Vision-Based Driver Behavior Detection for Enhanced Road Safety

Tanmay Bankar [*1] **Mihika Sanghvi** [*1] **Anushka Agarwal** [*1]

Abstract

Distracted driving remains a critical factor in road traffic accidents, contributing to significant injuries and fatalities each year. We address this challenge by developing a vision-based system for real-time driver behavior detection, aimed at enhancing road safety. Using Convolutional Neural Networks (CNNs), the system classifies driver behaviors into six categories: Safe Driving, Talking, Texting, Turning, Yawning, and Engaging in Other Activities. Three state-of-the-art models were implemented and evaluated—U-Net EfficientNetB7, DenseNet121, and an ensemble model combining AlexNet-like, VGG19, and ResNet50 architectures.

1. Introduction

Distracted driving has become a significant public safety concern, with road traffic accidents ranking among the leading causes of injuries and fatalities globally. According to the World Health Organization (WHO) (1) and the National Highway Traffic Safety Administration (NHTSA) (2), distracted driving accounts for a substantial proportion of road crashes, particularly in regions with increasing smartphone usage. Distractions can take many forms, including texting, talking on the phone, eating, adjusting in-vehicle controls, and even fatigue or drowsiness. These behaviors reduce the driver's ability to focus on the road, impair reaction times, and increase the likelihood of accidents.

To combat distracted driving, modern vehicles are increasingly equipped with Advanced Driver Assistance Systems (ADAS)(3). These systems utilize technologies such as lane-keeping assistance, collision avoidance, and adaptive cruise control to enhance driving safety. Additionally, some vehicles incorporate driver monitoring systems (DMS) (4) that use sensors and cameras to detect signs of drowsiness, inattentiveness, or risky behaviors. These systems often rely on metrics such as eye closure rates, head movements, or grip pressure on the steering wheel to infer the driver's state.

While ADAS and DMS represent significant advancements, their effectiveness can be limited by factors such as high costs, lack of real-time behavioral classification, and reliance on specific hardware configurations. Furthermore, many of these systems focus on detecting a narrow range of distractions, such as fatigue, while failing to account for complex, multi-modal behaviors like texting or turning to speak to passengers. There is a growing need for cost-effective, vision-based solutions that can comprehensively analyze driver behaviors in real-time and adapt to various driving scenarios.

Vision-based systems, powered by advancements in deep learning and computer vision (5), have emerged as a promising approach to addressing this challenge. By using real-time video data (6),(7), these systems can monitor driver behaviors, classify them into predefined categories, and trigger alerts to mitigate risks effectively. Unlike traditional ADAS, vision-based systems have the potential to offer a broader scope of distraction detection (8), making them suitable for both luxury and non-luxury vehicle segments.

This paper focuses on developing a vision-based driver monitoring system that classifies driver activities into six categories, including safe and unsafe behaviors. By integrating advanced deep learning models, the system aims to provide a scalable and efficient solution to address the global challenge of distracted driving while complementing existing ADAS and DMS technologies.

2. Related Work

Vision-based driver detection systems have emerged as a promising approach to enhance road safety by identifying driver states and behaviors in real-time. These systems leverage advancements in computer vision and deep learning to process video feeds or images captured from in-cabin cameras. Below is an overview of the key research efforts in this domain:

[*]Equal contribution [1]Department of Computer Science, Columbia University, NY, United States. Correspondence to: Tanmay Bankar <ttb2121@columbia.edu>, Mihika Sanghvi <mrs2356@columbia.edu>, Anushka Agarwal <aa5477@columbia.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

2.1. Convolutional Neural Networks (CNNs) for Distracted Driving Detection

CNNs are widely used in vision-based systems for detecting distracted driving behaviors. Datasets such as the State Farm Distracted Driver Detection dataset have been pivotal in enabling research in this field. CNN architectures, including AlexNet (6), (9). VGG16, and ResNet50, have demonstrated high accuracy in classifying behaviors like texting, talking on the phone, and reaching for objects. However, despite their effectiveness in controlled environments, these models often struggle with real-world challenges, such as variations in lighting conditions, occlusions, and diverse camera angles. Recent studies have incorporated transfer learning to mitigate these limitations, further improving their generalizability.

2.2. Facial Landmark Detection and Gaze Analysis

Many studies have focused on analyzing facial landmarks and gaze tracking to detect driver fatigue and distractions. Techniques involving infrared cameras, such as EyeSight by Subaru and Affectiva's automotive AI, detect eye movements, blinking patterns, and head orientation to monitor driver attention levels. While these approaches excel in detecting fatigue and drowsiness, they are less effective at identifying distractions involving physical activities, such as smartphone usage or eating, limiting their scope.

2.3. Hybrid Models Combining CNNs and Object Detection

Hybrid models have been proposed to improve classification accuracy by integrating CNNs with object detection techniques. For example, YOLO (10) (You Only Look Once)-based models have been combined with CNNs to simultaneously recognize actions like texting or eating and detect interacting objects, such as smartphones or food items. While these hybrid models improve detection accuracy, they often require high computational resources, making them less practical for real-time deployment in vehicles.

2.4. Temporal Modeling Approaches

Some researchers have explored the use of recurrent neural networks (RNNs) and long short-term memory networks (LSTMs) (11) to model temporal dependencies in driving behaviors. By analyzing sequences of frames over time, these models can capture transitions between driver states, such as shifting from safe driving to texting. However, temporal models typically involve higher computational complexity and require extensive labeled datasets for training.

2.5. Emerging Multi-Modal Systems

Recent efforts have focused on combining visual data with additional modalities, such as audio signals and physiological data, to enhance the robustness of distraction detection (12). For instance, integrating heart rate or voice cues with video analysis provides a more comprehensive understanding of driver behavior but adds complexity to system deployment and integration.

3. Dataset

The dataset used in this paper is an augmented version of a base dataset sourced from Kaggle, originally containing five classes of driver behavior: *Texting*, *Talking*, *Turning*, *Drive Safe*, and *Other Activity*. These classes represent common activities performed by drivers, ranging from distractions like texting and talking on the phone to safe driving practices. To enhance the dataset's utility and address a critical aspect of road safety, we introduced a new class labeled *Sleepy*, representing drivers exhibiting signs of fatigue or drowsiness. This addition broadens the scope of the dataset, enabling its use in applications aimed at identifying and mitigating drowsy driving, a significant contributor to road accidents.

The images for the new *Sleepy* class were collected from diverse sources, including extracted frames from YouTube videos and other publicly available repositories, ensuring a variety of scenarios and demographics. This augmentation increases the dataset's diversity, incorporating variations in lighting conditions, camera angles, and cultural contexts. By adding this sixth class, we not only expand the coverage of driver behaviors but also improve the dataset's real-world applicability for advanced driver-assistance systems (ADAS) and road safety research. The enhanced dataset now provides a comprehensive platform for training machine learning models capable of detecting a wide range of driver activities, including potentially dangerous behaviors.

4. Data Preprocessing

To ensure the quality of the dataset, defective images were identified and removed from their respective folders, thereby maintaining the integrity of the data used for training and evaluation. The dataset was then divided into three subsets for each class: *Training* (70%), *Validation* (15%), and *Test* (15%), ensuring a balanced distribution of images across these subsets. DataFrames for all six classes, including the newly added *Sleepy* class, were created and subsequently combined into unified *Train*, *Validation*, and *Test* sets for streamlined processing.

For consistency, all images were converted to the RGB color format and saved in JPEG format. Additionally, pixel

values were normalized to the range [0, 1] to facilitate faster convergence during model training. This normalization was performed using the *ImageDataGenerator* utility for the training, validation, and test sets, ensuring a uniform preprocessing pipeline. These steps collectively improved the quality and consistency of the dataset, enabling effective training of machine learning models.

5. Proposed method using CNNs

In this paper we develop a vision-based driver behavior detection system to combat distracted driving and enhance road safety. Utilizing state-of-the-art deep learning techniques, the system identifies and classifies driver behaviors into six categories: Safe Driving, Talking on the Phone, Texting on the Phone, Turning, Yawning, and Engaging in Other Activities. The aim is to provide a real-time solution capable of improving situational awareness and potentially reducing road accidents caused by driver distractions.

To ensure high accuracy and robustness, we implemented and evaluated three cutting-edge architectures: U-Net EfficientNetB7, DenseNet121, and an Ensemble Model combining AlexNet-like, VGG19, and ResNet50 architectures. Each architecture was tailored to maximize the detection and classification of driver behaviors.

5.1. DenseNet121:

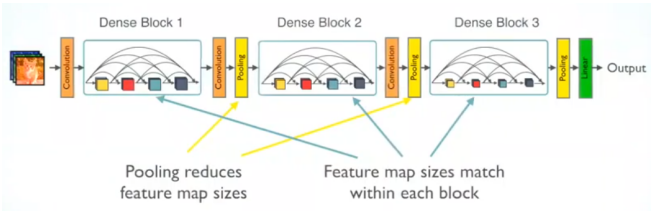

Figure 1. The architecture of DenseNet Model with three layers

DenseNet121 is known for its densely connected layers that encourage feature reuse and minimize computational redundancy. It transfers knowledge from pre-trained weights on ImageNet to effectively extract features relevant to driver behavior classification. The key features of the architecture include:
Base Model: DenseNet121, with its original fully connected layers removed (include_top=False), tailored for transfer learning.
Input Shape: Images resized to $240 \times 240 \times 3$ to fit the pre-trained model.
Custom Layers for Fine-Tuning: A GlobalAveragePooling2D layer was added to reduce the spatial dimensions while preserving features, making the model computationally efficient.
A Dense layer with 256 units and ReLU activation provided a learnable mapping of features to more abstract representations.
Dropout Regularization (rate=0.5) helped mitigate overfitting by randomly deactivating neurons during training. A final Dense layer with 6 output units and softmax activation mapped the learned features to the six driver behavior categories.

The DenseNet-based model was trained using a comprehensive configuration designed to optimize performance while maintaining generalizability. Categorical cross-entropy loss was chosen as the objective function, as it effectively measures the divergence between predicted probabilities and true labels in multi-class classification problems. The Adam optimizer was utilized due to its adaptive learning rate capabilities, enabling efficient convergence. The initial learning rate was set to 10^{-4}, and a learning rate scheduler was implemented to reduce the rate when the validation loss plateaued, ensuring steady progress. The training strategy effectively fine-tuned the pre-trained DenseNet121 model for the driver behavior classification task while ensuring its ability to generalize across diverse scenarios.

This customized architecture leverages the pre-trained feature extraction capabilities of DenseNet121 while being fine-tuned to classify specific behaviors effectively.

5.2. U-Net EfficientNetB7:

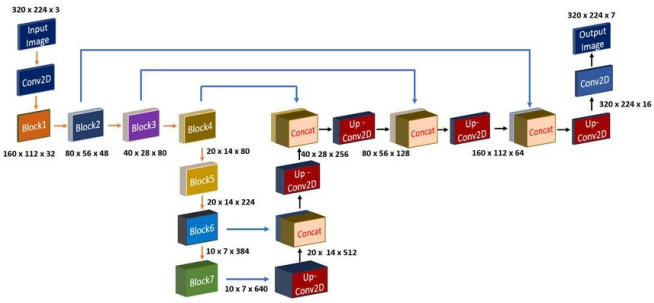

Figure 2. The architecture of U-Net EfficientNetB7 Model (13)

The Unet EfficientNetB7 model integrates the strengths of U-Net architecture and the EfficientNetB7 backbone for image classification tasks. The U-Net architecture, known for its capability to capture spatial hierarchies, has been customized to adapt to classification objectives rather than segmentation. EfficientNetB7, pre-trained on the ImageNet dataset, serves as the feature extraction backbone, leveraging its efficient and powerful deep learning capabilities to extract high-level features from input images. The model also incorporates a Global Average Pooling layer, which reduces spatial dimensions, followed by a Dense classification layer with softmax activation for multi-class prediction.

EfficientNetB7 is chosen for its exceptional feature extraction capabilities, achieved through a carefully optimized parameterization of depth, width, and resolution. The use of pre-trained weights from ImageNet significantly accelerates convergence during training, leveraging prior knowledge for effective transfer learning. Additionally, the lightweight yet powerful architecture of EfficientNetB7 makes it suitable for high-resolution image classification tasks, providing both computational efficiency and high performance. This combination of U-Net and EfficientNetB7 ensures robust feature extraction and efficient learning across all six driver behavior classes.

To prevent overfitting, the model employs several strategies. The use of Global Average Pooling instead of flattening reduces the number of trainable parameters, simplifying the model while preserving critical features. The pre-trained weights mitigate overfitting by enabling the model to build upon generalizable features learned from the ImageNet dataset. Furthermore, validation-based training is employed to monitor generalization performance, ensuring the model does not overfit the training data. Early stopping is applied as needed to halt training when no significant improvements in validation performance are observed. These measures collectively enhance the model's reliability and generalization capabilities.

5.3. Ensemble Model:

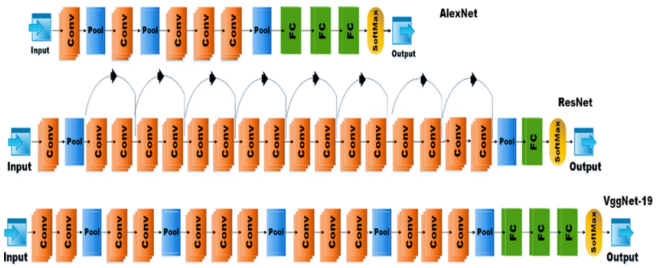

Figure 3. The architecture of Ensemble Model (14)

The Ensemble Model combines AlexNet-like, VGG19, and ResNet50 architectures which effectively uses their individual strengths. The AlexNet-like architecture is a custom convolutional neural network designed for hierarchical feature extraction. It incorporates varying kernel sizes for diverse spatial feature detection and uses max-pooling for dimensionality reduction. The network concludes with a Global Average Pooling layer to refine features. Its lightweight and flexible design makes it ideal for quick and efficient driver behavior analysis. VGG19, a deep learning architecture with 19 layers, is pre-trained on ImageNet to provide robust hierarchical feature extraction. It employs small 3×3 kernels and additional Global Average Pooling to optimize dimensionality. By enabling fine-tuning of base layers, VGG19 is effective in identifying subtle patterns, making it reliable for classifying complex driver behaviors in dynamic environments. ResNet50, a deep residual network with 50 layers, uses skip connections to prevent vanishing gradient issues, ensuring effective learning in deeper networks. Pre-trained on ImageNet, it captures detailed features and adapts to variations in lighting, occlusions, and driver profiles. Its strong generalization capabilities make it a key component for accurate and scalable driver behavior detection systems.

We then use a concatenation layer to combine feature representations from the three networks, followed by Global Average Pooling for dimensionality reduction. The dense layer with ReLU activation introduces non-linearity to the combined features, ensuring effective feature interaction before classification. To combat overfitting, the model incorporates dropout regularization, which randomly deactivates neurons during training, thereby improving generalization. Moreover, the softmax classifier enables multi-class predictions, ensuring that the model assigns probabilities to each class for accurate classification.

5.4. Distraction Score and Alarm Mechanism

The distraction score and alarm mechanism are integral components of the driver behavior monitoring system, providing actionable metrics to ensure road safety.

5.4.1. Distraction Score Calculation

The distraction score is calculated using predictions from the **ensemble model**, the best-performing model in our system. Each frame of the video is treated as an independent input to the model. The distraction score is computed as the percentage of frames classified as distracted out of the total number of frames processed.

Steps in Distraction Score Calculation

1. **Frame Extraction:** The video is split into individual frames, which are resized and preprocessed to meet the input dimensions of the ensemble model.

2. **Behavior Classification:** Each frame is passed through the ensemble model, which predicts one of six behaviors: *Safe Driving*, *Talking*, *Texting*, *Turning*, *Other Activities*, or *Sleepy*.

3. **Distraction Classification:** If the predicted behavior belongs to the distraction categories (*Talking*, *Texting*, *Turning*, *Other Activities*, or *Sleepy*), the frame is marked as distracted.

4. **Score Calculation:** The distraction score S is calcu-

lated using the formula:

$$S = \left(\frac{\text{Number of distracted frames}}{\text{Total number of frames}} \right) \times 100$$

5.4.2. Alarm Triggering Mechanism

'Sleepy' Behavior: The alarm is triggered immediately if any frame is classified as *Sleepy*, indicating a high-risk state requiring immediate intervention.

This system ensures real-time feedback to mitigate risks and enhance safety.

6. Results and Analysis

6.1. Densenet121

The DenseNet-based model achieved significant performance in classifying driver behaviors across six categories. The Confusion Matrix of this model is shown in Figure 4. During the evaluation, the model demonstrated a training accuracy of approximately 97.21%, a validation accuracy of around 98.82%, and a test accuracy of 98.42%. Metrics such as precision, recall, and F1-score indicated strong performance across most categories, other than other activities.

In terms of real-world applicability, the model exhibited low latency, making it suitable for integration into real-time driver monitoring systems. However, further optimization is necessary to address computational constraints on edge devices. Overall, the DenseNet-based model demonstrates a robust capability for driver behavior classification, offering significant potential for deployment in intelligent transportation systems to enhance road safety.

Table 1. Classification Report for DenseNet121

Class	Precision	Recall	F1-Score	Support
Drive Safe	0.96	0.97	0.97	316
Other Activities	0.99	0.97	0.98	331
Sleepy	1.00	1.00	1.00	344
Talking	0.98	0.98	0.98	326
Texting	0.99	0.99	0.99	331
Turning	0.98	0.99	0.99	309
Accuracy	0.98 (1957 samples)			
Macro Average	0.98	0.98	0.98	1957
Weighted Average	0.98	0.98	0.98	1957

6.2. U-Net EfficientNetB7

The evaluation of the Unet-EfficientNetB7 model using the confusion matrix and classification report as given in Figure 5 demonstrates strong overall performance with an accuracy of 91%. The model exhibits high precision, recall, and F1-scores across most classes, particularly excelling in detecting the *Sleepy* class with a precision of 1.00 and

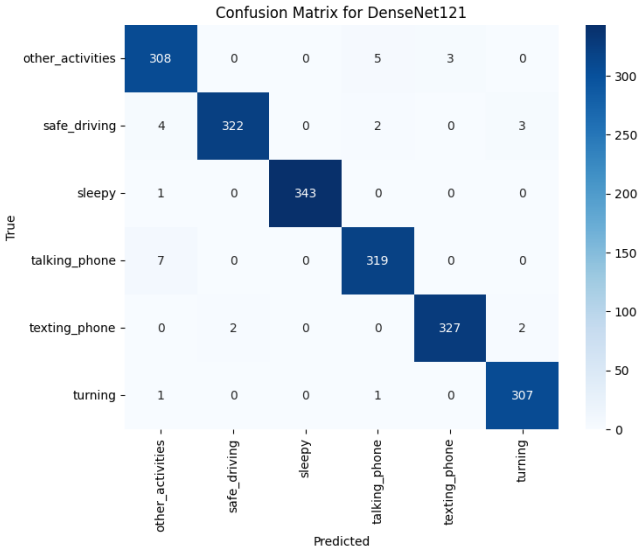

Figure 4. Confusion Matrix of Densenet121

an F1-score of 0.96. This highlights its ability to identify drowsy driving, a critical safety concern. Similarly, the *Drive Safe*, *Other Activities*, *Talking*, and *Texting* classes achieved F1-scores in the range of 0.92 to 0.94, showcasing robust classification performance. These results reflect the model's ability to effectively generalize to unseen data, leveraging its pre-trained EfficientNetB7 backbone and the custom U-Net architecture.

However, the model faces challenges, particularly with the *Turning* class, which achieved a relatively lower precision of 0.66 despite a high recall of 0.99. This indicates that while the model correctly identifies most instances of *Turning*, it also frequently misclassifies other behaviors as *Turning*, leading to a lower F1-score of 0.79. This confusion might be attributed to overlapping features or similarities between *Turning* and other classes, such as *Other Activities* or *Drive Safe*, as observed in the confusion matrix. These misclassifications suggest that the model could benefit from further refinement, such as enhanced data augmentation, better feature engineering, or the introduction of additional layers tailored to capture subtle variations in driver behaviors. Overall, while the model performs well for most classes, addressing these challenges could further improve its accuracy and reliability.

6.3. Ensemble Model

The Ensemble Model demonstrated exceptional performance, achieving a test accuracy of 99.51% with a test loss of 0.0277, outperforming the individual models used in its architecture. The classification report highlights precision, recall, and F1-scores close to or at 1.00 for all six

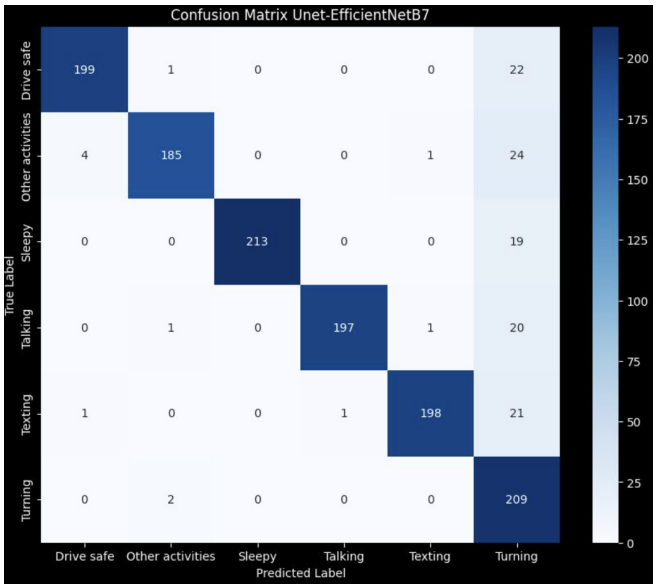

Figure 5. Confusion Matrix of Unet-EfficientNetB7

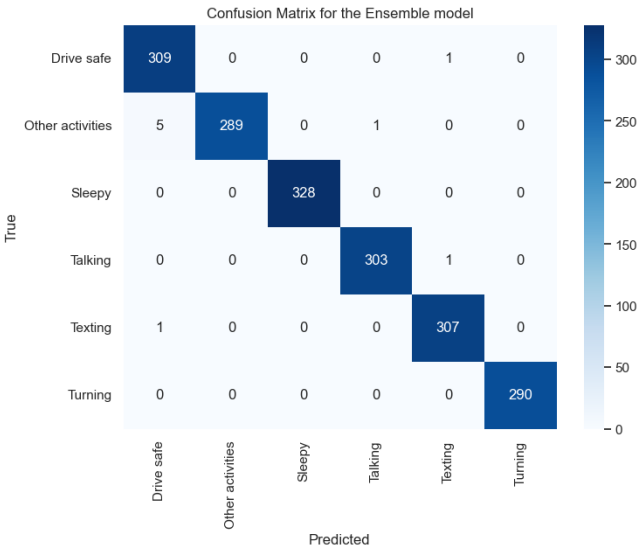

Figure 6. Confusion Matrix of the Ensemble Model

Table 2. Classification Report for Unet-EfficientNetB7

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Safe driving	0.98	0.90	0.93	222
Activities	0.98	0.86	0.92	214
Sleepy	1.00	0.92	0.96	232
Talking	0.99	0.90	0.94	219
Texting	0.99	0.90	0.94	221
Turning	0.66	0.99	0.79	211
Accuracy	0.91 (1319 samples)			
Macro Average	0.93	0.91	0.91	1319
Weighted Average	0.94	0.91	0.92	1319

Table 3. Classification Report for the Ensemble Model

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Drive Safe	0.98	1.00	0.99	310
Other Activities	1.00	0.98	0.99	295
Sleepy	1.00	1.00	1.00	328
Talking	1.00	1.00	1.00	304
Texting	0.99	1.00	1.00	308
Turning	1.00	1.00	1.00	290
Accuracy	1.00 (1835 samples)			
Macro Average	1.00	0.99	1.00	1835
Weighted Average	1.00	1.00	1.00	1835

driver behavior categories, including Safe Driving, Texting, and Talking, among others. The confusion matrix further confirms the model's superior ability to classify driver behaviors accurately, with minimal misclassifications and solid performance across diverse test scenarios.

Compared to the standalone models—U-Net EfficientNetB7, DenseNet121, and ResNet50—the Ensemble Model consistently achieved higher classification metrics. By combining the strengths of AlexNet-like, VGG19, and ResNet50 architectures, it leveraged complementary features and representations, which significantly enhanced its ability to generalize across varying conditions such as lighting, occlusions, and camera angles. While individual models exhibited certain limitations, such as overfitting tendencies or reduced generalizability in specific categories, the ensemble approach effectively mitigated these issues through feature diversity and regularization techniques.

The results underscore the Ensemble Model's advantage in real-world applications where high accuracy and generalizability are crucial, especially for detecting high-risk situations like driver drowsiness. Its ability to outperform the individual architectures validates the effectiveness of ensemble learning and is a reliable solution for integration into driver monitoring systems aimed at reducing road accidents and ensuring safe driving.

6.4. Frame-Wise Processing and Role of the Ensemble Model

The proposed system processes video data frame by frame, treating each frame as an independent input to the ensemble model. This approach eliminates the need for temporal modeling or video-specific input formats, enabling a scalable and real-time implementation.

6.4.1. Aggregation Over Frames

- Predictions for individual frames are aggregated over the entire video to compute the distraction score and

trigger alarms.

- While this does not capture temporal dependencies explicitly, it provides a robust overview of driver behavior across the video.

6.4.2. Role of DenseNet and Other Models

- DenseNet121 and other models (*U-Net-EfficientNet* and individual models in the ensemble) were used during experimentation phases.

- For final deployment, the **ensemble model** was selected due to its superior classification performance. This model processes frames sequentially, ensuring accuracy and reliability in real-time settings.

No model in the system processes multiple images simultaneously. Each frame is processed independently, classified by the ensemble model, and used to calculate the distraction score or trigger alarms.

This design ensures that the system is lightweight and efficient, suitable for practical deployment in real-world scenarios.

6.5. Real-Time Detection and Deployment

The developed system processes video input in real time, sampling frames at three-second intervals to ensure efficient and timely analysis. Each sampled frame is resized and preprocessed to meet the input requirements of the model, which then predicts the driver's behavior. The predictions, along with their corresponding confidence scores, are overlaid on the frame to provide visual feedback. This real-time processing capability ensures the system's applicability in practical driving scenarios, offering an intuitive and user-friendly interface for monitoring driver behavior.

To enhance safety, the system integrates an alarm mechanism that is triggered in response to drowsiness (*Sleepy*) or unsafe behaviors. This immediate feedback mechanism alerts drivers to potentially hazardous situations, reducing the likelihood of accidents. The system was tested on three sample videos, yielding promising results with accurate classification of driver behaviors and timely alarm activations for detected unsafe actions. Since video demonstrations cannot be included, we provide two examples of image frames in Figures 7 and 9, along with their respective logs shown in Figures 8 and 10. These illustrate the model's predictions, confidence scores, and whether the alarm was triggered during testing. These results underscore the system's effectiveness and its potential for real-world applications in ensuring driver safety.

Figure 7. Frame showing *Sleepy* behavior prediction with confidence score and alarm triggered.

```
Distraction Detection Log
Processing video: video3.mp4

Alarm triggered at frame 0
Alarm triggered at frame 59
Alarm triggered at frame 118
Alarm triggered at frame 236
Alarm triggered at frame 295
Alarm triggered at frame 354
Alarm triggered at frame 413
Alarm triggered at frame 472
Alarm triggered at frame 531
Alarm triggered at frame 590
Alarm triggered at frame 649
Alarm was triggered during the video due to 'Sleepy' detection.
```

Figure 8. Log showing alarm triggered for *Sleepy* behavior detection.

7. Statement of Contribution

7.1. Anushka Agarwal - DenseNet Implementation

Anushka was responsible for implementing DenseNet. Anushka's implementation focussed on leveraging the layer-by-layer connections that DenseNet offers, ensuring that all layers receive gradients directly from the loss function, promoting better learning. Anushka worked on fine-tuning this architecture to balance efficiency with accuracy, and her task includes optimizing DenseNet to minimize latency in real-time driver detection. She also evaluated DenseNet's individual performance in isolation before it's integrated into the ensemble model, documenting key metrics and contributing to the overall performance validation.

7.2. Mihika Sanghvi - UNet-EfficientNetB7 Integration and Data Preprocessing

Mihika performed data preprocessing, ensuring consistent formatting and quality, which was utilized across all models

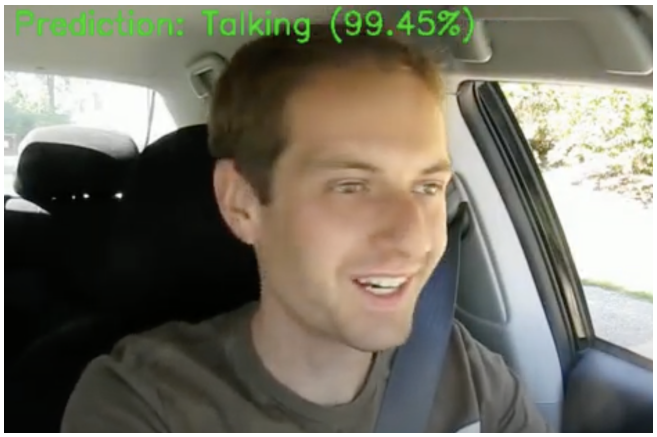

Figure 9. Frame showing *Talking* behavior prediction with confidence score.

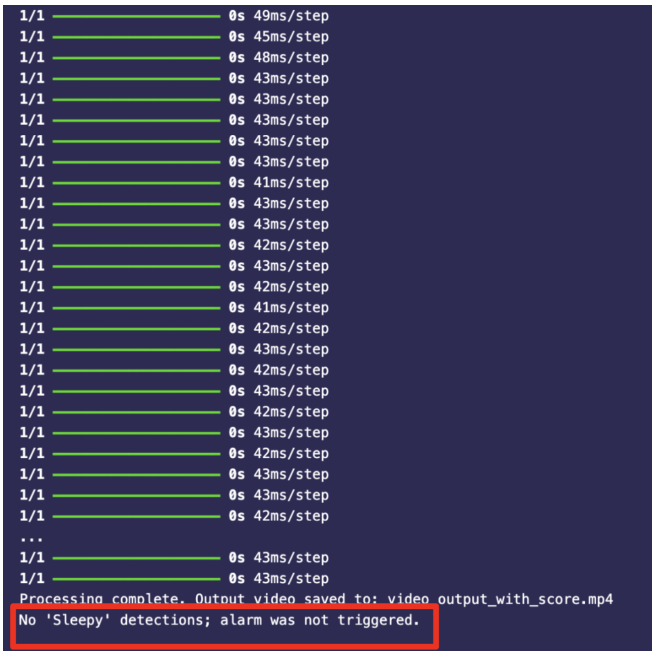

Figure 10. Log showing no alarm triggered for *Talking* behavior detection.

in the system. Mihika was also tasked with implementing UNet with EfficientNetB7 as the backbone to enhance feature localization in our system. Mihika integrated the EfficientNetB7 backbone for efficient and deep feature extraction while using the UNet architecture's skip connections to maintain spatial accuracy. Her focus was on training and optimizing the UNet-EfficientNetB7 while maintaining high-resolution feature maps, which are essential for identifying subtle indicators of driver distraction. Additionally, Mihika worked on configuring the model to adapt to a diverse set of driver behavior scenarios and contribute to the alarm and distraction score functionality by implementing feature localization techniques.

7.3. Tanmay Bankar - Ensemble Model Development and Alarm Trigerring Mechanism

Tanmay's role was to implement and optimize the ensemble model. Tanmay's focus was on determining the most effective ensemble technique to achieve high accuracy in detecting driver behaviors. He worked on tuning the ensemble model for a seamless balance between accuracy and real-time performance. Tanmay was also responsible for testing the complete ensemble on diverse datasets to ensure consistent behavior detection. Furthermore, Tanmay incorporated the final distraction detection mechanism, including the alarm trigger and distraction score calculation, as part of the ensemble's output.

References

[1] W. H. Organization, "Distracted driving: A global concern," *WHO*, 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

[2] N. H. T. S. Administration, "Distracted driving 2022 statistics," 2022. [Online]. Available: https://www.nhtsa.gov/risky-driving/distracted-driving

[3] D. Zhao and W. Xiang, "Advanced driver assistance systems (adas) technologies and their applications," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 3, pp. 355–366, 2021.

[4] Y. Abouelnaga, H. M. Eraqi, and M. N. Moustafa, "Driver monitoring systems (dms): A critical review and future directions," *IEEE Access*, vol. 8, pp. 120 367–120 381, 2020.

[5] S. Jain and V. Kanhangad, "Driver activity recognition for intelligent vehicles using deep learning," *IEEE Sensors Journal*, vol. 20, no. 10, pp. 5500–5508, 2020.

[6] Y. Abouelnaga, H. M. Eraqi, and M. N. Moustafa, "Real-time distracted driver posture classification using alexnet," 2017. [Online]. Available: https://arxiv.org/abs/1706.09498

[7] J. Huang, X. Wang, and L. Liu, "Deep learning approaches for driver behavior detection using resnet architectures," *Pattern Recognition Letters*, vol. 147, pp. 35–42, 2021.

[8] T. Ko, D. Shin, and D. Lee, "Driver monitoring via deep learning techniques," *IEEE Access*, vol. 8, pp. 191 771–191 785, 2020.

[9] Y. Cao, Y. Xing, and X. Li, "Driver distraction detection using gaze behavior and deep learning methods," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 5, pp. 4991–5004, 2022.

[10] X. Huang, X. Wang, and L. Ma, "A yolo-based real-time object detection approach for driver distraction recognition," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 3, pp. 359–369, 2020.

[11] V. Ramanishka, L. A. Hendricks, A. Rohrbach, and T. Darrell, "Driver activity analysis using temporal convolutional networks and lstms," *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 303–320, 2018.

[12] H. Ko and J. Lee, "Multimodal distraction detection using video and audio signals in driver monitoring systems," *IEEE Transactions on Multimedia*, vol. 23, pp. 3481–3492, 2021.

[13] B. Baheti, S. Innani, S. Gajre, and S. Talbar, "Eff-unet: A novel architecture for semantic segmentation in unstructured environment," 06 2020, pp. 1473–1481.

[14] I. Wani and S. Arora, "Osteoporosis diagnosis in knee x-rays by transfer learning based on convolution neural network," *Multimedia Tools and Applications*, vol. 82, pp. 1–25, 09 2022.