the vanishing gradient problem. Meanwhile, softmax activation is typically applied in multiclass classification output layers, and sigmoid activation is favored for binary classification output layers. These techniques collectively contribute to improving the performance and robustness of neural network models.

Performance evaluation

Accuracy

Accuracy is a measure of the overall correctness of the model. It represents the ratio of correctly predicted instances to the total instances. A high accuracy score indicates that the model is making a high proportion of correct predictions. Accuracy = (True Positives + True Negatives) / Total no of instances.

Precision

Precision is a measure of how many of the predicted positive instances are actually positive. It focuses on minimizing false positives. High precision means that when the model predicts a positive result, it is likely to be correct. Precision = True Positives/Actual Results where Actual Results = True Positives + False Positives.

Recall

Recall is a measure of how many of the actual positive instances were successfully predicted by the model. It focuses on minimizing false negatives. High recall means that the model is good at capturing all positive instances in the dataset. Recall = True Positives/Predicted Results where Predicted Results = True Positives + False Negatives.

F1 score: The F1 score is the harmonic mean of precision and recall. It is particularly useful when you want to balance precision and recall, and you want a single metric that considers both. The F1 score is high when both precision and recall are high. F1 Score = 2*(Precision*Recall) / (Precision + Recall)

Experimental results

The evaluation of the proposed models' efficiency is based on four types of performance measurement metrics: accuracy, precision, recall, and F1-score. Merging the four listed performance metrics provides a comprehensive assessment of the models' effectiveness across different aspects of classification tasks. While accuracy alone provides a general overview of model performance, precision, recall, and F1-score offer insights into the model's ability to correctly identify positive and negative instances, as well as its balance between precision and recall. Figure 6 summarizes the values of the four metrics after applying the CNN and TL models to the six employed datasets. The details and analysis of each type of the implemented model are analyzed separately in the following section.

In our experiments, we start with the (YAWDD, CEW, GLASSES) dataset. As we delve into the YAWDD dataset, our model shows more accuracy in detection. Our experiments continue to the OC dataset, where the model can detect closed and open eyes, making our model capable of decoding various eye states as we noticed in YAWDD, and (YAWDD, CEW, GLASSES). Next, is the MRL dataset, with a large number of data entries. The NTHU dataset shows improvement in detecting drowsiness. Our journey ends with the Driver-Drowsiness dataset, where the model achieves the highest performance.

Results and discussion

Interesting insights into the performance of the three models—CNN, Inception V3, and MobileNet V2—can be gained by comparing their outcomes across various datasets. All three models show great accuracy, often above 95%, across most datasets. With a score of 99.94% on the Driver-Drowsiness dataset, the MobileNet V2 model has the best accuracy. However, the Inception V3 model often achieves somewhat higher recall and precision. While MobileNet V2 is quite flexible and performs well on various datasets, its recall may occasionally be a little bit lower. Despite the overall good performance of the CNN model, it might not be able to match Inception V3's levels of precision and recall.

Generally, specific needs like precision-recall balance, adaptability to various datasets, and computational efficiency may determine which of these models is best. Inception V3 typically offers slightly higher precision and recall, MobileNet V2 offers adaptability, and CNN exhibits balanced performance.
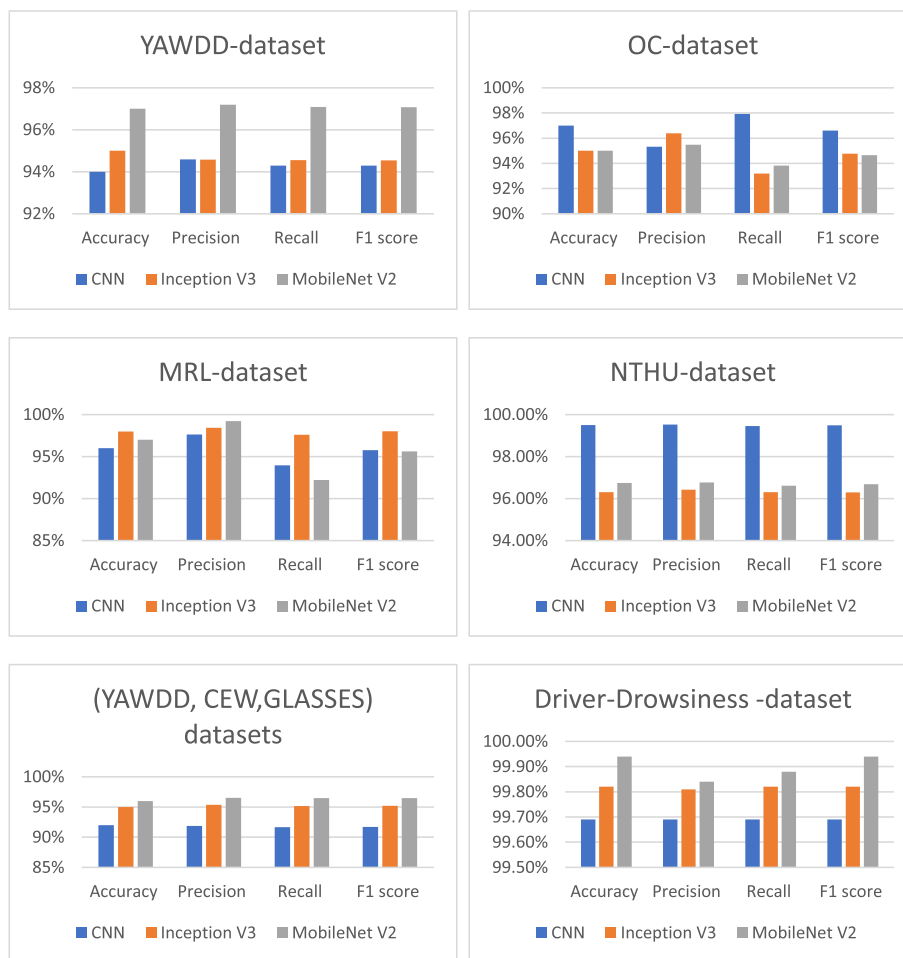
Fig. 6 Comparison between the performance of the three models on the six used datasets

Convolutional neural networks

The results indicate that the CNN model performs exceptionally well across various datasets, demonstrating high levels of accuracy, precision, recall, and F1 scores. Starting with the (YAWDD, CEW, GLASSES) dataset, the model achieves an accuracy of 92% with precision, recall, and F1 scores around 91–92%. Moving to the YAWDD dataset, there's a slight improvement in accuracy to 94% with higher precision, recall, and F1 scores around 94–95%. The UC dataset shows even better performance with an accuracy of 97% and precision, recall, and F1 scores above 95–97%. The MRL dataset maintains high performance across the board with an accuracy of 96% and precision, recall, and F1 scores around 94–98%. The NTHU dataset demonstrates exceptional performance with accuracy, precision, recall, and F1 scores all above 99%, showcasing the robustness of the model. Finally, on the Driver-Drowsiness dataset, the model achieves near-perfect scores across all metrics with accuracy, precision, recall, and F1 score of 99.69%. Overall, these results highlight the effectiveness and versatility of the CNN model in detecting various eye states and drowsiness across different datasets, with performance consistently high and sometimes near perfect.

Transfer learning

*InceptionV3 model*

The performance analysis of the Inception V3 model across various datasets reveals consistently high accuracy, precision, recall, and F1-scores. Beginning with the (YAWDD, CEW, GLASSES) datasets, the model achieves an accuracy of 95% with precision, recall, and F1-scores around 95–96%, indicating robust performance in detecting eye-related states. The YAWDD-dataset shows similar performance with an accuracy of 95% and precision, recall, and F1 scores around 94–95%. Transitioning to the OC-dataset, while accuracy remains at 95%, precision is notably higher at 96.39%, though recall slightly decreases to 93.19%, resulting in a slightly lower F1-score of 94.77%.

The MRL dataset demonstrates excellent performance with an accuracy of 98% and precision, recall, and F1-scores around 97–98%, indicating the model's effectiveness in handling diverse data. The NTHU-dataset maintains high performance with accuracy, precision, recall, and F1-scores all above 96%, highlighting the model's consistency across different datasets. Lastly, on the Driver-Drowsiness dataset, the Inception V3 model achieves outstanding performance with near-perfect scores across all metrics, showcasing its capability in accurately detecting drowsiness with accuracy, precision, recall, and F1-score of 99.82%. Overall, the analysis underscores the efficacy and reliability of the Inception V3 model across various datasets, with consistently high performance in detecting eye states and drowsiness (Table 3).

MobileNetV2 model

The performance analysis of the MobileNet V2 model across various datasets demonstrates consistently high accuracy, precision, recall, and F1-scores. Beginning with the (YAWDD, CEW, GLASSES) datasets, the model achieves an accuracy of 96% with precision, recall, and F1-scores around 96–97%, indicating robust performance in detecting eye-related states. The YAWDD-dataset exhibits even better performance with an accuracy of 97% and precision, recall, and F1-scores around 97–97.2%, showcasing the model's capability to handle variations in the data effectively. Transitioning to the OC-dataset, although the accuracy remains at 95%, precision is slightly higher at 95.49%, with a slight decrease in recall to 93.82%, resulting in a slightly lower F1 score of 94.65%.

The MRL dataset also demonstrates strong performance with an accuracy of 97% and exceptionally high precision of 99.24%, though recall drops to 92.23%, resulting in a

Table 3 Comparison between our results and other published work

Dataset	CNN (our work)	Inception V3 (our work)	MobileNet V2 (our work)	CNN (published work)	Inception V3 (published work)	MobileNet V2 (published work)
YAWDD, CEW, GLASSES	92%	95%	96%	–	–	–
YAWDD	94%	95%	97%	96.69% [8]	–	–
OC	97%	95%	95%	–	–	–
MRL	96%	98%	97%	94% [5]	–	–
NTHU	99.50%	96.31%	96.74%	92% [6]	–	–
Driver-Drowsiness	99.69%	99.82%	99.94%	–	–	–
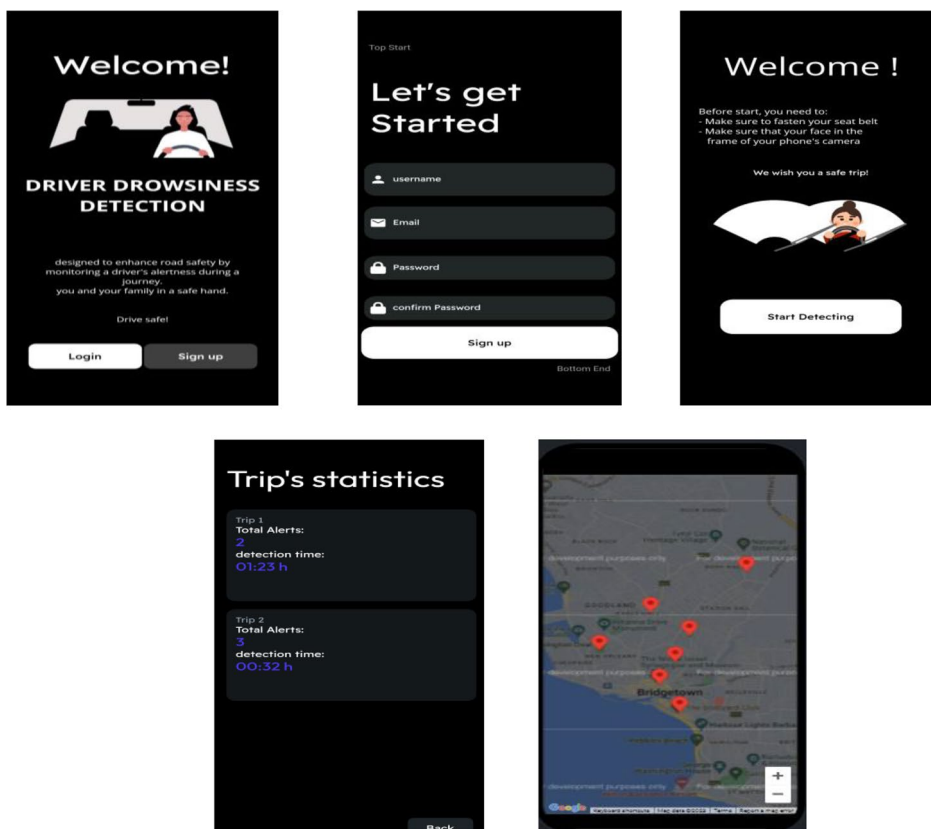
Fig. 7 Samples from the developed application

slightly lower F1 score of 95.61%. The NTHU dataset maintains high performance with accuracy, precision, recall, and F1-scores all above 96%, highlighting the model's consistency across different datasets. Lastly, on the Driver-Drowsiness dataset, the MobileNet V2 model achieves outstanding performance with near-perfect scores across all metrics, showcasing its capability in accurately detecting drowsiness with accuracy, precision, recall, and F1-score of 99.94%. Overall, the analysis underscores the efficacy and reliability of the MobileNet V2 model across various datasets, with consistently high performance in detecting eye states and drowsiness.

Application

We initially explored various applications for our drowsiness detection system, including integration with educational platforms like Microsoft Teams to monitor student attentiveness during sessions and prevent sleep-related issues. Additionally, we considered entertainment services such as Netflix, where our system could pause content if the user fell asleep, ensuring an uninterrupted viewing experience. However, recognizing the paramount importance of a life-saving application, we made a strategic decision to prioritize real-world deployment with a particular emphasis on driver drowsiness detection. This focus directly contributes to road safety, aiming to prevent accidents caused by driver fatigue. Screenshots of some pages of our developed application "Driver Drowsiness detection" are shown in Fig. 7.

Our driver drowsiness detection app, using Flutter Flow, is built after choosing the best model performance (MobileNetV2 with the dataset Driver Drowsiness). It prioritizes road safety by detecting and preventing instances of driver fatigue. Upon logging in or signing up, users are welcomed to a central hub featuring a 'Start Detection' button, leading them to the detection page. The app comprises essential sections such as Home, Sign Up, Log In, About Us, Contact Us, Profile, Edit Profile, Detection Page, and FAQs, ensuring a comprehensive user experience.

There are additional features to enhance user experience like the statistics page that provides insights into the frequency of user drowsiness during trips, fostering awareness and accountability. Another notable addition involves integrating APIs, such as Google API, to assist users in locating nearby rest areas or coffee places, promoting responsible driving habits. This approach aims not only to save lives but also to offer valuable insights and support throughout the user's journey.

Conclusions

This research paper presented a study and implementation of drowsiness detection models using CNNs and transfer learning. The conducted experiments on multiple datasets have shown promising results in terms of accuracy, making this approach a viable solution for various real-world applications. The potential for enhancing safety and engagement in domains like transportation, education, and entertainment is substantial, demonstrating the importance of continued research in this field. As part of future work, we aim to further enhance the accuracy of our drowsiness detection system by extending the models to more collected datasets.

Moreover, a plan to apply it for other real-world deployments, like educational platforms, such as Microsoft Teams, and entertainment services like Netflix can be of great help in saving various resources. Recognizing the diverse needs of users, the application can be enhanced by introducing a customizable alert system in the form of vibrations integrated into the steering wheel, allowing users to choose their preferred alarm option. This user-centric approach ensures a personalized experience, as individuals can select the most effective alert mechanism based on their preferences.

We believe this feature will not only contribute to increased safety but also address the unique requirements of users in different contexts. Several more features can be added such as the integration of haptic feedback into the system which can enhance both safety and user satisfaction.

Declarations

Competing interests

The authors declare that they have no competing interests.

References

1. K. Blake, Everything you need to know about drowsiness, 1 Agust 2019. [Online]. Available: https://www.healthline.com/health/drowsiness.
2. M. A. Kamran, M. M. N. Mannan and M. Y. Jeong (2019) Drowsiness, fatigue and poor sleep's causes and detection: a comprehensive study, IEEE Access, 99
3. Mitru G, Millrood DL, Mateika JH (2022) The impact of sleep on learning and behavior in adolescents. Teachers College Record 104(4):704–726
4. Chand HV, Karthikeyan J (2021) CNN based driver drowsiness detection system using emotion analysis. Intelligent Automation & Soft Computing 31(2):717–728
5. R. S. Duggal, (2022) Deep learning for driver drowsiness, 20
6. Ed-Doughmi Y, Idrissi N, Hbali Y (2020) Real-time system for driver fatigue detection based on a recurrent neuronal network. Journal of Imaging. 6(3):8. https://doi.org/10.3390/jimaging603000
7. M. Gomaa, R. Mahmoud and . A. Sarhan (2022) A CNN-LSTM-based deep learning approach for driver drowsiness prediction, J Engineering Res 6
8. F. Majeed, U. Shafique, M. Safran, S. Alfarhood and I. Ashraf (2023) Detection of drowsiness among drivers using novel deep convolutional neural network model, 26
9. M. Jain, . B. Bhagerathi and . Sowmyarani CN (2021) Real-time driver drowsiness detection using computer vision. Int J Eng Adv Technol 11;5
10. M. Arceda, . C. and . F. Fabian (2020) A survey on drowsiness detection techniques 10
11. S. Pachouly1, N. Bhondve, A. Dalvi, V. Dhande and N. Bhamare (2020) Driver drowsiness detection using driver drowsiness detection using behaviour. Int J Creative Res Thoughts 8; 6
12. G. S. M. Diyasa, A. H. Putra, M. R. M. Ariefwan, P. A. Atnanda, F. Trianggraeni and I. Y. Purbasari (2022) Feature extraction for face recognition using Haar Cascade Classifier, in International Seminar of Research Month 2021
13. Ali AH, Yaseen MG, Aljanabi M, Abed SA (2023) Transfer learning: a new promising techniques. Mesopotamian Journal of Big Data 2023:31–32
14. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna (2016) Rethinking the inception architecture for computer vision, in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
15. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L.-C. Chen (2018) MobileNetV2: inverted residuals and linear bottlenecks, in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition
16. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521:436–444