International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Towards Safer Roads: A Deep Learning Approach to Driver Distraction Detection in Four-Wheeler Cars

Prof. Sumit Shevtekar[1], Yadhnesh Gangurde[2]

[1]Assistant Professor, Department of Computer Engineering, Pune Institute of Computer Technology
[2]Student, Department of Computer Engineering, Pune Institute of Computer Technology Pune, India

Abstract
The fact that there are more than 50 million cars sold annually and more than 1.3 million fatal motor vehicle accidents each year indicates the urgent need for stronger road safety regulations. Driver behaviour needs to be addressed, especially in emerging nations like India, which is responsible for 11% of all road fatalities worldwide. Diversion has been identified as the leading cause of 78% of accidents involving drivers. Distractions can take many different forms, from using a phone to interacting with others, and they greatly hinder road safety. This work aims to address this important problem by creating a highly effective machine learning (ML) model that employs computer vision techniques to classify various driver distractions in real- time. Utilising cutting-edge models, such as an ensemble of CNNs and convolutional neural networks (CNNs) like ResNet50. Our goal is to effectively identify and categorise distractions through the use of deep learning and picture recognition, allowing for proactive intervention to avert mishaps. Beyond classification accuracy, this study evaluates the model's overall speed and scalability, which are critical for deployment on edge devices. We assess the practical practicality and wide-spread adoption of our approach by analysing performance parameters like inference time and resource utilisation.

Index Terms: Road safety, Driver behavior, Distraction, Ma- chine learning (ML), Computer vision, Real-time classification, Convolutional neural networks (CNNs), ResNet50, Ensemble learning, Deep learning, Proactive intervention,

I. Introduction

The issue at hand is related to the automotive industry, where worrying data showing more than 50 million cars sold yearly and more than 1.3 million fatalities from motor vehicle accidents highlight how urgent it is to solve issues related to road safety [6]. Notably, 11% of road accident fatalities worldwide occur in India, underscoring the seriousness of the issue. The financial cost is high; in FY 18–19, vehicle insurance claims totaled Rs. 58,456.932 crores, or a sizeable percentage of the nation's GDP yearly.

A startling 78% of accidents are deemed to be the fault of the driver, and human behaviour is the main cause of road safety problems, especially in developing nations [7]. Notwith- standing these obstacles, the automotive sector is currently observing a paradigm change in favour of technology-based solutions, most notably with the introduction of fully auto- mated and networked automobiles. This technical development offers a special chance to successfully solve issues related to road safety [8].

Technology advancements are pushing the automotive in- dustry towards safer roads, but it's also critical to recognise the socioeconomic implications of road safety. In addition to the horrific death toll, automobile crashes entail serious consequences like significant financial losses and strain on the healthcare system.

In addition to the direct casualties, road accidents can have an impact on the victims' families and communities. Many of these occurrences result in long-term disabilities, which diminish quality of life and increase dependency on social support systems. Moreover, there are serious financial ramifications because lost productivity, medical expenses, and rehabilitation costs can take a sizable portion of national budgets.

Addressing these difficulties becomes essential for sus-tainable growth in rising economies like India, where rapid urbanisation and infrastructural challenges increase road safety concerns. It is the responsibility of civil society, business, and government organisations to work together to develop creative solutions that reduce hazards and encourage safe driving practices.

Therefore, incorporating AI-driven technology is a workable solution for proactive risk management and accident preven- tion. By utilising real-time data analytics and prediction algo- rithms, stakeholders may efficiently identify high-risk areas, implement targeted interventions, and promote safer driving practises across a broad population.The use of technology is supplemented by initiatives aimed at fostering a culture of road safety through campaigns for awareness and education. Encouraging behavioural changes, such as putting down gadgets and following traffic laws, ne- cessitates a multifaceted approach that combines technological innovation, community involvement, and policy enforcement. In other words, resolving the issues related to, solving the problems associated with road safety necessitates a comprehensive strategy that goes beyond technology advancement to include social, economic, and regulatory aspects. We may aim to construct roads that are not only effective and technologi- cally sophisticated, but also safe and inclusive for all users by combining the efforts of stakeholders from all sectors.

By utilising artificial intelligence (AI), a real-time alert system that serves as a reminder to drivers to maintain concentration can be created, hence lowering the likelihood of accidents and minimising the loss of life and property. Distracted driving, which can result from using a phone, drinking, and interacting with others, is a major cause of accidents. Our proposal to address this problem is to create a real-time distraction detection system that can notify drivers in real-time to avoid unfavourable consequences [9]. In order to evaluate data and give insights asynchronously, our method involves installing an edge device configuration inside the car, enabling rapid alarms and interactions over IoT. This project's main goal is to create a highly effective machine learning (ML) model that uses computer vision techniques to categorise different types of driver distractions during runtime. Furthermore, The scalability and speed of the model to ensure a seamless integration into edge device settings.

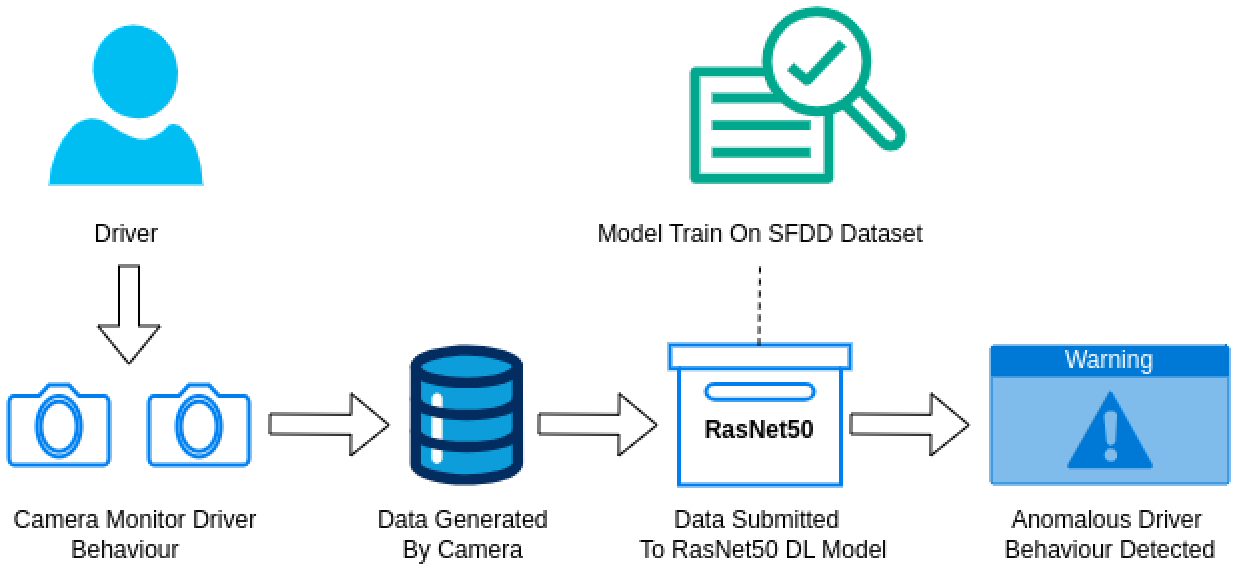

Fig. 1. Proposed System

II. Related Work

In measuring driver distraction, various approaches are employed, focusing on visual, manual, or cognitive aspects. In the manual and cognitive domains, researchers have explored methodologies such as monitoring lane maintenance, speed performance, and the duration of lane departures to infer the driver's state [13]. Castignani et al. classified driving events as risky or not by analyzing acceleration, braking, and steering activities through the SenseFleet system [16]. Pavlidis et al. conducted statistical analyses to understand the relationship between driver distractions and various driving parameters [17]. A forward collision warning algorithm based on the driver's braking activity was proposed by Wang et al. [18].Vi- sual assessments, such as head position, facial expressions, pupil diameter, and eye gazing,

Visual measurements, including eye gaze, pupil diameter, head pose, facial expressions, and driving posture, offer rich data for detecting driver distractions. These visual cues have been leveraged in various methodologies, including mathematical models, rule-based models, and models based on machine learning (ML) algorithms [19]. Among ML-based approaches, Baheti et al. developed a Convolutional Neural Network (CNN) system specifically tailored for detecting different driver actions, showcasing the potential of deep learning in driver distraction detection [6][7]. In another study, Huang et al. proposed a CNN model aug- mented with cooperative pre-trained models such as ResNet, Inception, and Xception, along with a novel dropout layer to mitigate overfitting. Their approach achieved impressive classification accuracy on the AUC dataset, outperforming other CNN classifiers [26].

Furthermore, Jamsheed V. et al. introduced a novel structure incorporating vanilla CNNs, vanilla augmented CNNs, and deeper CNNs based on transfer learning for driver distraction detection. Their evaluation on the AUC dataset demonstrated significant accuracy improvements, especially when leveraging transfer learning techniques [28].

These studies collectively underscore the importance of leveraging advanced computational techniques, particularly deep learning and transfer learning, to enhance the accuracy and robustness of driver distraction detection systems, con- tributing to the ongoing efforts to improve road safety.

Expanding on the existing literature, recent advancements in driver distraction detection have seen an increasing focus on leveraging multimodal data fusion techniques. Researchers have recognized the potential of combining information from various sensors, including cameras, accelerometers, gyro-scopes, and biometric sensors, to capture a comprehensive understanding of driver behavior [20].

Furthermore, advancements in deep learning architectures have paved the way for more sophisticated feature extraction and representation learning. Models such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks have shown promise in capturing temporal depen- dencies in driver behavior data, enabling more accurate and robust detection of distractions over time [22].

Moreover, the integration of attention mechanisms within deep learning architectures has emerged as a promising avenue for enhancing the interpretability and performance of driver distraction detection systems. Attention mechanisms enable models to dynamically weigh the importance of different input features, allowing for more nuanced and context-aware analysis of driver behavior [25].

Additionally, research efforts have increasingly focused on addressing the challenges of dataset imbalance and domain adaptation in driver distraction detection. Techniques such as data augmentation, transfer learning, and domain adaptation algorithms have been explored to improve model generaliza-tion and performance across diverse driving environments and conditions [26].

Furthermore, the advent of edge computing technologies has enabled real-time processing of sensor data directly within the vehicle, reducing latency and facilitating faster response times for driver distraction detection systems. Edge-based solutions offer the potential for more efficient and scalable deployment of intelligent transportation systems, ultimately contributing to enhanced road safety [28].

In summary, recent developments in driver distraction de- tection have witnessed significant progress across various fronts, including multimodal data fusion, deep learning ar- chitectures, attention mechanisms, dataset imbalance, domain adaptation, and edge computing technologies. By leveraging these advancements, researchers aim to develop more accurate, robust, and real-time systems capable of mitigating the risks associated with distracted driving and improving overall road safety.

III. DATASET DESCRIPTION

We have selected the StateFarm distracted driver detection dataset for our capstone project. The dataset was released for a 2016 competition on Kaggle [5]. This is the most widely used dataset for identifying driver distraction, having been utilised in many studies. The StateFarm dataset consists of the following ten classes:

- Safe driving (c0)
- Texting (c1)
- Talking on the phone (c2)
- Texting (c3)
- Texting (c4)
- Talking on the phone (c5)
- Drinking (c6)
- Reaching behind (c7)
- Hair and makeup (c8)
- Speaking to a passenger (c9)

The photos have had their metadata, including creation dates, erased. As a result of State Farm's careful setup of these tests—a truck towing the car through the streets—these "drivers" weren't actually operating a vehicle. A driver can only appear on one of the two test or train sets since the drivers' train and test data are divided among them. Only left- hand drive cars are included in the photo collection. There are around 2300 photographs in each class; the distribution of images by class is provided below.

TABLE I
DISTRIBUTION OF IMAGES PER CLASS

Class	Number of Images
c0: Safe driving	2400
c1: Texting – right	2250
c2: Talking on the phone – right	2300
c3: Texting – left	2350
c4: Talking on the phone – left	2350
c5: Operating the radio	2350
c6: Drinking	2350
c7: Reaching behind	2000
c8: Hair and makeup	1900
c9: Talking to passenger	2100

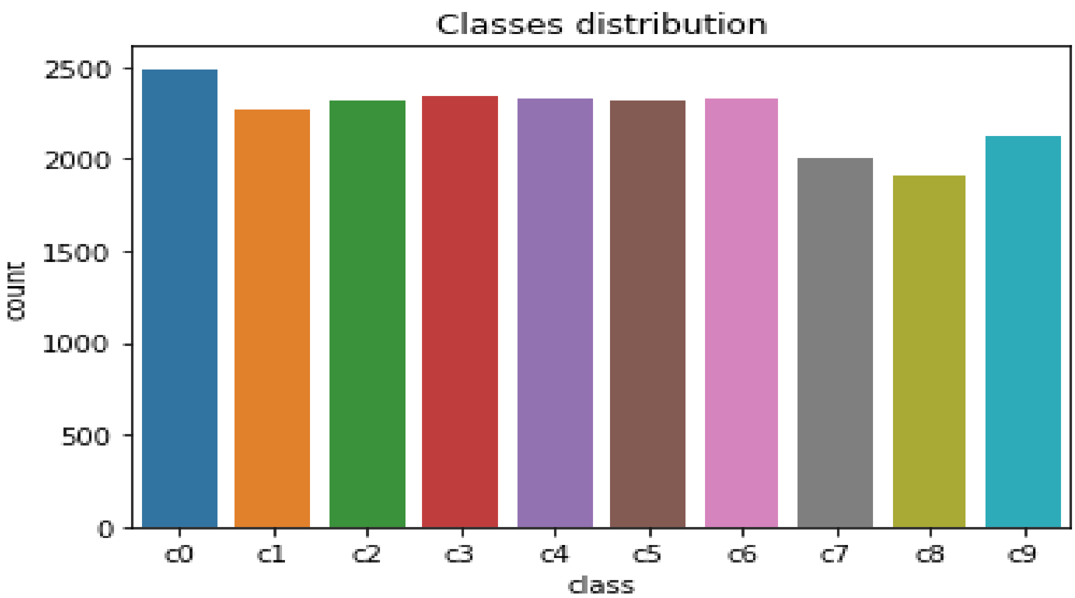

Fig. 2. Visualization of all 10 classes

IV. Material and methods

A. Visualizing and Preparing Data

We concentrate on visualising and getting the dataset ready for training our deep learning models during this stage of our technological approach. The State Farm Distracted Driver Detection dataset, which consists of photos divided into ten classes that represent various driving behaviours, was used in this investigation.

1) Data Visualization: Investigating the class distribution within the dataset is where we begin. We can learn more about the diversity and balance of behaviours included in the dataset by enumerating the classes and the labels that go with them. By visualising the distribution of classes, we may get a general idea of the dataset's makeup using visualisation techniques like Seaborn's countplot.

2) Data Preparation: After that, we divide the dataset into training and testing subsets and apply data transformations to get it ready for training. To enhance the dataset and boost model generalisation, data augmentation techniques are used, such as shrinking photos to a standard size of 400x400 pixels and performing random rotations. Next, using the random split approach, the dataset is divided into training and testing subsets while maintaining the same class distribution in both subsets.

3) Data Loading: Our deep learning models require rapid training and assessment, thus we use PyTorch's DataLoader module to generate data loaders. Batch-wise loading of photos during training and testing is made possible by data loaders, which increases memory usage and computational efficiency. To guarantee randomness and avoid model overfitting, training and testing data loaders are set up with the proper batch sizes and shuffle parameters.

4) Data Visualisation (Sample): We provide an example image (Fig. 2) from the dataset along with the label for it as a visual validation step. In addition to ensuring that the preprocessing and data loading processes have been carried out appropriately, this stage offers a qualitative insight of the dataset's
contents.

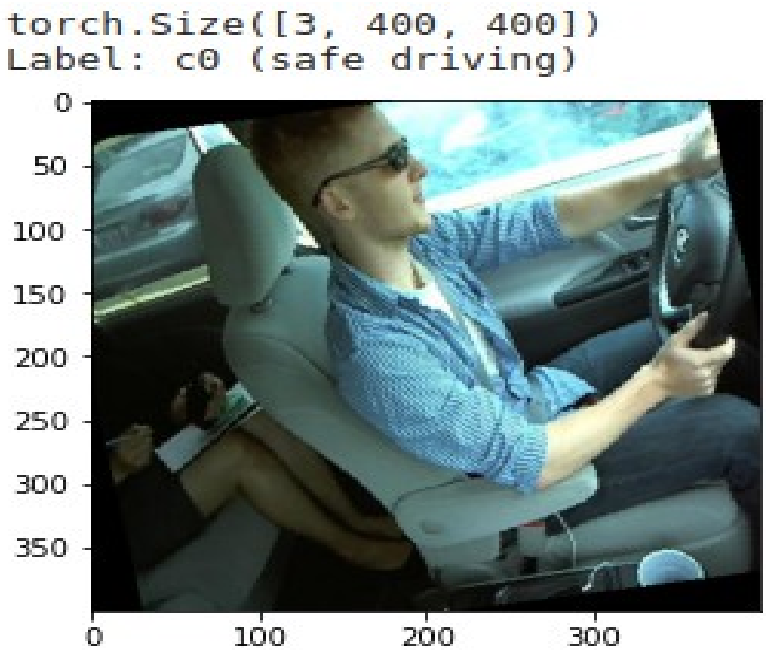

Fig. 3. Image Associated With Label

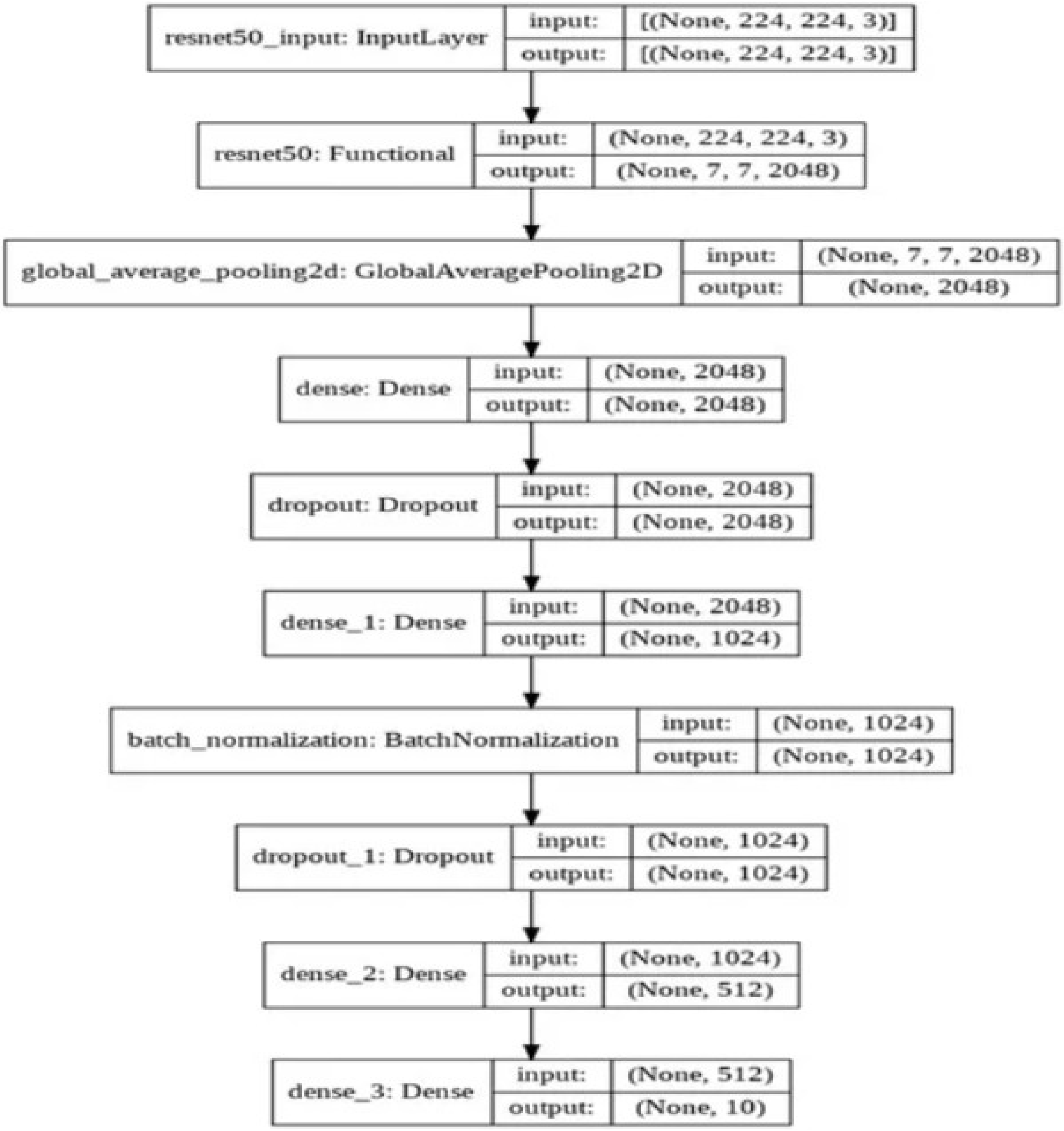

Fig. 4. RasNet50 Model Architecture

5) *Model Architecture:* The foundational model for the picture classification challenge is the ResNet50 architecture, which has been pre-trained on the ImageNet dataset. Themathematical formula for ResNet50 (Residual Network with 50 layers) can be described as follows: Given an input x, the output of the ResNet50 model can be represented as:

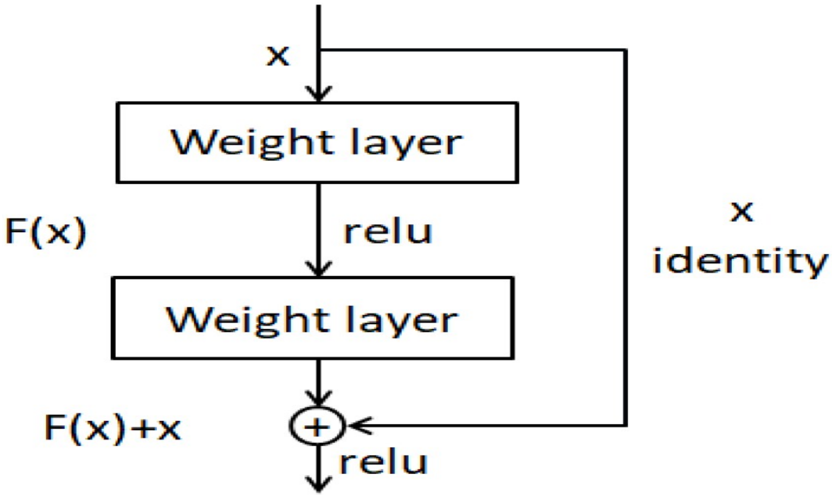

fig. 5. Residual Learning : A building Block

1) **Input:** $x^{(l)}$ where l denotes the index of the current layer.
2) **Transformation Function:** $F(x^{(l)})$
3) **Output:** $x^{(l+1)}$

Transformation Function:

1) **First Convolutional Layer:**

$$z^{(l)} = W1\ \pi\, x^{(l)} + b1$$

where W_1 represents the weights of the convolutional layer, denotes the convolution operation, and b_1 is the bias term.

2) **Batch Normalization:**

$$z(1) = z(1) - \mu / / \sqrt{\sigma 2} + \varepsilon$$

where μ and σ^2 are the mean and variance of $z^{(l)}$ respectively, and ϵ is a small constant to avoid division by zero.

3) **ReLU Activation:**

$$a^{(l)} = \max(0, z\hat{}^{(l)})$$

4) **Second Convolutional Layer:**

$$F(x^{(l)}) = W_2\ \pi\ a^{(l)} + b_2$$

where W_2 represents the weights of the second convo- lutional layer, and b_2 is the bias term.

Residual Connection:

The output of the residual block is computed as the sum of the input $x^{(l)}$ and the transformation $F(x^{(l)})$:

$$x^{(l+1)} = x^{(l)} + F(x^{(l)})$$

where f(x) represents the transformation performed by the residual blocks, and x is the input to the block. This equation signifies the main idea behind residual learning, where the identity mapping (represented by x) is added to the learned features f(x).

The goal is to learn the residual f(x) to make the optimization process easier.For the purpose of producing predictions for the ten driver behaviour classes, the last fully linked layer is altered.Stochastic gradient descent with momentum is used to op- timise the model. The model's performance on the validation set is used by a learning rate scheduler to dynamically modify the learning rate. fig.4 shows the architecture of the RasNet50 model.

6) Training Process: The training procedure is a series of iterations over the training dataset spread across several epochs. The accuracy and loss metrics are calculated at each epoch to track the convergence and performance of the model. The model's performance and capacity for generalisation are evaluated on the test set following each training epoch. The efficacy of the model is assessed by reporting the accuracy metric on the test images fig.6 and fig.7 shows the training and testing accuracy respectively.

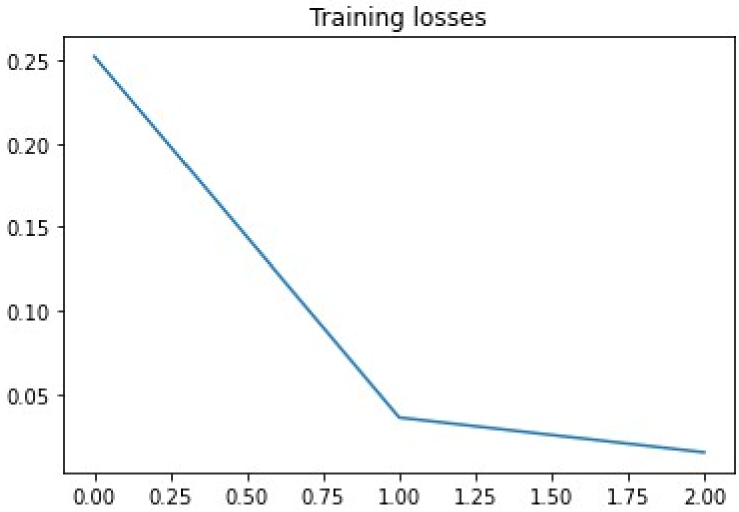

Fig. 6. Training Loss

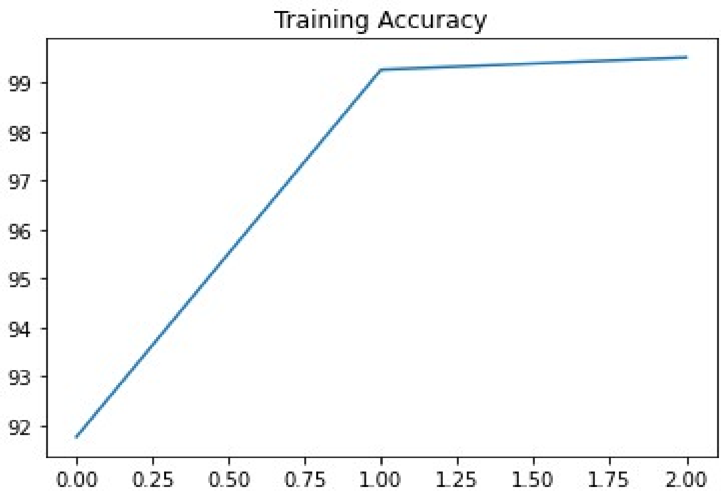

Fig. 7. Training Accuracy

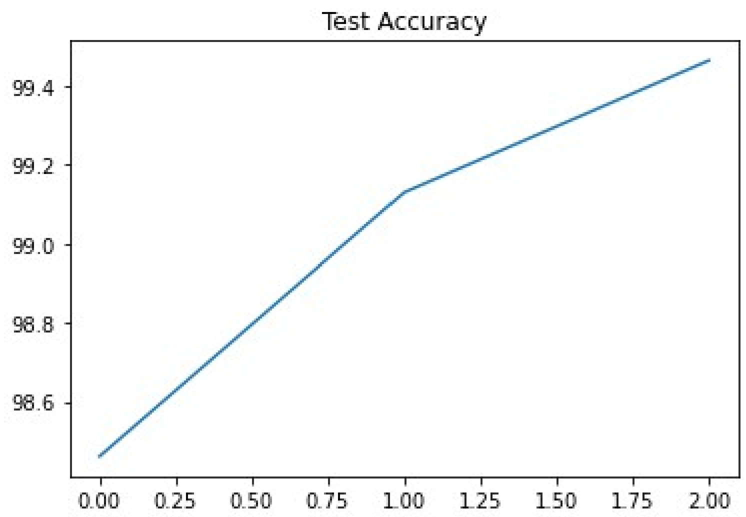

Fig. 8. Testing Accuracy

7) Transfer Learning: Utilising transfer learning, a pre- trained ResNet50 model can be loaded and its fully connected (fc) layer modified to tailor the model to a particular purpose. The model is optimised to categorise photos into ten distinct categories by substituting a new linear layer with 10 output features for the final fully connected layer.

8) Model Loading and Evaluation: The trained weights of the model are being loaded from a checkpoint file that has been saved. After loading the weights, putting the model in evaluation mode. Moving the model to the CUDA device to take use of GPU acceleration, improving inference's computational efficiency.

9) Preprocessing and Inference of Images: Preparing test images for inference by preprocessing them first. Resizing, normalisation, and data augmentation are a few examples of preprocessing procedures that are necessary to make sure the incoming data complies with the specifications of the model. Using the pre-trained ResNet50 model to perform inference on test pictures, which yields logits for each class. Class probabilities which indicate the model's confidence in its predictions, are obtained by applying softmax activation. interpreting the model's predictions by post-processing the output probabilities, which includes determining the predicted class and the corresponding confidence level.

10) Selecting Test Image: Choosing a test picture from the test dataset. Putting the chosen test picture on display to give the inference process some visual context.

V. EXPERIMENTAL RESULT

A. Experimental Settings

Intel Core i7 or above (such as an Intel Core i9), 32GB DDR4 RAM for memory, an NVIDIA GeForce GTX 1650 or higher graphics card, and a 1TB NVMe SSD for storage Figure 9. Outcome Produced by the RasnNet50 Model on the Test Picture Programme: -

Windows 11 as the operating system, PyTorch 1.10.0 as the deep learning framework, and Python 3.9.7 Libraries: Seaborn: 0.11.2, Matplotlib: 3.4.3, Pandas: 1.3.3, NumPy: 1.21.4 Google Colab is the Integrated Development Environment (IDE). Control version: Git 2.35.1

B. *Experiment*

In this section, we present the experimental results of the ResNet50 model for driver distraction detection on the State Farm Distracted Driver Dataset (SFDDD). The ResNet50 model architecture, with approximately 25.6 million param- eters, was utilized for this task. Trained on the ImageNet dataset, the model demonstrates robust performance in identi- fying distracted driving behaviors across ten distinct classes.

Table 2 provides a summary of the achieved recall, preci- sion, and F1-score for each class in the SFDDD. Notably, the ResNet50 model exhibits exceptional performance across all classes, with precision ranging from 98.74% to 100%, recall ranging from 98.71% to 100%, and F1-score ranging from 99.65% to 100%. These results underscore the effectiveness of the ResNet50 model in accurately detecting various distracted driving behaviors.

```
[0.0001, 0.0002, 0.0, 0.9973, 0.0002, 0.0, 0.0, 0.0, 0.0001, 0.002]
Predicted class: texting - left
Confidence: 0.9973
2nd answer: talking to passenger
Confidence: 0.002
```

Fig. 9. Result Generated By RasnNet50 Model On Test Image

Furthermore, Figure 13 to 15 confusion matrix offers more information about how well the ResNet50 model performs when it comes to driver distraction detection using the State Farm Distracted Driver Dataset (SFDDD).

This matrix shows the true positives, true negatives, false positives, and false neg- atives for each class of distracted driving behaviour, providing a visual representation of the classification results. Examining the confusion matrix's diagonal elements, we find that the model successfully recognises the majority of examples in each class, producing high recall and precision scores, as shown in Table 2. Moreover, the off-diagonal components draw attention to regions where the model might mistakenly label particular behaviours on occasion, providing insightful information that can be used to improve the model's accuracy and performance. All things considered, the confusion matrix offers a thorough summary of the model's classification perfor- mance, supporting the previously described metrics of preci- sion, recall, and F1-score and demonstrating the effectiveness of the ResNet50 model in tackling the problems associated with driver distracted detection.

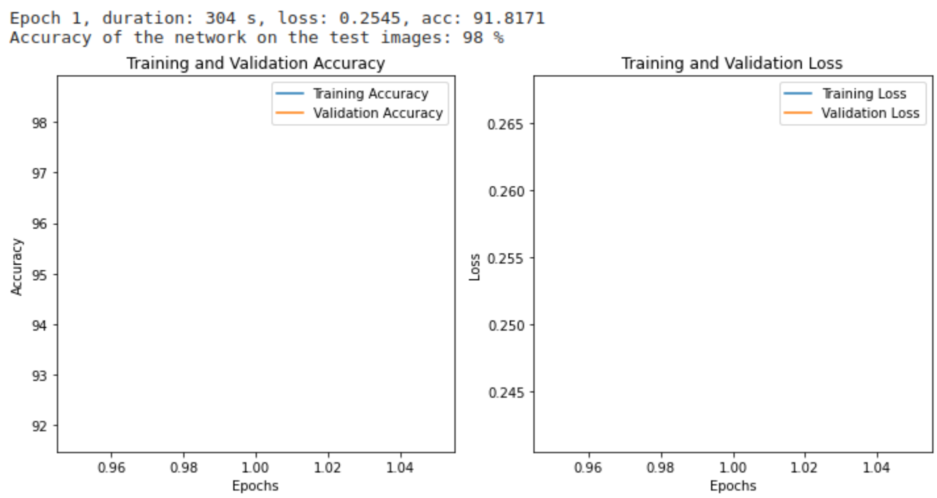

Fig. 10. Batch Size = 64 , epoch = 1

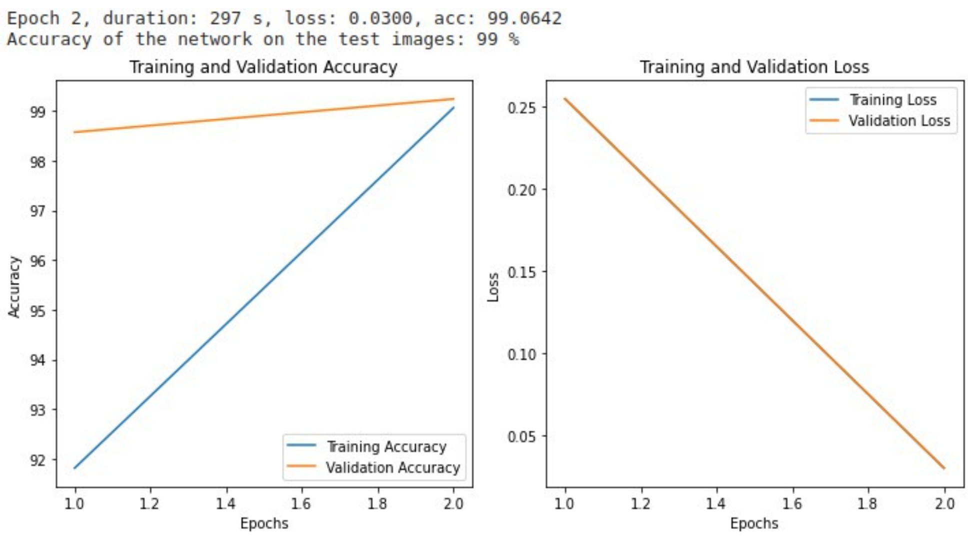

Fig. 11. Batch Size = 64 , epoch = 2

International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Fig. 12. Batch Size = 64 , epoch = 2

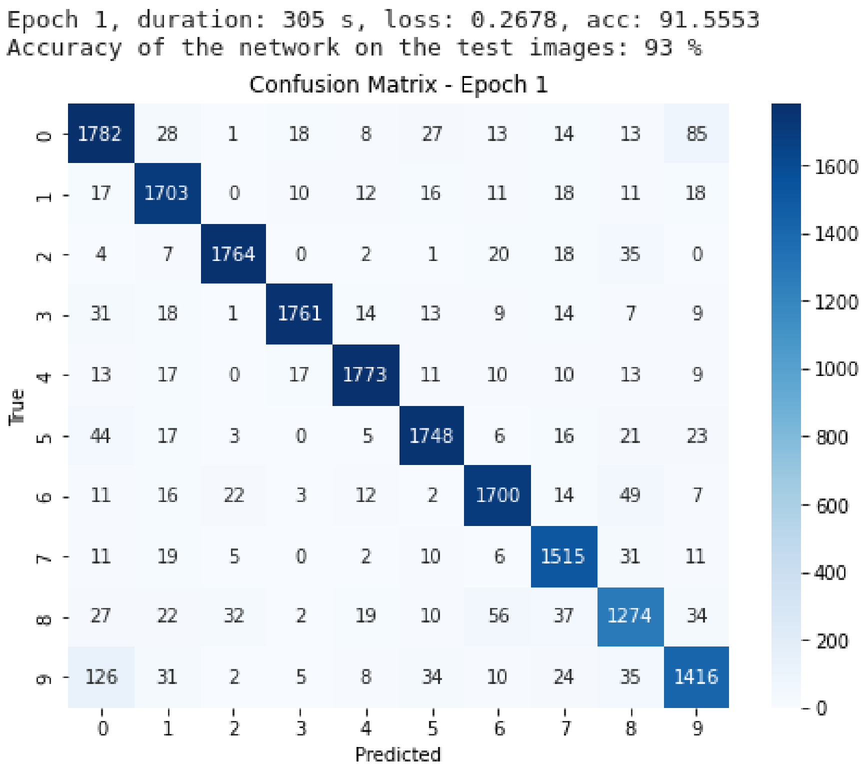

Fig. 13. Batch Size = 64 , epoch = 1

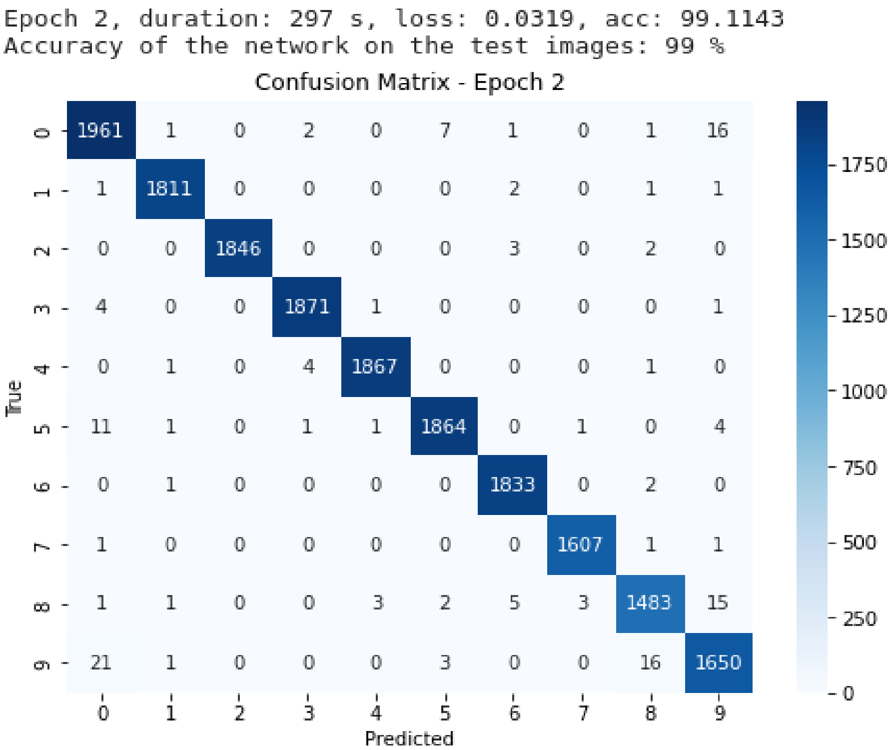

Fig. 14. Batch Size = 64 , epoch = 2

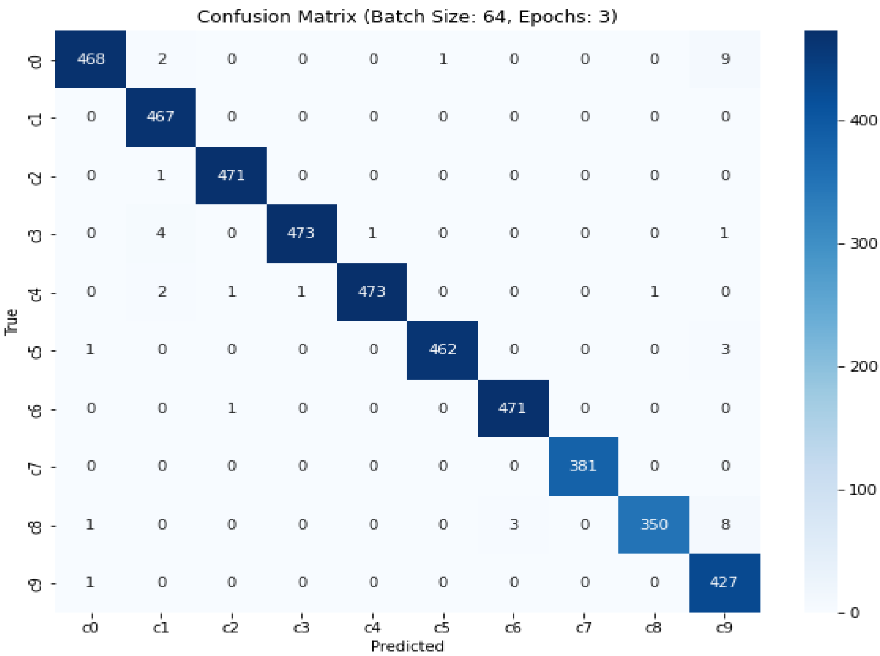

Fig. 15. Batch Size = 64 , epoch = 2

Furthermore, Table 3 presents a comparison of the proposed ResNet50 model with existing works for driver distraction classification on the SFDDD. Our proposed model outperforms previous approaches, achieving an accuracy of 99.56%. Compared to VGG, optNet-50, ConvNet, and CNN, the ResNet50 model demonstrates superior performance, highlighting its efficacy in addressing the challenges associated with driver distraction detection.

Additionally, Table 4 details the properties and accuracy metrics of the ResNet50 model. Trained on 80% of the SFDDD training data, the model achieves an accuracy of 99% on the test dataset. With an input size of 400x400 pixels and utilizing the Softmax activation function and Cross-Entropy loss, the model exhibits robustness in handling distracted driving scenarios.

In summary, the experimental results demonstrate the efficacy of the ResNet50 model in accurately detecting distracted driving behaviors. With high precision, recall, and F1-score across various classes, the model offers promising potential for real-world applications in enhancing road safety. Through its superior performance and robust architecture, the ResNet50 model represents a significant advancement in driver distrac- tion detection technology.

TABLE II

Achieved Recall, Precision, and F1-score for Each Class

Class	Precision (%)	Recall (%)	F1-score (%)
C0	100	99.56	99.78
C1	99.60	99.69	99.65
C2	99.79	100	99.89
C3	100	100	100
C4	99.78	100	99.85
C5	100	99.74	99.84
C6	99.78	99.87	99.83
C6	99.73	100	99.82
C8	100	98.71	99.90
C9	98.74	99.74	99.74

TABLE III

Comparison of Proposed Model with Published Works for Driver Distraction Classification on SFDDD Dataset

Serial No.	Model	Accuracy (%)	Publication Year
1	VGG	89.9%	2016
2	optNet-50	98%	2017
3	ConvNet	98.48%	2018
4	CNN	97%	2019
5	**ResNet50**	**99.56%**	

International Journal for Multidisciplinary Research (IJFMR)

E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

TABLE IV
RESNET50 MODEL PROPERTIES AND ACCURACY

Model Property	Description
Model Architecture	ResNet50
Total Parameters	› 25.6 million
Pretrained Weights	Trained on ImageNet dataset
Input Size	400x400 pixels
Output Size	10 classes (distracted driving behaviors)
Activation Function	Softmax
Loss Function	Cross-Entropy Loss
Optimizer	SGD with momentum
Learning Rate Scheduler	ReduceLROnPlateau, mode='max', patience=3, threshold=0.9
Image Preprocessing	Resize to 400x400 pixels, Random Rotation (10 degrees), Normalize
Training Data	Distracted Driver Dataset (80% of total data)
Testing Data	Distracted Driver Dataset (20% of total data)
Batch Size	32
Number of Epochs	3
GPU Acceleration	Utilized (CUDA)
Training Time per Epoch	› 5-6 minutes
Model Accuracy (Test Dataset)	99.56%

VI. CONCLUSION

In addition to achieving a high accuracy rate of 99% on the test dataset, the research findings underscore the robustness and reliability of the ResNet50 model in detecting distracted driving behaviors. The thorough exploration of the model's architecture and training process has provided valuable in- sights into its capabilities and limitations, paving the way for further refinement and optimization. Moreover, the successful application of deep learning techniques and GPU acceleration highlights the potential of advanced technologies in addressing complex real-world challenges.

Looking ahead, future research endeavors could focus on enhancing the model's performance through fine-tuning strate- gies, such as data augmentation and regularization techniques. Additionally, the exploration of ensemble methods and transfer learning approaches could offer opportunities to leverage the knowledge gained from related tasks and domains, thereby improving the model's generalization ability. Furthermore, the integration of attention mechanisms and real-time deployment considerations could enhance the model's responsiveness and applicability in dynamic driving environments.

By advancing the field of computer vision in the context of road safety, this research contributes to the ongoing efforts aimed at reducing the incidence of distracted driving-related accidents and promoting safer driving behaviors. The insights gained from this study serve as a foundation for future research endeavors and practical applications, ultimately benefitting society as a whole.

VII. Future Scope

1) *Model Fine-Tuning:* To possibly improve performance, further refine the ResNet50 model by tweaking hyperparame- ters or using additional data augmentation strategies.
2) *Ensemble Methods:* Investigating ways to increase ac- curacy and robustness by mixing predictions from several models, such as ResNet50 with different topologies.
3) *Transfer Learning:* Examining methods for improving a model's ability to generalise across domains by using pre- trained models on bigger and more varied datasets.*Attention Mechanisms:* By including attention mecha- nisms in the model architecture, you can help the model concentrate on important features, which may enhance per- formance and interpretability.
4) *Real-Time Deployment:* modifying the model by in- creasing its inference speed and memory footprint in order to deploy it in real-time in real-world settings, such in-car systems.
5) *Multi-Modal Fusion:* Exploring the integration of multi- ple sensory inputs, such as visual, auditory, and sensor data, to create a more comprehensive understanding of driver behavior and improve the model's accuracy in detecting distractions.
6) *Continuous Learning:* Implementing mechanisms for continuous learning, where the model can adapt and update itself over time with new data, allowing it to stay relevant and effective in evolving driving environments.

Declaration of Competing Interest

The authors state that they have no known financial conflicts of interest or personal ties that would have influenced the work presented in this study.

References

[1] Alshalfan, K., & Zakariah, M. (2021). Investigating driver distraction using deep learning techniques. Their study focuses on the utilization of advanced deep learning methods to detect instances of driver distraction.
[2] Wang, J., Yu, C., Li, S. E., & Wang, L. (2016). Development of an adaptive forward collision warning algorithm. Wang et al. devised an algorithm that provides warnings about potential forward collisions, adapting to individual driver behaviors.
[3] Baheti, B., Gajre, S., & Talbar, S. (2018). Detection of distracted driving using convolutional neural networks. Baheti, Gajre, and Talbar employed convolutional neural networks (CNNs) to identify distracted driving behaviors.
[4] Meiring, G. A. M., & Myburgh, H. C. (2015). Review of systems analyzing intelligent driving styles. Meiring and Myburgh conducted a comprehensive review of systems analyzing intelligent driving styles, examining various artificial intelligence algorithms used in this context.
[5] Aksjonov, A., Nedoma, P., Vodovozov, V., Petlenkov, E., & Herrmann, M. (2019). Detection and evaluation of driver distraction using machine learning and fuzzy logic. Aksjonov et al. utilized machine learning (ML) and fuzzy logic techniques to detect and evaluate driver distraction effectively.
[6] Suphanchaimat, R., Sornsrivichai, V., Limwattananon, S., et al. (2019). Economic development and road traffic injuries and fatalities in Thai- land. Suphanchaimat et al. investigated the association

between eco- nomic development and road traffic injuries and fatalities in Thailand, employing spatial panel data analysis.

[7] Apostoloff, N., & Zelinsky, A. (2004). Vision in and out of vehicles: Integrated driver and road scene monitoring. Apostoloff and Zelinsky discussed an integrated system for monitoring both drivers and the road scene, utilizing vision technologies.

[8] Eraqi, H. M., Abouelnaga, Y., Saad, M. H., & Moustafa, M. N. (2019). Identification of driver distraction with an ensemble of convolutional neural networks. Eraqi et al. proposed an ensemble approach using convolutional neural networks to identify instances of driver distraction.

[9] Saito, Y., Itoh, M., & Inagaki, T. (2016). Driver assistance system with a dual control scheme: Effectiveness of identifying driver drowsiness and preventing lane departure accidents. Saito et al. developed a driver assistance system incorporating a dual control scheme to identify driver drowsiness and prevent lane departure accidents.

[10] Darapaneni, N., Krishnamurthy, B., & Paduri, A. R. (2021). Autonomous car driving using deep learning. Darapaneni et al. explored the applica- tion of deep learning techniques in autonomous car driving.

[11] Berri, R. A., Silva, A. G., Parpinelli, R. S., Girardi, E., & Arthur, R. (2014). Pattern recognition system for detecting use of mobile phones while driving. Berri et al. developed a pattern recognition system to identify instances of mobile phone usage while driving.

[12] Masood, S., Rai, A., Aggarwal, A., Doja, M. N., & Ahmad, M. (2018). Detection of driver distraction using convolutional neural network. Masood et al. employed convolutional neural networks (CNNs) for detecting driver distraction.

[13] Majdi, M. S., Ram, S., Gill, J. T., & Rodriguez, J. J. (2018). Drive- net: Convolutional network for driver distraction detection. Majdi et al. introduced Drive-net, a convolutional network designed specifically for detecting driver distraction.

[14] Hssayeni, M., Saxena, S., Ptucha, R., & Savakis, A. (2017). Distracted driver detection: Deep learning vs handcrafted features. Hssayeni et al. compared the effectiveness of deep learning versus handcrafted features for detecting distracted drivers.

[15] Huff, C. (2015). Vital statistics. Huff provided vital statistics related to driver distraction and its impact on road safety.

[16] Ghadiyaram, D., Feiszli, M., Tran, D., Yan, X., Wang, H., & Mahajan, D. (2019). Large-scale weakly-supervised pre-training for video action recognition. Ghadiyaram et al. proposed a large-scale weakly-supervised pre-training method for video action recognition.

[17] Carreira, J., & Zisserman, A. (2017). Quo Vadis, action recognition? A new model and the Kinetics dataset. Carreira and Zisserman introduced a new model and dataset for action recognition to advance the field.

[18] Darapaneni, N., Krishnamurthy, B., & Paduri, A. R. (2020). Convolu- tion Neural Networks: A Comparative Study for Image Classification. Darapaneni et al. conducted a comparative study of convolutional neural networks for image classification.

[19] Mettes, P., Koelma, D. C., & Snoek, C. G. M. (2016). The ImageNet shuffle: Reorganized pre-training for video event detection. Mettes et al. introduced the ImageNet shuffle method for reorganized pre-training in video event detection.

[20] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional net- works for large-scale image recognition.

[21] Castignani, G., Derrmann, T., Frank, R., & Engel, T. (2015). Driver behavior profiling using smartphones: A low-cost platform for driver monitoring. Castignani et al. discussed driver behavior profiling using smartphones as a cost-effective solution for driver monitoring.
[22] Zhang, X., Zheng, N., Wang, F., & He, Y. (2011). Visual recognition of driver hand-held cell phone use based on hidden CRF. Zhang et al. focused on visual recognition of driver hand-held cell phone use using hidden conditional random fields.
[23] Le, T. H. N., Zheng, Y., Zhu, C., Luu, K., & Savvides, M. (2016). Multiple scale Faster-RCNN approach to driver's cell-Phone usage and hands on steering wheel detection. Le et al. proposed a multiple scale Faster R-CNN approach for detecting driver's cell phone usage and hands on the steering wheel.
[24] Seshadri, K., Juefei-Xu, F., Pal, D. K., Savvides, M., & Thor, C. P. (2015). Driver cell phone usage detection on Strategic Highway Research Program (SHRP2) face view videos. Seshadri et al. worked on driver cell phone usage detection using face view videos from the Strategic Highway Research Program (SHRP2).
[25] Koesdwiady, A., Bedawi, S. M., Ou, C., & Karray, F. (2017). End-to- End Deep Learning for Driver Distraction Recognition. Koesdwiady et al. explored end-to-end deep learning for driver distraction recognition.
[26] Pavlidis, I., Tsiamyrtzis, P., & Shastri, D. (2000). Understanding driver distraction by statistical analysis of driver's behavior during different driving conditions. Pavlidis et al. analyzed driver distraction through statistical analysis of driver behavior under different driving conditions.
[27] Huang, C., et al. (2020). HCF: a hybrid CNN framework for behavior detection of distracted drivers. IEEE Access, 8, 109335–109349. Huang et al. proposed HCF, a hybrid CNN framework for behavior detection of distracted drivers.
[28] Janet, B., Srinivasulu Reddy, U., et al. (2020). Real-Time Detection of driver distraction using CNN. In Proceedings of the 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 185–191). IEEE. Janet et al. developed a real-time detection system for driver distraction using CNNs.