

Minimizing Parameters, Maximizing Safety: A Deep Learning Approach to Real-Time Distracted Driver Identification

Georgios Markos Chatziloizos, Andrea Ancora, Christian Barat, Andrew I. Comport

▶ **To cite this version:**

Georgios Markos Chatziloizos, Andrea Ancora, Christian Barat, Andrew I. Comport. Minimizing Parameters, Maximizing Safety: A Deep Learning Approach to Real-Time Distracted Driver Identification. SIA Vision, Oct 2024, Paris, France. hal-05405999

HAL Id: hal-05405999
https://hal.science/hal-05405999v1

Submitted on 9 Dec 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Minimizing Parameters, Maximizing Safety: A Deep Learning Approach to Real-Time Distracted Driver Identification

Georgios Markos Chatziloizos[1,2], Andrea Ancora[2], Andrew I. Comport[1], Christian Barat[1]

1: I3S-CNRS/Universite Cote d'Azur, Nice, France
2: Ampere, Nice, France

Abstract: One of the most concerning causes of road accidents is distracted driving due to talking to the phone, texting, drinking, etc. Thus, this work aims to develop a highly accurate and robust system for timely detection of distracted drivers. This research delved into a comprehensive exploration of various methodologies ranging from foundational models (like VGG 16 and DINOv2) to state of the art Convolutional Neural Networks (CNNs). The foundational models explored in this study, while demonstrating notable performance, exhibit limitations in real-time responsiveness, which is a crucial aspect for addressing the dynamic nature of road safety concerns. Despite their effectiveness, the examined foundational models fall short in delivering the reaction times necessary for real-time applications. Our proposed framework, based on Convolutional Neural Networks, not only identifies distracted drivers but also discerns the specific causes of distraction. Emphasizing practical applicability and adoption within the automotive industry, our models are meticulously designed to minimize parameter count. Experimental findings reveal that our system surpasses previous methodologies in the literature. Furthermore, we present architectures with a significant reduction in parameters, ensuring efficiency and reduced response time compared to traditional models. This reduction in response time contributes to the system's effectiveness in issuing timely warnings to the driver to enhance road safety.

Keywords: Deep Learning, Autonomous Driving, Distracted Driver Identification

1. Introduction

Distracted driving has become one of the leading causes of road accidents, with drivers engaging in various activities such as talking on the phone, texting, eating and even drinking while behind the wheel [2,16]. These distractions divert attention from the road, increasing the likelihood of collisions and fatalities. As the automotive industry moves towards greater levels of automation and safety features, the need for intelligent driver monitoring systems has gained significant attention. The development of an accurate and efficient system capable of detecting distracted drivers in real-time is crucial for enhancing road safety.

In this work, we focus on building a robust system for timely identification of distracted drivers, which can be deployed in edge devices or automotive systems. Several deep learning methodologies have been explored for this purpose, ranging from classical models like VGG16 to more recent advances like DINOv2 [18]. While these foundational models are highly accurate, they often fall short in terms of real-time responsiveness, an essential requirement for real-world deployment. Given that road safety demands swift reaction times to prevent accidents, models that require high computational resources or exhibit slower inference times are unsuitable for edge deployment. The Convolutional Neural Networks (CNNs) are designed with a focus on efficiency and low-latency response. These CNN-based architectures aim to strike a balance between high accuracy and real-time performance. Furthermore, we emphasize parameter reduction without sacrificing model performance, making these models lightweight and optimal for deployment in resource-constrained environments, such as in-vehicle systems. In summary, this research introduces a novel approach to driver monitoring that not only detects distractions but also identifies their specific causes. Through extensive experimentation, we show that our models outperform existing solutions in the literature in terms of both accuracy and computational efficiency, making them well-suited for real-time driver monitoring applications.

2. Related Work

Distracted driving detection has been the subject of extensive research, with approaches ranging from traditional computer vision techniques to modern deep learning methodologies. Early work in this area primarily focused on using handcrafted features to detect specific distractions [19], such as eye gaze detection or head movement tracking. These systems, while effective in controlled environments, often failed to generalize well to diverse real-world conditions. The advent of deep learning brought about a paradigm shift in driver monitoring systems. Models such as VGG16 and ResNet have been employed for driver behaviour analysis, leveraging their strong feature extraction capabilities. VGG16, for

instance, has been widely used in various vision tasks, including distracted driving detection, due to its robust performance on large datasets. However, its large number of parameters and slow inference times make it unsuitable for real-time applications, particularly on edge devices with limited computational resources.

In the field of distracted driver classification, researchers have explored various neural network architectures and techniques, each offering distinct contributions and innovations. For instance, in the work by [15], the authors used the VGG-GAP architecture (VGG with Global Average Pooling) [17] to visualize the patterns learned by the neural network. This structure is particularly designed to provide insights into the features the network focuses on during the task, helping to better analyze the learning process. Additionally, they implemented an ensemble of VGG-16 [13] and VGG-GAP, which resulted in slightly improved performance, achieving an accuracy of 92.6%.

In a separate study by [8], the researchers developed a Vanilla CNN model with multiple convolutional layers, followed by a flattening layer and dense layers. To enhance the model's generalization, they applied data augmentation techniques. They also introduced a transfer learning approach, employing a CNN with fewer convolutional and dense layers to optimize training time without sacrificing accuracy.

Another innovative approach was presented by the authors of the DHCNN model [11], which introduced a unique neural network structure for classifying distracted drivers. This model focused on extracting features using the Histogram of Oriented Gradients (HOG) method [3], where 2x2 regions of 224x224 grayscale images were used to generate eight-dimensional histograms for each image. The D-HCNN architecture consisted of four convolutional layers, each with progressively smaller filter sizes and used dropout to prevent overfitting.

In a different study, the authors of the Hybrid CNN Framework (HCF) [7] introduced a model composed of three key modules. The first module, the cooperative CNN, extracted features related to distracted behaviour using pretrained models, which were later fine-tuned through transfer learning. The second module fused the extracted features to create inputs for the final module, which was responsible for classifying various distracted driving behaviours.

These diverse approaches highlight the ongoing evolution of neural network architectures and methodologies in distracted driver classification, ranging from visualization techniques to transfer learning and innovative architectures for feature extraction. Our research aims to advance technology in driver safety while addressing the critical need for real-time decision-making. By accurately detecting the source of distractions, our system allows for timely interventions, potentially preventing accidents while using minimal memory resources. Ultimately, such systems aim to reduce distractions and improve overall road safety.

3. Dataset

The dataset used for training our models, as described in **Figure 1**, is the State Farm Distracted Driver Detection dataset [1]. It consists of ten different classes, making it a multi-class classification problem.



Figure 1: The 10 classes of the dataset

The first class represents normal driving, while the other nine involve various forms of distractions. These include texting with the right hand, talking on the phone using the right hand, texting with the left hand, talking on the phone using the left hand, operating the radio, drinking, reaching behind, grooming (such as doing hair and makeup) and talking to a passenger.

4. Architectures

In this work, we experimented with several architectural designs, balancing the need for high accuracy and the real-time constraints imposed by edge devices in automotive environments. The goal was to find lightweight, efficient models capable of detecting driver distractions swiftly and accurately, while minimizing the computational overhead associated with more complex models like transformers.

4.1. CNN and CNNsmall Architectures

The Convolutional Neural Network (CNN) architecture introduced in [6] was originally developed to classify images into 10 distinct categories, with the goal of identifying driver behaviour. This network is built using a series of MixNet-inspired blocks [14], which include convolutional layers, depthwise separable convolutions and residual connections.

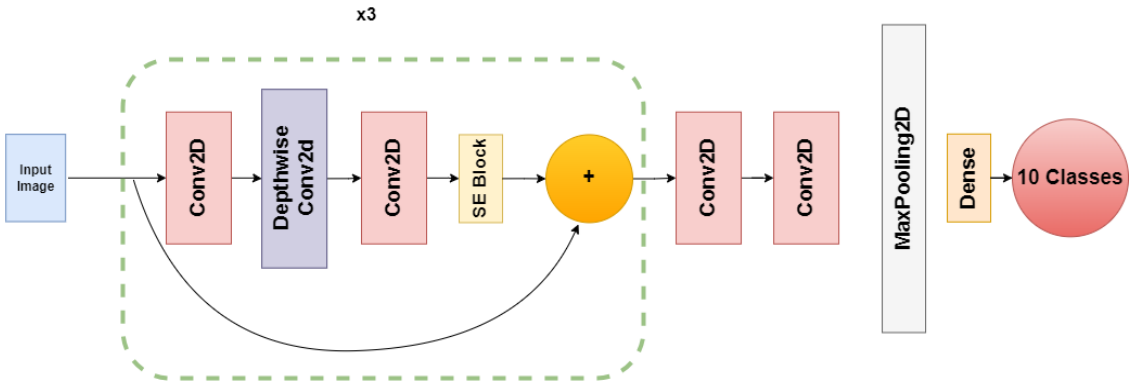


Figure 2: Architecture of CNN model [6]

As shown in **Figure 2**, the initial block starts with a convolutional layer containing 60 channels and a 5x5 kernel, followed by batch normalization and a depthwise separable convolution using a 3x3 kernel with the same padding. This block also integrates a Squeeze-and-Excitation (SE) mechanism [5] to enhance the network's ability to capture detailed patterns. Subsequent blocks refine this structure by progressively expanding the depthwise convolutions to use larger kernels, such as 5x5 and 7x7.

To improve performance, the Swish activation function [12] is applied, as it demonstrated better results compared to ReLU [10] and GeLU [4] in preliminary testing. Dropout layers are also incorporated to reduce the risk of overfitting during the training process.

After the series of MixNet-style blocks, the network concludes with a final convolutional block. This block includes convolutional layers with Swish activation, batch normalization and max-pooling, all of which help establish the network's spatial hierarchy. The output layer, using a softmax activation function, assigns probabilities to classify the input images into one of the specified categories. The Adam optimizer [9] is employed to adjust the model's parameters, aiming to maximise classification accuracy while maintaining low parameter count.

CNNsmall is a simplified version of the original CNN architecture, with a significantly reduced parameter count of 50k. Despite its smaller size, CNNsmall maintains the same underlying architecture, allowing for efficient deployment on devices with limited computational resources.

4.2. DINOv2 Architecture

For comparison, we also evaluated DINOv2, a transformer-based model renowned for its high accuracy in vision tasks. DINOv2 employs a self-supervised vision transformer (ViT) architecture, which allows the model to learn representations from unlabeled data. Its architecture consists of multiple transformer blocks that leverage multi-head self-attention mechanisms to capture long-range dependencies in the input images. While DINOv2 achieves one of the highest accuracies in our experiments, it comes at the cost of significantly higher computational requirements. DINOv2 has 22 million parameters, making it over 100 times larger than CNN. Depending on the hardware it can make it

makes unsuitable for real-time applications in edge devices, where both memory and processing power are limited.

4.3. Model Optimization Techniques

To further enhance the efficiency of our models, we applied optimization techniques commonly used in the deployment of deep learning models in resource-constrained environments. We used in our architectures early stopping, which prevented overfitting during training by halting the process once the validation loss ceased improving. This not only improved generalization but also reduced the overall training time for our models.

5. Result

We conducted extensive experiments to evaluate the performance of our models, focusing on accuracy, parameter count and inference times on both GPU and NPU platforms. We trained the models on A6000 Nvidia GPUs and tested the inference time on Qualcomm SA8255P platform which is specifically designed for automotive applications.

As shown in **Table 1**, numbers that are bold are the best in their category and the underscored ones are the second best. Our CNNsmall and CNN models achieved high accuracy while significantly reducing the number of parameters compared to DINOv2. CNNsmall, with just 50 thousand parameters, achieved an accuracy of 99.04%, making it one of the most efficient models in terms of both accuracy and inference speed. CNN, which slightly increased the parameter count to 0.2 million, improved the accuracy to 99.46%. Both CNN models demonstrated remarkable inference speed, particularly on NPUs, where CNNsmall achieved an inference time of just 0.2 ms and CNN processed images in 0.5 ms. This makes both models suitable for real-time applications on edge devices. In comparison, DINOv2, while achieving the highest accuracy at 99.62%, exhibited significantly slower inference times, with 9.6 ms on NPUs and 18 ms on GPUs and a much larger parameter count of 22 million.

	Accuracy	**#Parameters (in Millions)**	**Npu time (in ms)**	**GPU time (in ms)**
VGG-16	90.2	140	-	-
VGG-GAP	91.3	140	-	-
Ensemble VGG-16 & VGG-GAP	92.6	140 + 140	-	-
Vanilla CNN with Data Augmentation	97	26	-	-
CNN with Transfer Learning	71.72	3.5	-	-
HCF	96.74	72.3	-	-
D-HCNN	**99.86**	0.76	-	-
CNNsmall	99.04	**0.05**	**0.2**	**1.3**
CNN	99.46	<u>0.2</u>	<u>0.5</u>	<u>3.1</u>
DINOv2	<u>99.62</u>	22	9.6	18

Table 1: Accuracies and inference times for different architectures

6. Conclusion

In this study, we presented efficient and highly accurate systems for detecting distracted driving in real-time. Our research demonstrated that while transformer-based models like DINOv2 offer impressive accuracy, their computational complexity and slower inference times make them less suitable for real-time applications in edge environments for easier tasks. By contrast, our proposed CNNsmall and CNN models strike an optimal balance between performance and efficiency. CNNsmall and CNN provide comparable accuracy to state-of-the-art models, with CNN achieving an accuracy of 99.46%, very close to the 99.62% accuracy of DINOv2. However, our models offer a significant reduction in both parameter count and inference time, with

CNNsmall performing inference in just 0.2 ms on NPUs, making it ideal for deployment in vehicles where real-time performance is critical. This reduction in response time allows our system to issue timely warnings to drivers, potentially preventing accidents caused by distractions. Our work highlights the importance of optimising deep learning models for real-time deployment in safety-critical applications and future research will focus on further refining these models and extending their capabilities to additional tasks related to driver behaviour monitoring and accident prevention.

7. Acknowledgement

The authors would like to acknowledge the Association Nationale Recherche Technologie (ANRT) for CIFRE funding (2023/0751).

8. References

[1] Anna Montoya SF_data_science Dan Holman. 2016. “State Farm Distracted Driver Detection.” Kaggle. https://kaggle.com/competitions/state-farm-distracted-driver-detection.

[2] Chien, Tzu-Chih, Chieh-Chuan Lin, and Chih-Peng Fan. 2020. “Deep Learning Based Driver Smoking Behavior Detection for Driving Safety.” Journal of Image and Graphics 8 (January): 15–20. https://doi.org/10.18178/joig.8.1.15-20.

[3] Dalal, N., and B. Triggs. 2005. “Histograms of Oriented Gradients for Human Detection.” In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, 1:886–893. https://doi.org/10.1109/CVPR.2005.177.

[4] Hendrycks, Dan, and Kevin Gimpel. 2023. “Gaussian Error Linear Units (GELUs).” https://arxiv.org/abs/1606.08415.

[5] Hu, Jie, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. 2019. “Squeeze-and-Excitation Networks.” https://arxiv.org/abs/1709.01507.

[6] Chatziloizos, Georgios Markos, Andrea Ancora, Andrew I. Comport, and Christian Barat. 2024. "Low Parameter Neural Networks for In-Car Distracted Driver Detection." *Proceedings of the 2024 9th International Conference on Machine Learning Technologies*, 204–208. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3674029.3674062.

[7] Huang, Chen, Xiaochen Wang, Jiannong Cao, Shihui Wang, and Yan Zhang. 2020. “HCF: A Hybrid CNN Framework for Behavior Detection of Distracted Drivers.” IEEE Access 8: 109335–49. https://doi.org/10.1109/ACCESS.2020.3001159.

[8] Jamsheed, V. Abdul, B. Janet, and U. Srinivasulu Reddy. 2020. “Real Time Detection of Driver Distraction Using CNN.” In *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 185–91. https://doi.org/10.1109/ICSSIT48917.2020.9214233.

[9] Kingma, Diederik P., and Jimmy Ba. 2017. “Adam: A Method for Stochastic Optimization.” https://arxiv.org/abs/1412.6980.

[10] Nair, Vinod, and Geoffrey E. Hinton. 2010. “Rectified Linear Units Improve Restricted Boltzmann Machines.” In *International Conference on Machine Learning*. https://api.semanticscholar.org/CorpusID:15539264.

[11] Qin, Binbin, Jiangbo Qian, Yu Xin, Baisong Liu, and Yihong Dong. 2022. “Distracted Driver Detection Based on a CNN with Decreasing Filter Size.” IEEE Transactions on Intelligent Transportation Systems 23 (7): 6922–33. https://doi.org/10.1109/TITS.2021.3063521.

[12] Ramachandran, Prajit, Barret Zoph, and Quoc V. Le. 2017. “Searching for Activation Functions.” https://arxiv.org/abs/1710.05941.

[13] Simonyan, Karen, and Andrew Zisserman. 2015. “Very Deep Convolutional Networks for Large-Scale Image Recognition.” https://arxiv.org/abs/1409.1556.

[14] Tan, Mingxing, and Quoc V. Le. 2019. “MixConv: Mixed Depthwise Convolutional Kernels.” https://arxiv.org/abs/1907.09595.

[15] Zhang, Ben. 2016. “Apply and Compare Different Classical Image Classification Method: Detect Distracted Driver.” In. https://api.semanticscholar.org/CorpusID:37775168.

[16] Zhao, Li, Yang Liu, and Nianqiang Li. 2019. “Fatigue Driving Detection System Based on Bayes’ Theorem.” Journal of Image and Graphics 7 (January): 76–81. https://doi.org/10.18178/joig.7.3.76-81.

[17] Zhou, Bolei, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2015. “Learning Deep Features for Discriminative Localization.” https://arxiv.org/abs/1512.04150.

[18] Oquab, Maxime, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. 2023. "DINOv2: Learning Robust Visual Features without Supervision." *arXiv:2304.07193*.

[19] O'Mahony, Niall, Sean Campbell, Anderson Carvalho, Suman Harapanahalli, Gustavo Velasco Hernandez, Lenka

Krpalkova, Daniel Riordan, and Joseph Walsh. 2020. "Deep Learning vs. Traditional Computer Vision." In *Advances in Computer Vision*, edited by Kohei Arai and Supriya Kapoor, 128–144. Cham: Springer International Publishing.