UNIVERSITI TEKNOLOGI MARA

IMPLEMENTATION OF DEEP LEARNING NETWORK TO DETECT DISTRACTED DRIVING BEHAVIORS IN NIGHT DRIVING

MUHAMMAD FIRDAUS BIN ISHAK

MSc

June 2026

UNIVERSITI TEKNOLOGI MARA

IMPLEMENTATION OF DEEP LEARNING NETWORK TO DETECT DISTRACTED DRIVING BEHAVIORS IN NIGHT DRIVING

MUHAMMAD FIRDAUS BIN ISHAK

Thesis submitted in fulfilment
of the requirements for the degree of
Master of Science
(Electrical Engineering)

Faculty of Electrical Engineering

June 2026

CONFIRMATION BY PANEL OF EXAMINERS

I certify that a Panel of Examiners has met on 17 December 2025 to conduct the final examination of Muhammad Firdaus Bin Ishak on his Masters of Science thesis entitled "Implementation of Deep Learning Network to Detect Distracted Driving Behaviors in Night Driving" in accordance with Universiti Teknologi MARA Act 1976 (Akta 173). The Panel of Examiners recommends that the student be awarded the relevant degree. The Panel of Examiners was as follows:

Ahmad Sabirin Zoolfakar, PhD
Professor
Faculty of Electrical Engineering
Universiti Teknologi MARA
(Chairman)

Fazlina Ahmat Ruslan, PhD
Senior Lecturer
Faculty of Electrical Engineering
Universiti Teknologi MARA
(Internal Examiner)

Azlee Zabidi, PhD
Senior Lecturer
Faculty of Computing
Universiti Malaysia Pahang Al-Sultan Abdullah
(External Examiner)

PROFESSOR DR HJH ZURAEDA IBRAHIM
Dean
Institute of Postgraduates Studies
Universiti Teknologi MARA
Date: 15 Jun 2026

AUTHOR'S DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Teknologi MARA. It is original and is the results of my own work, unless otherwise indicated or acknowledged as referenced work. This thesis has not been submitted to any other academic institution or non-academic institution for any degree or qualification.

I, hereby, acknowledge that I have been supplied with the Academic Rules and Regulations for Postgraduate, Universiti Teknologi MARA, regulating the conduct of my study and research.

Name of Student	:	Muhammad Firdaus Bin Ishak
Student ID. No.	:	2021655474
Programme	:	Master of Science (Electrical Engineering) – CEEE750
Faculty	:	Electrical Engineering
Thesis Title	:	Implementation of Deep Learning Network to Detect Distracted Driving Behaviors in Night Driving
Signature of Student	:	……
Date	:	June 2026

ABSTRACT

Driver distraction during night driving is a leading cause of road accidents, creating the need for real-time intelligent monitoring systems. Currently, most existing distracted driving detection systems rely on visible (VIS) cameras and perform well under daylight conditions; however, their performance significantly degrades at night due to poor illumination, noise, and low contrast. Some studies have explored infrared (IR)-based systems to overcome low-light limitations, but limited comparative analysis between IR and VIS imaging under night driving conditions has been reported. In addition, many existing systems are developed and tested on high-performance computing platforms without considering deployment constraints on embedded systems. Existing models often fail to perform accurately in low-light conditions, offer limited comparison between infrared (IR) and visible (VIS) imaging, and face challenges when deployed on embedded systems due to restricted computational power. This research aims to implement convolutional neural network (CNN) models for detecting distracted driving behaviors at night using both IR and VIS image sequences. A deep learning framework was developed and trained using four CNN architectures InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet on a custom dataset consisting of six distracted behaviors collected from 44 participants. The six distracted behaviors include normal driving, yawning, nodding, texting, calling, and talking. Data augmentation was applied to improve generalization. The models were evaluated across four types of images: DAY-VIS, NIGHT-VIS, NIGHT-VIS with CLAHE preprocessing, and NIGHT-IR. Accuracy and inference time were used as the main performance metrics. The final stage involved deploying the trained models on the NVIDIA Jetson Nano to assess real-time performance in an embedded system environment. InceptionV3 achieved the highest classification accuracy on the NIGHT-VIS dataset with 85.47 % accuracy and the NIGHT-IR dataset with 89.27 %, while ResNet-50 recorded the best accuracy of 90.29 % on the NIGHT-VIS-CLAHE dataset. However, ResNet-50 demonstrated more consistent performance across different datasets and maintained real-time capability above 10 fps on the NVIDIA Jetson Nano, making it the most suitable model for deployment in night-time distracted driving detection. Thus, CLAHE implementation achieved highest accuracy for distracted driving behavior in night driving. The findings indicate that ResNet-50 combined with CLAHE preprocessing provides the most reliable performance, achieving the highest accuracy in NIGHT-VIS-CLAHE conditions, and is therefore recommended as the proposed method for distracted driving detection at night. Although InceptionV3 recorded the highest accuracy in certain sessions, ResNet-50 demonstrated more consistent high performance across different datasets. MobileNetV2, while being the fastest model with stable inference times on the NVIDIA Jetson Nano, suffered from relatively low accuracy and thus cannot be suggested as a practical solution. On the other hand, ResNet-50 achieved more than 10 fps, thus ResNet-50 can be deployed on NVIDIA Jetson Nano with acceptable performance. The proposed solution has the potential to reduce night-time road accidents and improve safety for drivers and other road users through reliable and timely behavior detection.

ACKNOWLEDGEMENT

Firstly, I wish to thank Allah SWT for giving me the strength, health, and patience to complete this research journey. I am deeply grateful for the opportunity to pursue my Master of Science in Electrical Engineering and to bring this thesis to completion despite the challenges along the way.

I would like to express my sincere appreciation and deepest gratitude to my main supervisor, Prof. Madya Ir. Ts. Dr. Fadhlan Hafizhelmi Kamaru Zaman, for his invaluable guidance, constructive feedback, and consistent encouragement throughout this research. I am also thankful to my co-supervisors, Ts. Dr. Ahmad Khushairy bin Makhtar and Ir. Dr. Ng Kok Mun, for their expertise, support, and insightful suggestions that have shaped the direction of this thesis.

My heartfelt thanks also go to all the lecturers and staff of the Faculty of Electrical Engineering, Universiti Teknologi MARA, for providing the necessary facilities, resources, and academic environment. Special thanks to my colleagues and friends who have helped directly and indirectly, especially during the data collection phase and model development process, for their continuous support and encouragement.

Finally, I would like to dedicate this thesis to my beloved father, who has supported me financially and emotionally, and to my mother, whose unwavering prayers and motivation have inspired me not to give up. This accomplishment would not have been possible without both of you. This piece of success is dedicated to you. Alhamdulillah

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviations

ADAS	Advanced Driver Assistance Systems
ARAP	As Rigid As Possible
CLAHE	Contrast Limited Adaptive Histogram Equalization
CNN	Convolutional Neural Network
DAY-VIS	Daytime Visible Spectrum Images
ECG	Electrocardiogram
EEG	Electroencephalography
EMG	Electromyography
GAN	Generative Adversarial Network
HCF	Hybrid Classification Framework
IADS	Intelligent Assisted Driving Systems
IQR	Interquartile Range
IR	Infrared
LSTM	Long Short-Term Memory
MIROS	Malaysian Institute of Road Safety Research
ML	Machine Learning
NHTSA	National Highway Traffic Safety Administration
NIGHT-IR	Nighttime Infrared Spectrum Images
NIGHT-VIS	Nighttime Visible Spectrum Images

NIGHT-VIS-CLAHE	Nighttime Visible Spectrum Images with CLAHE
Q1	First Quartile
Q3	Third Quartile
SDG	Sustainable Development Goal
SOBI	Second-Order Blind Identification
ViT	Vision Transformer
VIS	Visible
VSOL	Value of Statistical Life
YOLOv3	You Look Only Once version 3

CHAPTER 1
INTRODUCTION

1.1 Research Background

Transportation is one of our most crucial needs, substantially enhancing our capability to move from one location to another. The increase in automobile use in Malaysia is due to economic growth and a rising demand for enhanced mobility. Traffic accidents are reported daily in newspapers and on television. Most accidents are attributed to careless driving rather than vehicle defects. While transportation simplifies our lives, it also exposes us to potential disasters, including fatal accidents. Despite their undesirability, accidents are inevitabilities of life, occurring as unpredictable and unintended events or circumstances. According to the 2023 Global Status Report on Road Safety (*Global Status Report on Road Safety 2023*, n.d.) the number of road traffic deaths worldwide has slightly decreased to 1.19 million per year, indicating that some road safety initiatives are having a positive effect. However, the report emphasizes that global mobility costs remain excessively high and that immediate action is essential to meet the United Nations Decade of Action for Road Safety 2021-2030 target of halving road traffic deaths and injuries by 2030. The report, the fifth in its series, provides an overview of progress made between 2010 and 2021 and sets a baseline for further international efforts to improve road safety. Profiles are available for countries or territories that participated in the 2023 data collection survey.

According to statistics from the Ministry of Transport Malaysia (*Ministry of Transport Malaysia*, n.d.), approximately 1.35 million people die in road crashes globally each year, with an average of 3,700 fatalities occurring daily. Road traffic injuries result in substantial economic losses for individuals, their families, and nations overall. The Malaysian Institute of Road Safety Research (MIROS) reported that in 2018, the Malaysian government experienced significant economic losses for each life lost, based on the value of statistical life (VSOL). In Malaysia, road accidents claim an average of 18 lives daily, presenting a serious public health challenge that underscores the urgent need for effective policy responses. Since 2010, Malaysia has demonstrated a steady decline in road fatalities, with a notable percentage of fatalities attributed to passenger cars. According to data from the New Straits Times(*Malaysia Has 32.3*

Million Motor Vehicles, 15.8 Million Drivers, n.d.), as of December 31, 2020, Malaysia recorded 32.3 million motor vehicles, reflecting an average increase of over 1.2 million new vehicles compared to the prior year. The number of licensed drivers in Malaysia also rose by 2.86 percent, reaching 15.8 million, up from 15.3 million in 2019. One of the primary factors contributing to road accidents today is distracted driving (NHTSA, 2025). Recently, there has been increasing attention on developing driver assistance systems that monitor a motorist's behavior and provide safety recommendations. These studies typically rely on images of the driver, capturing their face, arms, and hands through a camera mounted within the vehicle (Abouelnaga et al., 2018); however, other types of data such as the driver's physical condition, audio-visual features, and vehicle information are also utilized.

According to the National Highway Traffic Safety Administration (*Distracted Driving Dangers and Statistics | NHTSA*, n.d.) distracted driving is any activity that diverts attention from the primary task of driving, including talking or texting on your phone, eating and drinking, conversing with passengers, and adjusting the stereo or navigation system. Texting is identified as the most alarming distraction; sending or reading a text message takes your eyes off the road for approximately five seconds, which at 55 mph equates to driving the length of an entire football field with your eyes closed. It is emphasized that safe driving requires full attention, and any non-driving activity increases the risk of crashing. According to Dulan Perera et al. (Perera et al., 2020) distracted driving behavior refers to any activity that diverts a driver's attention from the primary task of operating a vehicle. This includes visual distractions such as looking at something other than the road, auditory distractions like hearing unrelated sounds, manual distractions such as engaging in activities not related to driving, and cognitive distractions that involve thinking about matters other than driving. These distractions can significantly impair a driver's ability to maintain control of the vehicle, increasing the risk of accidents, injuries, and fatalities on the road.

F. Omerustaoglu et al. (Omerustaoglu et al., 2020), the authors propose a system that detects distracted driving behaviors by integrating in-vehicle sensor data with image data, aiming to enhance the overall performance of driver assistance systems. This approach addresses the critical issue of distracted driving, which is one of the leading causes of traffic accidents today. By combining various data types, including images of the driver captured by an in-car camera, the system seeks to improve its capability for generalization and adaptability in real-world driving

situations. This integration of sensor data significantly enhances detection accuracy and overall functionality, thereby potentially reducing the risks associated with distracted driving for both the driver and other road users. Moreover, deep learning techniques have been applied to identify mobile phone usage among drivers using vision systems. Xiong et al. (Xiong et al., 2019), employ Progressive Calibration Networks to establish the area for detecting phone calls.

The method for detecting a driver's cell phone using convolutional neural networks is designed to identify the mobile phone within a specified candidate area. Additionally, Rajput et al. (Rajput et al., 2020) implement the Single-Shot MultiBox Detector and the Faster Region-Based Convolutional Neural Network to detect instances of mobile phone usage. Security camera technologies are commonly utilized for surveillance and monitoring purposes to ensure safety and enhance security. These cameras are expected to function continuously and deliver high-resolution images. Frequently, security cameras are equipped with both day and night modes, which allow them to capture high-quality images under various lighting conditions. During the day mode, the cameras activate an IR filter to produce colour images. Switching to night mode disables the IR filter, enabling the camera to use infrared light to generate black-and-white images. To further enhance the infrared light in the scene, cameras may activate infrared LEDs. Furthermore, these cameras can automatically transition between day and night modes in addition to the two specified settings (Lionardi et al., 2017).

Distracted driving is a prevalent issue ((NHTSA), 2018). Recently, many technologies have been developed and tested to detect sleepiness, and the results are promising. However, existing technologies have limitations that restrict their application in vehicles (Systems, 2022). This article addresses these limitations by employing physiological data from conventional wearable sensors. The classifiers were developed to demonstrate a 99.9% accuracy rate using invasive electrodes. Researchers utilized data from diverse age groups and individual driving profiles to establish universal driver models (Li et al., 2015). Recognized indicators of fatigue and drowsiness include blinking, yawning, and changes in heart rate. In this study, image sequences are gathered as raw data through non-contact optical sensors embedded in cell phones. By analysing video streams that capture the subject’s face, the physiological factors in each image can be identified. The driver’s status is determined by evaluating various source signals together. The method's robustness is assessed

across different lighting conditions and head movements. The multi-channel Second-Order Blind Identification (SOBI) approach, which integrates multiple physiological signals more simply, shows potential for accurately detecting drowsiness (Zhang et al., 2019).

Integrating these Advanced Driver Assistance Systems (ADAS) technologies into existing vehicles after production poses significant challenges and costs, especially since it would necessitate alterations to the car's mechanical and electrical systems, making it far from practical. Without effective mechanisms to mitigate the effects of driver distractions, most vehicles on the road remain at risk. Furthermore, there has been a notable increase in the use of commercial vehicles, including delivery trucks, logistics vehicles, and ride-hailing services (*Delivery Services a Booming Industry*, n.d.).

In terms of real-time applications, the Jetson Nano is an efficient embedded platform designed for deep learning and computer vision applications. The Jetson Nano is an efficient embedded platform designed for deep learning and computer vision applications. In this paper (Basulto-Lantsova et al., 2020) highlight that the Jetson Nano demonstrates significant capabilities in object detection using the OpenCV Template Matching method. This processing power is crucial for tasks that require real-time image analysis and computer vision, making the Jetson Nano a suitable choice for various applications. Alternatively, another paper (Suzen et al., 2020) present a benchmark analysis comparing the Jetson Nano with other platforms such as the Jetson TX2 and Raspberry Pi, specifically through the use of Deep-CNN. The study indicates that the Jetson Nano effectively balances performance and cost, providing a solid foundation for machine learning projects while consuming less power compared to its counterparts. Although the Jetson Nano has been widely used for real-time computer vision applications, its performance in distracted driving behavior detection remains underexplored.

Additionally, automated behavior recognition encounters considerable difficulties when it comes to night driving (Kurian & Kumar Soori, 2023). There is a lack of sufficient data comparing IR and VIS images for detecting driver behavior at night. Comprehensive studies are necessary to assess the effectiveness of the behavior detection system in real-world environments. Alarmingly, 50% of traffic fatalities occur during nighttime hours, even though only about a quarter of all driving occurs at these times (Contrast, 2007). Night driving poses increased risks, regardless of a driver’s

familiarity with the route. According to Injury Facts, over 42,000 people lost their lives in automobile accidents in 2020 (*Driving at Night - National Safety Council*, n.d.).

Recent research has focused on utilizing artificial intelligence to mitigate distracted driving behavior, which is a major contributor to road accidents. Tarabin et al. (Tarabin et al., 2023), employ the DarkNet53 model to classify driver attention by analysing various image datasets. While achieving high accuracy, the study highlights limitations in detecting distracted driving behaviors under different lighting conditions, particularly at night. Similarly, Darapaneni et al. (Darapaneni et al., 2022) discuss an affordable and modular solution for monitoring driver distraction. However, the effectiveness of this system in low-light environments remains uncertain, as its deployment primarily targets daytime conditions. Moreover, Kurian (Kurian & Kumar Soori, 2023) demonstrates a method that tracks driver drowsiness and distraction, showing promising accuracy metrics. Nevertheless, challenges in accurately identifying distractions during nighttime driving continue to pose barriers to practical deployment (Jia et al., 2020). Collectively, previous studies have highlighted significant advancements in AI-based distracted driving detection; however, they also emphasize the need for more robust methods capable of operating effectively under low-light and nighttime conditions (Jia et al., 2020).

This research addresses the issue of distracted driving at night using deep learning, a form of artificial intelligence for learning from data. Models such as ResNet-50, InceptionV3, MobileNetV2, and GoogLeNet are utilized to identify distracted driving behaviors. The performance of the proposed model in detecting distracted driving behavior at night, as compared to daytime, is evaluated using both IR and VIS images. To validate its effectiveness, the model is implemented on the Jetson Nano as a prototype. The architecture employed is ResNet-50, a specialized CNN network. In addressing this issue, DVR technology offers a practical and cost-efficient solution, as it is simple to install in most vehicles. However, a limitation of the current DVR system is that it typically records only through the front windscreen and, in some cases, the rear or other windows. By actively alerting the driver to any distractions while driving, this technology can be an effective means of preventing driver distractions. The primary challenge of this study lies in addressing low-brightness conditions, as maintaining reliable computer vision performance during night driving is inherently difficult. 44 people are participating in the data collection process. Initially, seven driving behaviors were recorded during the data collection phase, including normal driving, yawning,

nodding off, texting while using a phone, calling while using a phone, talking to passengers, and interacting with in-vehicle features. However, the interacting with car features category was excluded from the final analysis due to overlap with other distraction behaviors. Consequently, this study focuses on six distracted driving behaviors, as described in the abstract.

The findings of this study will improve advanced driver assistance systems, which can detect driver behaviors and contribute to achieving the 2030 Agenda for Sustainable Development Goals. The eleventh SDG focuses on making cities and human settlements inclusive, safe, resilient, and sustainable, aligning with the theme of sustainable cities and communities. This study can improve road safety if we can monitor distracted driving behaviors at night. The research will have a positive impact on improving the safety of the vehicle and road users.

1.2 Motivation for This Work

The motivation for this research arises from the urgent need to address the high rate of road accidents during nighttime driving. According to the National Highway Traffic Safety Administration (NHTSA), approximately 49% of passenger vehicle occupant fatalities occur during nighttime hours, although only about 25% of total travel occurs during this period (Contrast, 2007). Although existing driver distraction detection systems have demonstrated promising performance under daytime conditions, their robustness significantly decreases under low-light environments due to illumination variability and reduced image quality (Jia et al., 2020). Therefore, this research aims to improve nighttime distraction detection using deep learning models such as ResNet-50 deployed on the Jetson Nano platform. By targeting key distracted driving behaviors such as yawning, nodding off, and texting, the aim is to develop a reliable driver assistance system for low-light conditions. This system intends to reduce accidents and improve road safety, supporting global goals for safer and more resilient communities. By enhancing detection and offering a solution for both new and existing vehicles, this research strives to make a significant societal impact by saving lives.

1.3 Problem Statement

Automated detection of driver behaviors presents significant challenges during

night driving due to low illumination and reduced visibility, which directly affect the accuracy of distracted driving behavior recognition. Behavior recognition is generally more effective during daytime conditions, where sufficient lighting allows clear capture of facial expressions and eye movements. Previous studies have largely focused on daytime or controlled lighting environments, with limited comprehensive comparison between nighttime driving behaviors captured using VIS and IR imaging modalities (Jia et al., 2020).

This study addresses several unexplored areas in the domain of nighttime driving behavior detection. Current models struggle with detecting distracted driving behaviors at night because of low light levels, which significantly affect image clarity and feature extraction performance (Shen et al., 2021). There is insufficient comparison of the effectiveness of VIS and IR images in recognizing distracted driving behaviors during nighttime driving scenarios (Jia et al., 2020). Furthermore, deploying solutions on embedded systems poses technical challenges due to limited computational power, memory constraints, and processing capability, which complicate real-time implementation and accurate detection (Kumar et al., 2017).

1.4 Research Objectives

The specific objective of this study is to implement CNN deep learning models for detecting six distracted driving behaviors during night driving, namely normal driving, yawning, nodding off, texting, calling, and talking. This objective is achieved through the following specific objectives:

a) To implement and compare multiple CNN architectures GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50, for detecting six distracted driving behaviors during night driving.

b) To evaluate the performance of the proposed models during night driving compared to day driving using IR and VIS images in terms of accuracy, confusion matrix analysis, and inference time.

c) To validate the real-time performance of the best-performing model through deployment on the NVIDIA Jetson Nano platform, focusing on inference time.

1.5 Research Question

i) How to develop a CNN that can detect driving behaviors at night?

ii) What is the difference in terms of performance between IR and VIS images in the detection of driving behaviors?

iii) How to improve the detection performance during night driving?

iv) How to validate the performance of the detector on the NVIDIA Jetson Nano?

1.6 Significance of Study

The contributions of this research are significant in several ways. This study aligns with the eleventh Sustainable Development Goal (SDG) by promoting the development of inclusive, safe, resilient, and sustainable cities and human settlements. By addressing the urgent need for disaster risk reduction, as emphasized in the 2030 Agenda for Sustainable Development, this study aims to enhance communication and sustainable urban initiatives. Additionally, the findings of this research are expected to improve advanced driver assistance systems by enabling the detection of driver behaviors, particularly during night driving. Furthermore, the research will enhance vehicle and road user safety by providing a robust solution for detecting distracted driving in low-light conditions. The implementation of the study's findings can lead to safer driving environments, ultimately contributing to the larger goal of enhancing road safety and protecting lives.

1.7 Scope and Limitations of Study

The scope of this study involves high-performance detection using CNN models that deliver accurate, real-time processing performance. Specifically, the research showcases the advantages of the ResNet-50 architecture, which strikes a balance between accuracy and processing speed in real-world applications, making it suitable for deployment. Due to the computational intensity of the AI model, careful design is necessary to ensure the onboard graphics processor can handle demanding tasks efficiently. The study focuses only on six distracted driving behaviors: normal driving,

yawning, nodding off, texting, calling, and talking to passengers. The category interacting with car features was excluded to avoid behavioral overlap and ambiguity in classification, ensuring clearer model discrimination among the selected classes. A significant challenge is maintaining optimal computer vision performance in low-brightness conditions typical of night driving. The developed model is deployed on the NVIDIA Jetson Nano platform to examine its feasibility for real-time driver monitoring applications. In this study, real-time capability refers to processing speeds exceeding 10 frames per second (fps), which is generally considered acceptable for embedded computer vision systems (Redmon & Farhadi, 2018). However, this study is subject to several limitations. The system is restricted to two imaging modalities, namely VIS and IR cameras, while other sensing approaches such as physiological signals or multi-sensor fusion methods are beyond the scope of this research. In addition, the computational capability of the NVIDIA Jetson Nano limits the complexity of the deep learning models that can be deployed. Variations in lighting conditions, driver head movements, and real-world driving environments may also affect the robustness of the detection system. Despite these limitations, the research aims to contribute to the development of intelligent driver monitoring systems by improving the understanding of distracted driving behavior detection during night driving and evaluating its feasibility on resource-constrained embedded platforms.

1.8 Thesis Outline

This section highlights the overall structure of this thesis. This thesis consists of five chapters, namely Chapter 1: Introduction, Chapter 2: Literature Review, Chapter 3: Methodology, Chapter 4: Result and Discussion, and lastly, Chapter 5: Conclusion.

Chapter 1 introduces the research background, problem statements, research objectives, research questions, significance of the study, and the scope and limitations. This chapter sets the context of the research by explaining the challenges of detecting distracted driving behaviors during night driving, the need for Convolutional Neural Network (CNN)-based solutions, and the role of infrared (IR) and visible (VIS) imaging in improving detection accuracy.

Chapter 2 reviews related works and previous research in distracted driving detection, focusing on CNN architectures, IR and VIS imaging applications, and image enhancement techniques such as Contrast Limited Adaptive Histogram Equalization

(CLAHE). This chapter also discusses performance comparisons between daytime and nighttime detection, identifies gaps in existing research, and justifies the need for this study.

Chapter 3 details the methodology undertaken in this research, including dataset collection from 44 participants, data preprocessing, and image augmentation methods. It describes the CNN architectures used, including GoogLeNet, InceptionV3, MobileNetV2, ResNet-50, the training and testing procedures, evaluation metrics, and the deployment of models on the NVIDIA Jetson Nano for real-time performance testing.

Chapter 4 presents the results of the experiments, including training and validation accuracy/loss graphs, confusion matrix analyses, and precision, recall, and F1-score metrics for each dataset. Performance comparisons are made between DAY-VIS, NIGHT-VIS, NIGHT-VIS-CLAHE, and NIGHT-IR datasets, with detailed discussions on the impact of CLAHE enhancement and infrared imaging on classification accuracy.

Chapter 5 concludes the thesis by summarizing the main findings, outlining the contributions of the research, and addressing its limitations. Recommendations for future work are provided, focusing on further improving distracted driving detection under low-light conditions and optimizing model performance on embedded systems.

.

CHAPTER 2
LITERATURE REVIEW

2.1 Introduction

Sleep and rest have been ignored in today's fast-paced environment, especially by the economically active population, for several reasons. Today's youth in most nations are severely sleep-deprived. Drowsy driving is one of the most significant issues this can lead to. Drowsy drivers put other drivers and pedestrians on the road in danger in addition to doing themselves harm. Drowsy driving has recently escalated to a major problem in both industrialized and developing nations. Drowsy driving detection systems must be created immediately to keep up with the technology's rapid advancement (Bhaskar, 2018). Drivers must pay attention to the road and have quick reflexes to respond to unexpected situations in the transportation sector. As a result of working long hours, drivers are more likely to become fatigued. This causes them to become less vigilant, which has increased the number of traffic accidents. Fatigue contributes to around 30% of modern-day traffic accidents. A tired driver has a higher risk of losing control of the vehicle than one with a blood alcohol level of 25%, according to studies. Reduced alertness causes a reduction in awareness, identification, and vehicle control abilities, which puts both their own and others' lives in grave danger (Pinto et al., 2019).

The economic impact of drowsy driving is substantial, with costs associated with medical expenses, productivity losses, and property damage running into billions annually. Social dynamics, such as increased work hours and reliance on stimulants to stay awake, further exacerbate the issue of sleep deprivation. Public health campaigns and policy initiatives aimed at mitigating sleep deprivation and drowsy driving remain crucial yet underfunded aspects of combating this growing problem.

2.2 Introduction to Driving Behavior Detection

To effectively tackle this issue, deep learning has emerged as a promising solution. Figure 2.1 shows that deep learning is a branch of machine learning that uses neural networks to teach computers to perform tasks such as classification or regression

directly from data, including images, text, or sound. These models can achieve state-of-the-art accuracy, often exceeding human-level performance. Deep learning models are trained using large sets of labeled data, and they typically learn features directly from the data without the need for manual feature extraction. The first artificial neural network was theorized in 1958, but it wasn't until the 2000s that computing power advanced enough to make deep learning viable on a large scale. High-performance GPUs, with their parallel architecture, are particularly efficient for deep learning tasks. When combined with clusters or cloud computing, these resources allow researchers to train deep learning networks in hours rather than weeks (*Deep Learning - MATLAB & Simulink*, n.d.).

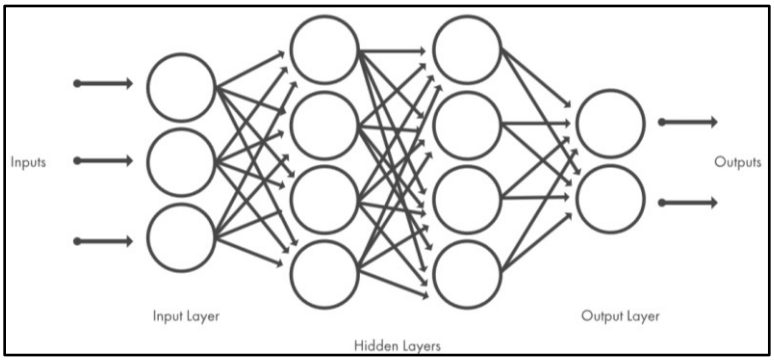


Figure 2.1 Viewing a Typical Neural Network Architecture (*Deep Learning - MATLAB & Simulink*, n.d.)

This article reviews some significant data on car accidents caused by midnight driving in the US. The National Highway Traffic Safety Administration's (NHTSA) publication Traffic Safety Facts 2018 is the exclusive source of the statistics in this article, unless another source is mentioned. There were 17,335 fatal nighttime collisions in 2018. More than 40% of deadly nighttime traffic collisions involved drunk driving. More than 55% of fatal collisions took place between 12 and 3 a.m. had at least one intoxicated driver in it (*Nighttime Driving Accident Statistics in the U.S. | Injury Claim Coach*, n.d.).

Current detection technologies include ADAS, which often rely on basic sensors and heuristic algorithms but lack real-time adaptability. Innovations in deep learning techniques offer potential improvements in accuracy and adaptability, making them well-suited for analyzing complex driving behaviors dynamically. The fusion of sensor data (from cameras, lidar, and GPS) in conjunction with deep learning algorithms can

significantly enhance the precision of behavior detection systems. There is an increasing body of research focusing on the early identification of behavioral markers indicative of fatigue and distraction, paving the way for preventive measures.

The detection of driving behavior is an evolving field that plays a critical role in enhancing road safety by identifying high-risk behaviors such as drowsiness and distraction. Current technologies, including Advanced Driver Assistance Systems (ADAS), provide foundational support but often lack the real-time adaptability necessary for the dynamic analysis of driving data. Recent innovations in deep learning leveraging neural networks and large-scale datasets have demonstrated significant improvements in both accuracy and flexibility across various applications (Lecun et al., 2015). These models outperform traditional methods by analyzing complex patterns directly from raw data, making them highly suitable for real-time applications. Moreover, the integration of sensor data from cameras, LiDAR, and GPS with deep learning algorithms further enhances the precision and responsiveness of behavior detection systems. This technological synergy holds great potential for the early identification of fatigue and distraction indicators, paving the way for proactive preventive measures and significantly improving the overall safety of transportation systems.

2.3 Conventional Methods to Detect Distracted Driving Behavior

Since there are significant differences in Electroencephalography EEG data between various people, training a subject-independent EEG classification model is difficult. A subject-independent EEG classification model that anticipates target labels without considering subjects, hence avoiding the cost issue. reducing the mutual information between the target and subject labels to learn subject dependency. Our model has two branches for predicting the labels for the target and subject after the feature embedding module. The feature embedding module and the subject prediction module are trained in opposition, encouraging the feature representation to be encoded subject-invariantly. The EEG-based drowsy driving detection task necessitates consistent results across subjects for practical applications. The examination of the SEED-VIG dataset shows that this strategy performs meaningfully in terms of accuracy and individual differences (Hwang et al., 2021). The most crucial component of current advanced driver aid systems is lane detection when driving ADAS. Conversational tasks

and distracted driving correlation studies are investigated. With the combination of technology and automobiles, it is easy for drivers to become preoccupied with unimportant activities like using a cell phone, using a navigation system, or listening to podcasts. The cognitive relationships between distracted driving behavior were examined utilizing EEG data, a linear support vector machine for classification, and discussion with fellow passengers as a secondary driving activity. A salient feature set for identifying driver distraction behavior may be provided by EEG characteristics. SVM classification resulted in a 5% increase in accuracy rating. Cognitive information is a reliable factor for categorizing distracted driving behavior (Bhattacharya & Bernadin, 2020).

In addition, Ju (Ju & Li, 2024) presented a survey on EEG-based driver state and behavior detection for intelligent vehicles underscores the significance of driver states in ensuring safety on the roads. This study provides a comprehensive overview of methods utilized in EEG-based intelligent assisted driving systems (IADS), detailing algorithms for signal acquisition, preprocessing, signal enhancement, feature calculation, feature selection, classification, and post-processing. The survey emphasizes the necessity of combining driver state detection and driver behavior detection for enhanced accuracy and reliability in real-world applications. As EEG research progresses, hybrid brain-computer interface applications are explored, representing a promising direction for future system improvements. Regarding driver attention assessment, recent research conducted by Affanni et al. (Affanni & Najafi, 2022), highlights the effectiveness of measuring blink rates extracted from EEG signals. In this paper, the authors describe a study involving ten volunteers driving on a simulator in three different configurations: manual driving, prudent autonomous driving, and aggressive autonomous driving. The findings demonstrate that manual driving is more mentally demanding than autonomous driving, regardless of the aggressiveness of the algorithm. This conclusion is supported by EEG data, indicating that blink rate serves as a reliable indicator of driver attention levels, thereby revealing crucial insights into the mental demands posed by various driving conditions.

On top of that, recent research highlights the assessment of drivers' cognitive stress responses using EEG and electrocardiogram (ECG) signals during real-traffic experiments with an electric vehicle. This research aims to demonstrate how different driving environments significantly influence stress responses, emphasizing the importance of this factor in the context of ADAS and autonomous driving systems. An

entropy EEG data analysis was conducted to quantitatively measure stress, and the study investigates differences in individual cognitive responses to driving stress (Noh et al., 2019). Additionally, a fatigue driving detection system based on EEG signals has been developed to effectively monitor driver attentiveness. This system utilizes EEG and EMG signals captured by a TGAM chip, allowing for comprehensive data analysis to determine the driver's fatigue state by assessing parameters such as attention, meditation, blink frequency, and power spectral density. If a driver is identified as fatigued, the system triggers an alarm, prompting the driver to recharge and restore focus, thereby reducing the likelihood of traffic accidents. This approach highlights the significant potential of EEG-based methods in real-time applications for driver fatigue detection (Feng, 2021).

The ECG provides a trustworthy signal that reliably, inexpensively, and with little intrusion, can define physiological changes. Aspect of ECG signal processing used to forecast driver distraction in the naturalistic driving experiment, which involved eight volunteers aged 24 to 45, two different methods caused distraction: 1) a phone call and 2) an active dialogue between the driver and the passenger. a powerful Wavelet Packet Transform frequency subband analysis to pinpoint the effects of distracting factors. Due to the high dimensionality of the WPT-generated space, Linear Discriminant Analysis was used to reduce the dimensionality of the feature space while maintaining the prediction model's capacity to discriminate. High potentials for detecting normal vs. distracted driving conditions were shown by the WPT transform in conjunction with the linear discriminant dimensionality reduction (Deshmukh & Dehzangi, 2017). Utilizing physiological, behavioral, and vehicle signals, a monitoring system is created. During the on-road driving session that eight subjects participated in, motion signals (accelerometer and gyroscope), ECG, galvanic skin response, and CAN-Bus signal were recorded. These signals were then used to extract features. To detect driver preoccupation, the feature space from each signal was individually assessed. The evaluation of the multimodal feature space was done to increase recognition accuracy. To create the best multi-modal feature space, feature selection techniques were used because the high dimension of the fused feature space is plagued by dimensionality (Dehzangi & Galster, 2018). Although EEG and ECG-based approaches provide valuable physiological insights into driver states, these methods often require wearable sensors attached to the driver’s body, which may cause discomfort and reduce practicality for real-world driving environments. In addition, physiological sensing

systems involve complex signal acquisition and preprocessing procedures, making real-time implementation in commercial vehicles more challenging. Therefore, non-intrusive vision-based approaches using cameras have gained increasing attention as a more practical solution for driver monitoring systems.

The conventional methods for detecting distracted driving behaviors primarily rely on physiological measurements such as EEG and ECG, which provide valuable insights but face challenges like inter-individual variability and the complexity of real-time implementation. These techniques utilize various features, including blink frequency and cognitive responses, to identify states of fatigue and distraction. However, advances in deep learning have transformed this field by offering powerful models capable of processing large datasets with greater accuracy and adaptability. Deep learning approaches can integrate multiple data sources, including EEG, ECG, and external sensors, to offer a more comprehensive understanding of driver behavior. These models outperform traditional methods by automatically extracting meaningful features and learning complex patterns without extensive manual intervention.

In contrast to conventional signal-processing approaches that require handcrafted feature extraction and multi-stage processing pipelines, deep learning models enable end-to-end learning, thereby reducing computational overhead and improving inference efficiency. Once trained, optimized deep learning models can be deployed for real-time inference using GPU acceleration and parallel processing architectures, making them more suitable for embedded and intelligent transportation applications. While conventional methods remain important, the adoption of deep learning techniques marks a significant advancement toward real-time, robust, and scalable distracted driving detection systems that contribute to safer road environments.

2.4 Vision-Based Methods to Detect Distracted Driving Behavior

Huang et al. (Huang et al., 2020) proposed a cooperative pre-trained model that integrates ResNet-50, InceptionV3, and Xception to extract driver behavior features based on transfer learning to improve the accuracy of the driving activity detection system. The pretrained models extract independent features that enable the framework to capture richer behavioral information from large-scale datasets. In addition, the Hybrid Classification Framework (HCF) employs fully connected layers to detect anomalies and hand motions related to distracted driving behaviors. An enhanced

dropout strategy is also introduced to prevent the proposed HCF from overfitting to the training dataset. The class activation mapping technique is applied to highlight ten classes of typical distracted driving behaviors. The state farm distracted driver detection dataset has eight classes: chatting on the phone, texting, regular driving, using the radio, being inactive, talking to a passenger, glancing behind, and drinking, all performed by 26 participants. In contrast to the 26 participants used in (Huang et al., 2020), this study involves 44 participants to improve dataset diversity and enhance model generalization. A larger number of participants helps reduce inter-individual variability bias and improves the robustness of the trained CNN models, particularly under nighttime driving conditions. Furthermore, the participant size was determined based on the research allocation plan, which required a minimum of 30 recruited drivers, with additional participants included from the laboratory research team to strengthen data variability and class balance. As the foundation of EfficientDet, the transfer values of the pertained model EfficientNet are utilized. For accurate forecasts and cutting-edge outcomes, the EfficientDet model analyses the photos to identify the objects participating in these distracting activities and the region of interest for the body parts. For detection purposes, the Efficientdet model implemented five variants: Efficientdet (D0-D4). Yolo-V3 and Faster R-CNN were used to compare the best Efficientdet version (Sajid et al., 2021).

DarkNet is a deep learning framework commonly used as the backbone architecture for object detection models such as YOLO. In the proposed system, DarkNet is utilized to process visual data and support the detection of distracted driving behaviors. The framework integrates a data collection module and an analytics engine that utilizes machine learning (ML) techniques to classify driving behaviors based on sensor inputs. Data from an internal camera and an inertial measurement unit (IMU) mounted inside the vehicle are used as input sources (Sarker, 2021). The system employs the You Only Look Once version 3 (YOLOv3) convolutional neural network (CNN) to automatically extract facial features with sufficient accuracy and speed. The features extracted by YOLOv3 are subsequently organized into temporal sequences representing driver behaviors such as eye blinking and yawning durations. These temporal sequences are then used as input for a Long Short-Term Memory (LSTM) neural network to learn and recognize the temporal patterns of distracted driving behaviors. The dataset used for the LSTM training process is generated from the features extracted by the YOLOv3 CNN. These features are structured as two-

dimensional temporal sequences consisting of eye blinking frequency and yawning duration, which allow the LSTM model to capture temporal dependencies in driver behavior. The dataset also considers environmental disturbances such as lighting conditions and variations in the driver's head position. A multi-threaded framework is implemented to enable the CNN and LSTM models to operate simultaneously, allowing real-time distracted driving behavior detection (Faraji et al., 2021). A technique based on innovative multi-feature fusion that uses conventional image processing and heart rate variability to identify weariness and sleepiness. The first decision is performed by an LSTM using the features collected by InceptionV3 to process the video sequence for recognition. The suggested method performs initial feature extraction using InceptionV3 CNN. The LSTM eliminates static distortions while delivering precise and coherent sequence recognition. After the extracted features are fused, blood volume pulse signals derived from heart rate variability are incorporated to support the final classification of driver fatigue states. This paper tested the viability of our technology by evaluating its ability to detect driver weariness, as fatigue recognition is typically used to monitor driver drowsiness (Zhao et al., 2020). Christopher et al. (Streiffer et al., 2017) proposed a multimodal distracted driving detection framework that integrates a DarkNet-based CNN with recurrent neural networks (RNN) and inertial measurement unit (IMU) signals. The system combines visual features extracted from in-vehicle cameras with motion data captured by mobile device sensors to classify six distracted driving behaviors. By leveraging transfer learning and sequence modeling, the framework captures both spatial and temporal behavioral patterns. The study reported an overall accuracy of 87.02% on a custom driving dataset. However, the implementation faced limitations related to edge device constraints and generalization capability. Unlike the multimodal approach in (Streiffer et al., 2017), the present study focuses primarily on vision-based deep learning under low-light conditions, emphasizing robust feature extraction using VIS and IR imaging modalities for nighttime distracted driving detection.

Likewise, research on unsafe driving behaviors in taxi drivers shows new methods to improve driver monitoring. This study focused on detecting driver fatigue and ensuring safety during the recent epidemic by developing a system that registers faces and monitors signals like EEG and electromyography (EMG). The system uses a MobileNetV2 algorithm to check if drivers are wearing masks and combines MTCNN and Face-Net for recognizing drivers' faces to enhance monitoring. Moreover, this

research uses a neural network model to merge 12 signals related to fatigue and create a complete fatigue monitoring system. This system is more accurate than traditional monitoring methods and demonstrates the potential to combine various physiological signals in real-time to improve driver safety and performance (X. Wu et al., 2020). Although fatigue detection is closely related to distracted driving, this study does not treat fatigue as a standalone category. Instead, fatigue-related behaviors such as yawning and nodding off are included as observable distracted driving behaviors within the vision-based detection framework. This decision was made because the proposed system focuses on non-contact optical sensing (VIS and IR imaging), whereas fatigue monitoring systems based on EEG and EMG require physiological sensors that are beyond the scope of this study. Therefore, the research emphasizes visually detectable distraction behaviors rather than physiological-state-based fatigue assessment. Furthermore, a deep learning-based driver activity recognition system has been developed that focuses on assessing driver behaviors to enhance safety. This research identifies seven common driving activities, including normal driving, right mirror checking, rear mirror checking, left mirror checking, using in-vehicle radio devices, texting, and answering mobile phones, thereby classifying the first four as normal driving tasks and the latter three as distractions. The system utilizes deep convolutional networks to process experimental images captured using a low-cost camera during naturalistic data collection involving ten drivers. The activity recognition models achieved an average accuracy of 81.6% using AlexNet, while comparison models such as GoogLeNet and ResNet-50 achieved 78.6% and 74.9% accuracy, respectively. Notably, the binary classification system identified whether the driver was distracted, achieving an accuracy rate of 91.4%. This underscores the advantages of using the deep learning approach for effective driver monitoring and the potential for enhancing road safety (Xing et al., 2019).

Next, the study on driver activity recognition aims to improve driver monitoring by employing deep learning techniques to assess driver behavior. This research identifies seven common driving activities, including normal driving, right mirror checking, left mirror checking, using in-vehicle radio, texting, and answering a mobile phone. The integration of deep learning in driver activity recognition demonstrates the system's capability to assess driver behaviors effectively. Findings illustrate that a deep learning approach efficiently recognizes these activities, establishing a foundation for distinguishing between normal tasks and distractions. Recent research employs

MobileNetV2 and DenseNet to accurately identify these activities, with MobileNetV2 showing particularly strong performance. The system utilizes a low-cost camera for image capture and applies algorithms to segment the data before training convolutional neural networks, thereby underscoring the importance of utilizing deep learning in driver monitoring systems to enhance safety features and improve road safety (Goel et al., 2023). In comparison to the seven activities identified in (Goel et al., 2023), this study focuses on six distracted driving behaviors: normal driving, yawning, nodding off, texting, calling, and talking to passengers. The behaviors selected in this work emphasize activities that are directly associated with distraction and fatigue risks rather than routine driving tasks such as mirror checking or radio adjustment. Mirror checking, for instance, is considered a necessary driving maneuver rather than a distraction behavior. Therefore, this study prioritizes high-risk behaviors that significantly divert driver attention from the primary driving task, particularly under nighttime driving conditions

Similarly, a study on driver drowsiness detection explores using NASNet Mobile, MobileNetV2, and EfficientNetB0 to identify driver fatigue through visual cues. This research highlights the importance of recognizing driver behavior as a critical driving safety factor. It identifies several common activities that indicate normal driving or distraction, such as texting, using in-vehicle devices, and checking mirrors. The study demonstrates that these models can accurately identify distracted driving activities, with MobileNetV2 showing powerful results. This underscores the value of deep learning techniques in developing advanced systems for detecting driver drowsiness and enhancing overall road safety (W. Wu et al., 2022).

2.4.1 Overview of Deep Learning Architectures

Deep learning has revolutionized how complex patterns are recognized in data, and several architectures have been pivotal in this advancement. Understanding these models is crucial in the context of detecting distracted driving behavior, particularly as they were all applied in the experiments discussed in this section.

Transitioning to ResNet-50, this convolutional neural network developed by Microsoft Research in 2015 is renowned for its effectiveness in image classification tasks. Figure 2.3 shows the model architecture of ResNet-50, which is named after its innovative use of "Residual Network" design, incorporating skip connections or

residual blocks to address the vanishing gradient problem common in deep neural networks. This design allows the model to learn residual functions that map inputs to outputs, enabling training on significantly deeper networks without performance degradation. The architecture comprises several components: convolutional layers for feature extraction, identity blocks that retain information by summing inputs with outputs, and convolutional blocks that include a 1x1 convolution to optimize filter counts before the main 3x3 convolution. Additionally, batch normalization and ReLU activation functions are applied within the convolutional layers to enhance feature extraction, while max pooling reduces the spatial dimensions of the feature maps. The final fully connected layers use a softmax activation function to produce class probabilities. This architecture allows ResNet-50 to achieve state-of-the-art results on large datasets, making it a powerful model for the distraction detection system studied in this research (Quach, 2024). The residual learning mechanism enables the network to capture deeper feature representations without suffering from vanishing gradient issues, which contributes to improved recognition of subtle driver behaviors in complex driving environments.

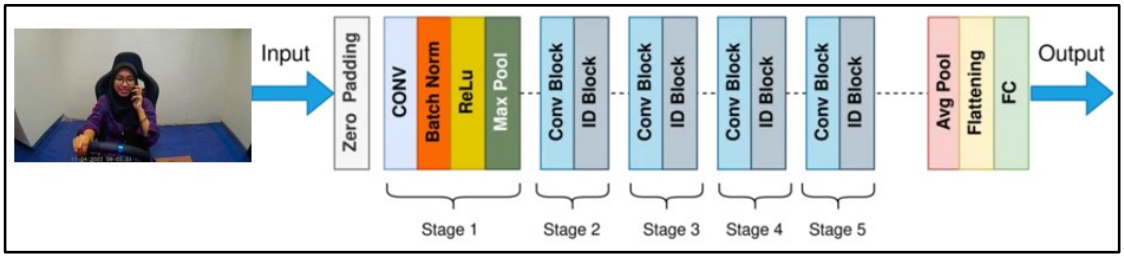


Figure 2.2 Model Architecture of ResNet-50 (Quach, 2024)

InceptionV3 is a CNN within the Inception family, known for its efficient design that enhances both accuracy and computational efficiency in image recognition tasks. It employs factorized convolutions, label smoothing, and batch normalization to significantly improve performance. As a pre-trained model utilizing transfer learning, InceptionV3 leverages learned weights from the ImageNet dataset and contains over 20 million parameters, having been trained using advanced hardware. The architecture features both symmetrical and asymmetrical building blocks, incorporating various operations such as convolution, average pooling, max pooling, concatenation, dropout, and fully connected layers. Batch normalization stabilizes and accelerates training, while classification is conducted via the Softmax function to produce class probabilities.

With 48 layers, InceptionV3 effectively captures complex image patterns, making it an essential component in the current research. Figure 2.2 illustrates the model's architecture and components, and users can access a pre-trained version that has been trained on over a million images from ImageNet, facilitating enhanced accuracy in various image classification tasks (Szegedy et al., 2016). The use of factorized convolutions and multi-scale feature extraction allows InceptionV3 to efficiently capture complex spatial patterns in driver behavior images, improving detection performance under varying visual conditions.

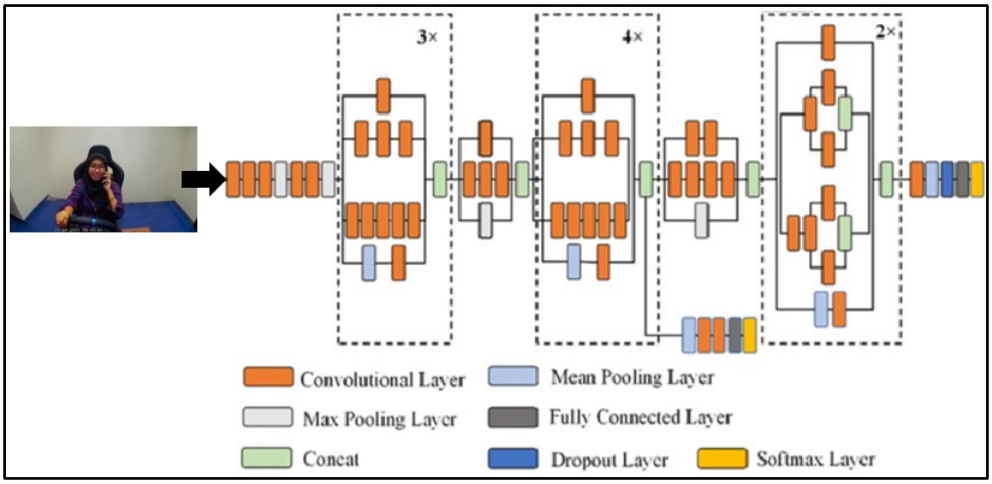


Figure 2.3 Model Architecture of InceptionV3 (Szegedy et al., 2016)

The use of MobileNetV2 further complements the research focus as it is a lightweight CNN specifically designed for mobile and embedded vision applications, recognized for its efficiency in reducing the number of parameters and operations without sacrificing accuracy. Figure 2.4 shows the model architecture of MobileNetV2, a 53-layer deep network that employs inverted residuals and linear bottlenecks to perform effectively in real-time scenarios on resource-constrained devices. Based on the concept of depth-wise separable convolutions, MobileNetV2 applies a single filter to each input channel and combines these outputs through point-wise convolutions. Trained on over a million images from the ImageNet database, it classifies images into 1,000 object categories, generating rich feature representations of various objects, including keyboards, mice, and animals. The architecture incorporates key features, including the bottleneck design and squeeze-and-excitation blocks, thereby balancing model size with competitive accuracy. This efficient deployment in diverse real-time applications further enhances the distracted driving detection system being developed

(Kulkarni et al., 2023). The depthwise separable convolution and inverted residual structure significantly reduce computational complexity, making MobileNetV2 suitable for deployment on embedded systems such as the NVIDIA Jetson Nano used in this study.

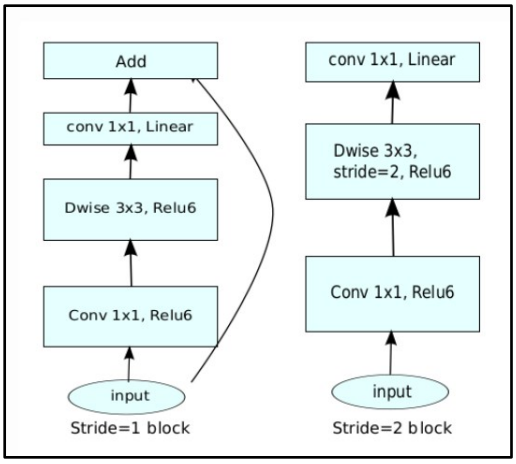


Figure 2.4 Model Architecture of MobileNetV2 (Kulkarni et al., 2023)

GoogLeNet, also known as Inception-V1, is a groundbreaking convolutional neural network introduced by researchers at Google in 2014 that significantly advanced the field of deep learning. Figure 2.5 shows the model architecture of GoogLeNet, a 22-layer deep network that emphasizes computational efficiency while achieving exceptional accuracy. A key innovation of GoogLeNet is the inception module, which facilitates the parallel processing of input data at multiple scales, enabling the network to capture features at different resolutions efficiently. Each inception module consists of several convolutional layers with varying filter sizes arranged in parallel, followed by concatenation and pooling, allowing the model to learn both large-scale and small-scale features. This design enhances feature extraction capabilities and addresses challenges such as vanishing gradients by stabilizing the training process through its multi-scale learning approach. As a result, GoogLeNet has established itself as a vital model in the development of sophisticated CNN architectures and has been applied in this research to strengthen distracted driving behavior detection efforts (Kaur, 2024). The multi-scale processing capability of the inception modules allows the network to capture both fine-grained and large-scale visual cues, which is beneficial for identifying diverse distracted driving behaviors.

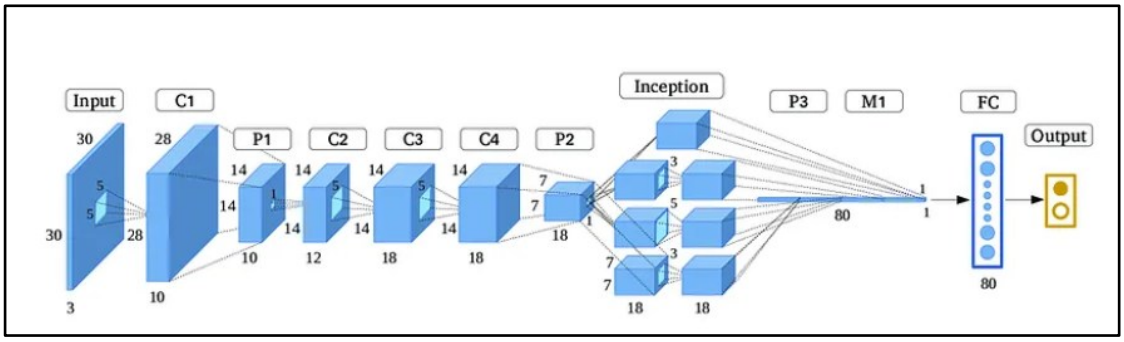


Figure 2.5 Model Architecture of GoogLeNet (Kaur, 2024)

Vision-based methods for detecting distracted driving behaviors have greatly benefited from the application of advanced deep learning models, including InceptionV3, ResNet-50, MobileNetV2, and GoogLeNet. These architectures are essential for extracting and analyzing intricate behavioral patterns from visual data. Each model brings unique strengths to the system: InceptionV3 handles diverse data transformations efficiently; ResNet-50 enables deeper networks with stable training through residual learning; MobileNetV2 offers lightweight deployment for edge devices; and GoogLeNet facilitates multi-scale feature learning. By leveraging these CNNs with transfer learning, the research enhances the precision, adaptability, and real-time performance of the distraction detection system, supporting broader goals in road safety and intelligent transportation systems.

2.5 CNN Applications in Computer Vision

Several convolutional neural network architectures such as MobileNetV2, GoogLeNet, AlexNet, and ResNet have been widely used in computer vision applications due to their strong feature extraction capabilities. To train a CNN for face recognition, facial photos are typically processed for detection and feature extraction using methods like Haar-like feature extraction. In one study, MobileNetV2 was enhanced using ArcFace for face classification and identification, and the model was tested on the Raspberry Pi 4B. Utilizing the concept of transfer learning, only the fully connected classification layers were trained, while the network's feature extraction layers used pre-trained weights. After 100 epochs, the model achieved 99.8% accuracy on the test set and 94.4% on the training set. The target accuracy was achieved by epoch 10, and the total training time on the Raspberry Pi was approximately 1 hour and 33 minutes (F. Wang et al., 2021). In another study, automatic pedestrian attribute

detection was explored to improve surveillance effectiveness. Key challenges included the low resolution of surveillance images and the multi-angle complexity of image acquisition. To address these issues, a model based on GoogLeNet with an inception structure was developed to enhance pedestrian attribute recognition by replacing the optimal local design with multiple dense substructures. The experiments showed that this approach significantly outperformed earlier methods (Jing et al., 2021). These applications demonstrate CNN's versatility, which could be extended to detecting specific behaviors, such as distracted driving, through advanced feature extraction techniques.

Architectures like ResNet have significantly enhanced performance in face detection, alignment, and identification. By utilizing ResNet's feature extractor and fine-tuning it with methods such as Approximate Nearest Neighbor Search, systems have achieved verification accuracies of 99.33% on the LFW benchmark, demonstrating the robustness of these models in identity verification tasks (Agarwal et al., 2021). This robustness proves essential for monitoring driver behavior, as subtle changes in facial expressions can indicate distraction or fatigue. Similarly, models like VGG-16 and VGG-19 have been applied in neurodegenerative disease detection, effectively classifying different dementia stages with high precision. These models' ability to analyse complex and subtle features in medical imaging can be adapted to detect signs of fatigue or distraction in drivers by analysing facial features and eye movements (Hsiao & Jang, 2019). Detecting dementia early provides a significant advantage, yet it remains challenging as symptoms often develop late. To address this, researchers enhanced the VGG-16 and VGG-19 architectures by adding a fully connected layer at the network's end. This modification aimed to classify four categories: very mild dementia, mild dementia, moderate dementia, and a non-dementia or control group. The model achieved impressive results, accurately identifying cases with a success rate of up to 99%. During training and validation, accuracy reached 99.68% and 99.38%, respectively. The study also included precision, recall, and F1 scores, key metrics derived from the confusion matrix, to further validate the model's effectiveness (Bagaskara & Suryanegara, 2021).

CNN models have also been widely applied in driver monitoring systems, particularly for detecting driver fatigue and drowsiness. In these systems, CNN architectures are used to analyse facial features such as eye closure, head pose, and mouth movements to determine the driver's alertness level. For instance, deep learning-

based driver drowsiness detection systems utilize convolutional layers to extract discriminative features from facial images and classify driver states into alert or fatigued conditions. Experimental results have demonstrated that CNN-based approaches can achieve high detection accuracy while maintaining real-time performance, making them suitable for integration into ADAS (Tirumalasetty, 2025). In addition, CNN-based methods have been used for head pose estimation and gaze tracking, which are important indicators of driver distraction. These systems analyse the orientation of the driver's head and eye gaze direction to determine whether the driver is paying attention to the road. By leveraging deep convolutional architectures, these models are capable of extracting subtle facial cues and behavioural patterns that indicate distracted driving activities such as looking at mobile devices or interacting with in-vehicle systems. Such applications demonstrate the effectiveness of CNN models in monitoring driver behaviour and improving road safety through intelligent vision-based systems (*No Title*, n.d.).

Correspondingly, traffic sign detection has become an essential aspect of novel driving assistance systems. A recent study compared the performance of InceptionV3 and VGG-19 algorithms for detecting traffic signs. Using a dataset of 44,028 images, with 80% used for training and 20% for testing, the study found that the InceptionV3 model achieved a higher accuracy of 94.38% compared to VGG-19's accuracy of 90.12%. The results demonstrated that the InceptionV3 model provided significantly better performance in traffic sign detection, which is crucial for improving safety features in advanced driving assistance systems (Verma & Bhoomi, 2023). In addition, an improved version of the GoogLeNet model was developed by Huang Zhonglin to address fatigue driving, a critical contributor to global traffic accidents. The study enhanced the GoogLeNet-v3 architecture by introducing a pruning algorithm to reduce unnecessary parameters and the Global Max Pooling technique to more effectively capture relevant image features, improving both accuracy and computational efficiency. A specific training strategy was applied using simulated driving data, which significantly refined the model’s ability to detect fatigue-related behaviors such as prolonged eye closure or head nodding. Furthermore, the optimized model greatly reduced the time required for image processing, achieving faster and more efficient performance compared to existing techniques. This reduction in processing time, while maintaining high detection accuracy, makes the model highly suitable for real-time fatigue monitoring, offering potential for integration into ADAS to enhance driving

safety (Zhonglin et al., 2023). Moreover, an innovative approach to key-point identification in animated drawing techniques has been developed, utilizing ResNet-50 to accurately extract key-points from stick figures and children's drawings. This methodology, combined with As Rigid As Possible shape manipulation and image segmentation, demonstrates the efficacy of ResNet-50 in the domain of dynamic animations. The ResNet-50 model, trained on the Amateur Drawings Dataset, enabled precise key-point detection and seamless integration with animation techniques, showcasing its potential for advancing visual representations in computer graphics. The model, while achieving moderate accuracy, provides a robust solution for converting static figures into lifelike animations by preserving the structural integrity of the original drawings (Prakash et al., 2024). Besides, research on pothole detection using deep learning has seen advancements with a modified ResNet-50 model specifically designed for training neural networks in pothole identification. In this study, the structure of the standard ResNet-50 model was modified to improve performance in training deep neural networks, using real-time traffic images from autonomous vehicles. By employing different training epochs and leveraging a dataset of images categorized into "with pits" and "without pits," the model's accuracy and validation percentage were significantly improved. These modifications demonstrated an enhanced capability for pothole detection in smart city systems, making the model more applicable for real-time processing in autonomous vehicle systems, a critical component in ensuring smoother and safer roads (Obreja & Dobrea, 2024).

The development of voice-based image captioning systems has seen significant progress with the use of the InceptionV3 transfer learning model, combined with a Gated Recurrent Unit variant of Recurrent Neural Networks. This approach utilizes a CNN for image recognition and integrates an attention model between the CNN and RNN to improve accuracy. The system generates coherent, grammatically correct captions from images and converts them into natural language speech via a text-to-speech translator. Designed for applications like assisting the visually impaired, self-driving vehicles, and virtual assistants, this model highlights the growing importance of deep learning in providing human-like abilities for machines. By employing InceptionV3 for visual analysis and GRU for language processing, the model effectively bridges the gap between image recognition and natural language generation, making it a robust solution for real-time image captioning (Thalanki, 2023).

2.6 Limitations of Computer Vision under Poor Lighting

An efficient absorbed light scattering model and the absorbed light scattering image generated under sufficient and consistent illumination can efficiently expose buried details in low-light photos. By controlling ambient light and transmittance, it can achieve the objective. The experimental findings showed that the ALSM algorithm demonstrated a certain ability to suppress low-density noise and achieved a decent balance in terms of structure, detail, and local feature similarity when compared to other state-of-the-art approaches (Y. F. Wang et al., 2019). Low-quality photos typically lose some important information and may lead Deep Learning approaches to produce subpar results Low-light conditions often produce low-quality images that degrade the performance of computer vision systems. To address this issue, several studies have explored image enhancement techniques such as generative adversarial networks (GAN) to reconstruct low-quality images and improve visual features for detection tasks. The low-quality defect images are rebuilt using a GAN, and a VGG16 network is created to identify the rebuilt images. The experimental outcomes for low-quality defect photos demonstrate that the suggested method obtains extremely good performances, with accuracies of 95.53-99.62% with various masks and noises, and they are significantly better than the other methods. Additionally, the improvement in image reconstruction quality was evaluated using several metrics, including Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), cosine similarity, and mutual information. Higher PSNR and SSIM values indicate better preservation of image details and structural information, while cosine similarity and mutual information measure the similarity between the reconstructed image and the original image. The results demonstrate that the reconstructed images achieved higher similarity with the original images, indicating improved image quality that is beneficial for accurate defect detection (Gao et al., 2021). Human eyes can track an object and identify it. The researcher builds an artificial vision system using these qualities as a guide. Machines for surveillance or automatic weapons can use this artificial vision system. Two cameras allow the vision to detect the target both during the day and at night. In this instance, image processing was used along with the Hue-Saturation-Value imaging method for creating pseudocolor images for day and night. Because of simulations, the program can locate the target both during the day and at night (Chen et al., 2018).

Additionally, a study proposes a computer vision system to address similar challenges in defect detection under poor lighting by enhancing defect contrast on transparent materials with structured light and neural networks. This system uses sinusoidal structured light to improve defect imaging, allowing the identification of defects even in low-contrast conditions. The study uses an unsupervised generative adversarial network to enhance sub-image quality and merge them into high-quality defect images, showing significant improvements over traditional dark-field illumination and other imaging techniques. This approach highlights the potential of advanced imaging systems to overcome limitations of computer vision in low-light environments, expanding applications like defect localization and segmentation (Liu et al., 2024).

Thus, the previous research addresses similar challenges of low-light image conditions by proposing a Retinex-based method for enhancement. This approach smooths the image while preserving edges, then divides the image by its smooth version to extract the reflectance component. These elements are processed to enhance image quality and maintain naturalness. This method helps tackle issues like haze or fog, improving visibility for applications such as surveillance and autonomous driving (Pandit & Parihar, 2022).

Besides, a study by R Balakrishnan focuses on object detection in low-light conditions using deep learning models such as the Single Shot Multibox Detector and Faster R-CNN. This research is crucial given that low-light environments significantly impact the performance of traditional object detection systems. To address this issue, the study employs advanced image enhancement techniques, including RetinexNet and Multi-Branch Low-Light Enhancement Network to improve visibility and image quality before detection occurs. These techniques help mitigate the challenges posed by poor lighting by enhancing the contrast and detail of the images captured. Experimental findings indicate that enhanced object detection accuracy can be consistently achieved, even in challenging lighting conditions. Specifically, the proposed methods improve the detection of various objects necessary for applications in autonomous driving and surveillance systems. This demonstrates the effectiveness of combining image enhancement with deep learning, offering promising avenues for better performance in low-light environments and contributing to the overall safety and functionality of intelligent vehicle systems (Balakrishnan et al., 2024). Also, recent work on Low-Light Object Detection introduces a method that addresses challenges in detecting objects

under poor lighting conditions through feature enhancement and fusion. This novel approach includes a feature fusion unit to improve the alignment of semantic features with object detection tasks and a low-light enhancement encoding unit to boost the semantic representation of low-light images. By first training an enhancement model, this method significantly improves object detection performance, even in low-light scenarios. The technique demonstrates the value of feature enhancement in mitigating the limitations posed by poor lighting, making it particularly beneficial for applications in autonomous driving and surveillance systems, where lighting conditions can often hinder object detection (Ye & Ma, 2023). Plus, recent research on noise-aware zero-reference low-light image enhancement has introduced a novel approach for improving object detection in low-light environments by incorporating noise information into the enhancement process. This method, based on the Zero-DCE approach, applies a noisy data training strategy and post-processing denoising techniques to enhance image structure and reduce the impact of noise. Experiments using the ExDark dataset have demonstrated that this technique can boost the precision of object detection tasks by up to 3%, further contributing to the overall performance of detection models in challenging low-light scenarios. The inclusion of noise-handling strategies offers a practical solution to the difficulties posed by uncontrolled environments, ensuring more reliable results for object detection models in real-world conditions (Ang et al., 2022).

These advancements show that while low-light conditions significantly hinder computer vision, innovative enhancement methods, deep learning architectures, and hybrid approaches are making detection more reliable. By improving image clarity, suppressing noise, and enriching feature representation, such systems are becoming increasingly robust and suitable for real-time safety-critical applications, including autonomous vehicles and nighttime driver monitoring.

2.7 Behavior Detection at Night

Under carefully regulated lighting conditions, face identification with great accuracy is produced via the development of deep neural networks and large-scale visible light face images. Even yet, it remains an extremely difficult task to recognize faces in low-light situations. Heterogeneous face identification using visible and near-infrared (VIS-NIR) images. Due in part to a person's right to privacy, it is difficult to collect pairs of VIS-NIR facial photographs. The abundance of tagged VIS face photos

is what makes face recognition successful in this field. An abundance of labelled NIR images, however, is difficult to come by. This encourages this paper to develop an efficient network architecture without needing a substantial quantity of paired face photos for training. Two goals for training are listed below. Learning the distribution of NIR/VIS facial features in the feature space is the first step. The second is to retain the face-identifying information while learning new components, such as texture, in pixel space (H. Wang & Wang, 2020). When there is little light, near infrared (NIR) imaging provides a practical and economical way to take high-quality pictures. The facial photos were captured in the near infrared and visible spectrums under a variety of lighting settings, including bright, dim, and intense illuminations. Under Near Infrared with strong, weak, and dark lighting circumstances, the quality of facial photographs remains consistent. In order to address the issue caused by fluctuating levels of illumination, work on NIR face images is widely implemented in mobile devices, video surveillance, night vision CCTV camera, and user identification applications (Salim et al., 2020).

Besides, research on vehicle detection at night has introduced a novel dataset that focuses on identifying vehicles in low-light conditions based on light reflections rather than direct visibility. This dataset comprises 59,746 annotated grayscale images from 346 different nighttime rural scenes. The images label vehicles, their headlights, and associated light reflections on surrounding objects like guardrails. By capturing these visual cues, the study attempts to mimic human anticipatory behavior in detecting oncoming vehicles using light reflections, a capability that traditional object detection models lack. This research highlights the importance of understanding subtle visual changes, like light reflections, in advancing autonomous driving and surveillance systems for improved nighttime performance. The dataset and methods discussed pave the way for developing enhanced object detection models that bridge the gap between human perception and computer vision systems in nighttime driving scenarios (Saralajew et al., 2021). Alternatively, research on dairy estrus detection at night has focused on developing a non-contact video monitoring system to avoid the discomfort caused by direct-contact sensors traditionally used in estrus detection. The system utilizes infrared cameras to monitor dairy cows at night, when estrus behaviors are most frequent, and employs image processing techniques to identify cows displaying estrus-specific behaviors such as mounting, increased activity, and other observable body and behavioral changes. By relying on imaging technology, this method reduces the need for constant human observation and minimizes errors in estrus detection, making it

easier for dairy farm owners to decide when to perform artificial insemination. This non-invasive approach offers a more efficient and comfortable solution for estrus detection in dairy cows (Yang et al., 2017).

Besides, advancements in night patrolling systems have introduced a multifunctional robot based on the rocker-bogie mechanism, designed to monitor extensive areas using various surveillance devices. This Arduino-based security system is equipped with multiple sensors, including fire, sound, temperature, and PIR sensors, and uses a six-wheel rocker-bogie configuration to navigate rough terrains while maintaining stability. The robot provides real-time monitoring by tracking parameters like outside temperature and fire detection, with alerts delivered via an alarm and GSM messages. The microcontroller, ATMega368, processes inputs from the sensors and delivers outputs on an LCD display while controlling the robot's movement to ensure effective night surveillance. This system offers significant improvements over traditional CCTV surveillance by overcoming limitations such as static camera angles and limited coverage through mobile monitoring capabilities. However, unlike the proposed system in this research, the multifunctional robot mainly focuses on environmental surveillance using sensor-based monitoring rather than vision-based driver behaviour analysis. The proposed study utilizes deep learning-based computer vision techniques to detect distracted driving behaviours using image data captured during night driving conditions. By employing CNN models and image preprocessing techniques, the proposed approach enables automated recognition of driver behaviours, which provides a more intelligent monitoring solution for improving road safety compared to conventional sensor-based surveillance system (P, 2022). Additionally, research has focused on providentially detecting vehicles at night by leveraging the emitted light from vehicle headlights, which can be detected before the vehicle is directly visible. A study introduced a test group to analyse the discrepancy between ordinary vehicle detection and provident detection, which humans naturally perform by detecting light features. The research collected a dataset containing annotations for light-related features and used it to train a neural network, demonstrating that machine learning models can learn to detect vehicles based on light reflections rather than direct visibility. This approach has the potential to enhance driver assistance systems, improving safety at night by allowing vehicles to adjust high beams or respond to approaching vehicles earlier. The findings underscore the importance of developing systems that mimic human anticipatory behavior in detecting vehicles at night, closing

the gap between human perception and current object detection technology (Oldenziel et al., 2020).

Moreover, research on provident vehicle detection in urban nighttime scenarios has introduced the PVDN-urban dataset, which focuses on detecting vehicles before they are directly visible by leveraging light reflections caused by headlights. This dataset contains over 14,000 images across 140 scenes, with detailed annotations of light reflections and bounding boxes for vehicles. It is designed to address the complexity of urban environments, where multiple light sources and reflections create more challenging conditions for detection. The study provides a semi-automated annotation pipeline and baseline results using state-of-the-art semantic segmentation models, demonstrating the difficulty of detecting small and irregularly shaped light reflections. The dataset's purpose is to advance the development of algorithms for provident vehicle detection at night in urban settings, significantly improving driving safety and comfort for autonomous and automated driving systems (Ewecker et al., 2024). This highlights the growing importance of fusing image-based cues such as reflections and near-infrared signals with machine learning techniques to enhance nighttime behavior detection systems and bridge the gap between human perception and artificial vision under low-light conditions.

2.8 Implementation of Deep Learning Model on an Embedded System

In implementing deep learning models on embedded systems, significant advancements have been made to overcome challenges such as high inference time and limited computational capability, which are common in resource-constrained platforms. An example is a real-time embedded target tracking system developed using a deep learning model that addresses these issues by introducing an adaptive detection strategy. This approach leverages the temporal coherence between video frames to reduce the frequency of deep model inference, thus minimizing system delays and improving overall speed. Thus, the system was designed with detailed hardware implementation, offering substantial improvements in power consumption and resource efficiency while maintaining detection accuracy. Experiments demonstrated that the proposed system reduced time delays by more than 50% while maintaining the same detection effect, making it suitable for applications like intelligent transportation, autonomous driving, and industrial inspection. This work illustrates the growing potential for deep learning

models to be effectively deployed on embedded platforms, ensuring both performance optimization and cost-effectiveness for real-time applications (Jiang et al., 2021).

Furthermore, Tsung-Han Tsai's research contributes significantly to the application of deep learning in embedded systems by presenting a sketch classifier technique realized on an embedded platform. The study demonstrates the use of depth-wise convolution layers to reduce the computational load of deep neural networks by approximately one-fifth. Using the Google Quick Draw dataset, the model achieves a high level of accuracy 98% for 10 categories and 85% for 100 categories while being implemented on the STM32F469I Discovery development board. This system facilitates real-time sketch classification, making it suitable for interactive applications, particularly in educational tools for children. The implementation of this system demonstrates the practicality of deep learning in resource-constrained environments, showcasing its potential for use in creative and educational applications while maintaining high performance and real-time execution(Tsai, 2019). In addition, Pratyush Raj's research presents an embedded deep learning-based traffic advisory system that focuses on detecting vehicle density in real time. The system employs two deep learning models, YOLOv3 and Tiny YOLOv3, which are optimized to provide fast inference times while maintaining a high level of accuracy. This enables timely traffic advice by informing users about the current road conditions, offering alternative routes when necessary. The system is designed to balance computational demands and real-time performance, which is critical for intelligent transportation systems within smart city environments. Moreover, the research compares multiple deep learning techniques, including YOLOv5, SSD, Faster R-CNN, and EfficientDet, analysing their effectiveness in vehicle detection tasks. This approach illustrates how embedded deep learning systems can be effectively implemented in traffic management, optimizing road safety and enhancing the flow of traffic (Raj, 2020). Additionally, Amir Gargouri's research presents an innovative approach to real-time photovoltaic fault detection using deep learning, implemented on the Raspberry Pi board. This system leverages the YOLOv8 model for fast and accurate fault detection, enhancing the reliability and efficiency of photovoltaic installations by enabling early identification of potential issues. The dataset used in the research was created with Roboflow software to optimize image editing and manipulation for the detection process. The implementation demonstrated strong performance in real-time detection tasks, providing a portable and cost-effective solution for photovoltaic fault management. The system's ability to

deliver accurate, real-time results highlights its applicability in renewable energy sectors, where timely maintenance is critical for sustaining optimal energy production and reducing equipment downtime (Gargouri & Haddeji, n.d.).

Similarly, Eleftherios Batzolis's research addresses the integration of machine learning in embedded systems, focusing on overcoming the inherent limitations such as restricted computational power, storage, and real-time performance. This study provides an overview of the challenges related to deploying machine learning algorithms in resource-constrained embedded environments, including efficient algorithm design and the necessity to balance computational demands with available resources. It highlights current solutions, such as leveraging more powerful embedded processors and optimizing machine learning methodologies, to enhance performance in real-time applications like data analysis and decision-making. Besides, the research discusses future challenges in the field, emphasizing the need for continuous advancements to fully realize the potential of machine learning in embedded systems, particularly in areas like the Internet of Things (IoT), where devices require both intelligence and efficiency to operate effectively (Batzolis et al., 2023). Luqman Ali et al. (Ali et al., 2021), research explores the performance evaluation of deep CNN-based techniques for crack detection and localization in concrete structures. The study proposes a customized CNN model and compares it against popular pretrained networks like VGG-16, VGG-19, ResNet-50, and InceptionV3. By using multiple datasets, the models were assessed based on computational time, crack localization accuracy, and classification metrics such as precision, recall, and F1-score. The findings emphasize how training data size and diversity impact model performance, with the customized CNN and VGG-16 models excelling in terms of both classification and computational efficiency. This research highlights the potential of CNN models to provide accurate and efficient automated crack detection solutions for maintaining and monitoring civil infrastructure. Sridevi Gadde's research focuses on SMS spam detection using various machine learning and deep learning techniques. The study utilizes a dataset from UCI and applies models like LSTM networks to detect spam messages in text form. The experimental results reveal that the LSTM model outperforms other methods, achieving an accuracy of 98.5% in detecting spam. By utilizing advanced machine learning techniques, this research tackles the growing issue of unsolicited messages sent by spammers for financial or promotional purposes. The study highlights how the growing volume of SMS traffic necessitates sophisticated filtering techniques to minimize spam

and enhance user experience. Using Python for implementation, the research demonstrates the practical application of machine learning and deep learning to a prevalent real-world problem, showcasing the effectiveness of LSTM in spam detection (Gadde et al., 2021).

Bita Darvish Rouhani et al. (Darvish Rouhani et al., 2017), introduce TinyDL, an end-to-end framework aimed at efficiently integrating deep learning models into resource-constrained embedded systems. TinyDL addresses key challenges such as limited memory bandwidth, energy constraints, and real-time performance by using a platform-aware signal transformation technique. This method optimizes data movement and computation, allowing deep learning models to execute within the resource limits of embedded devices. The framework demonstrated significant energy improvements, achieving up to two orders of magnitude better performance compared to previous solutions on the NVIDIA Jetson TK1 platform. TinyDL provides a practical solution for deploying deep learning in applications like healthcare, environmental monitoring, and portable sensing devices, showcasing its potential to make AI more accessible and efficient on constrained devices while maintaining accuracy. Collectively, these developments demonstrate how the integration of efficient model design, lightweight architectures, and hardware-specific optimization can enable the deployment of deep learning systems in real-time, resource-limited environments across a wide range of practical domains.

2.9 Summary

In summary, the literature highlights the growing significance of deep learning and advanced detection methods in addressing critical driving behavior and safety challenges. Drowsy and distracted driving remain major causes of accidents worldwide, emphasizing the need for improved detection systems. Deep learning models, particularly CNNs, have proven to be powerful tools for detecting risky driving behaviors, offering enhanced accuracy and adaptability compared to conventional methods. Vision-based techniques, EEG, and ECG-based methods, alongside sensor fusion approaches, have been explored for their potential to provide real-time, accurate assessments of driver behavior. Furthermore, research has addressed the limitations of computer vision in poor lighting conditions, particularly in night driving, by employing

advanced imaging and deep learning techniques such as NIR imaging, GAN-based models, and low-light object detection.

The implementation of deep learning models on embedded systems has also been a focal point, with significant advancements made to overcome challenges like high latency and resource constraints. These models have been effectively applied in real-time applications such as traffic management, photovoltaic fault detection, and vehicle detection at night. Techniques such as depth-wise convolution, along with frameworks like TinyDL, have optimized the computational load, energy consumption, and overall performance of deep learning models on resource-limited platforms. This progress demonstrates the practicality and potential of deploying deep learning models across various industries, including transportation, infrastructure maintenance, and renewable energy, enhancing real-time applications and overall safety.

Table 2.1 shows a detailed comparison of various studies on drowsy and distracted driving detection. It highlights key methods, findings, advantages, and disadvantages across different approaches, such as machine learning, deep learning, and sensor-based technologies. The table demonstrates significant progress in improving driving behavior analysis using CNNs, EEG, ECG, GANs, and other architectures. Each entry underscores the strengths and limitations of the methods used, helping to identify gaps for future research, particularly in achieving real-time accuracy, robustness, and adaptability in diverse driving environments.

Table 2.1
Summary of Literature Review

Author	Methode	Night Driving	Low-Light Illumination	Driving Behavior	Other Method	CNN Deep Learning	Embedded System	Accuracy Reported (%)	Dataset / Source	Limitation
Akshay Bhaskar (Bhaskar, 2018) 2018	EyeAwake system	✓	×	✓ eye blink, head nod, breathing	✓ ECG, heart rate	×	×	70	Own sensors	Latency, not multithreaded
Anshul Pinto (Pinto et al., 2019) 2019	CNN Eye Aspect Ratio	✓	×	✓ eye closure	×	✓	✓	96	Custom dataset	Focus only on eyes, no yawning
Sunhee Hwang (Hwang et al., 2021) 2021	EEG + CNN	✓	×	×	✓ EEG	✓	×	High accuracy subject-independent	SEED-VIG	Only EEG, limited models
Sylvia Bhattacharya (Bhattacharya & Bernadin, 2020) 2020	EEG + SVM	✓	×	✓ talking distraction	✓ EEG	×	×	85	EEG dataset	Young participants only
Shantanu Deshmukh (Deshmukh & Dehzangi,	ECG + Wavelet	✓	×	✓ conversation	✓ ECG	×	×	88.45	Naturalistic	Only 2 distraction types

2017) 2017										
Omid Dehzangi (Dehzangi & Galster, 2018) 2018	Multi-modal: physiological + motion + vehicle signals	✓	✓	✓ driving distraction detection	✓ ECG, GSR, CAN-Bus, motion sensors	×	×	99.1	On-road naturalistic 8 participants	Focus only on online distraction detection during natural driving
Chen Huang (Huang et al., 2020) 2020	Hybrid CNN	✓	✓	✓	×	✓	×	96.74	Image dataset	High computation cost
Faiqa Sajid (Sajid et al., 2021) 2021	EfficientNet	✓	✓	✓	×	✓	×	99.16	StateFarm	No audio detection
F. Faraji (Faraji et al., 2021) 2021	YOLOv3 + LSTM	✓	✓	✓ yawning, blinking	×	✓	✓ multithreaded	Robust detection	Custom dataset	Sensitive to illumination, head pose, glasses, expressions
Yifei Zhao (Zhao et al., 2020) 2020	InceptionV3 + LSTM	✓	✓	✓ fatigue, sleepiness, blinking	×	✓	×	+5% vs baseline	Custom video dataset	Cannot solve more complex problems

Yun Fei Wang (Y. F. Wang et al., 2019) 2019	ALSM	✓	✓	×	×	× Algorithm based	×	Balanced structure	Low-light image dataset	Does not solve noise amplification issue
Yiping Gao (Gao et al., 2021) 2021	GAN + VGG16	✓	✓	×	×	✓	×	95.53 – 99.62	Defect image dataset: with masks & noise	GAN training slow; retraining is required for new defects
Han Yang Chen (Chen et al., 2018) 2018	Modified Histogram Equalization + Neural Network	✓	✓	×	×	Partial (NN for correction)	×	Improved PSNR, better luminosity	MATLAB test images	Only 3 average brightness values
Fushuai Wang (F. Wang et al., 2021) 2021	MobileNetV2 + ArcFace	×	×	✓ face recognition	×	✓	✓ Raspberry Pi 4B	99.8% train, 94.4% test	3-user dataset	Very small dataset
Changhong Jing (Jing et al., 2021) 2021	GoogLeNet	×	✓	✓ surveillance	×	✓ pedestrian attributes	×	67.01	Surveillance dataset 17 attributes	Not for dynamic detection
Aman Agarwal (Agarwal et al., 2021) 2021	AlexNet	×	×	×	×	✓	×	96	Lung image dataset	Focus only on early diagnosis

Sheng-Hsing Hsiao (Hsiao & Jang, 2019) 2019	ResNet feature extractor + Re-ranking + ANNS	×	×	✓	×	✓	×	99.33	LFW, CASIA-WebFace, FG-NET	Edge device inefficiency, not optimized
Abitya Bagaskara (Bagaskara & Suryanegara, 2021) 2021	VGG16/19	×	×	×	×	✓	×	99.6 train, 99.38 validation	Dementia image dataset	No real-time, only cloud ML
Tanapat Wareechol (Wareechol & Chiracharit, 2021)	DarkNet-19 + Transfer Learning for Gait	×	×	✓ gait pattern	×	✓	×	99.83% walking, 98.10% backpack, 99.85% coat	CASIA-B dataset	DarkNet-19 not efficient, too similar objects
Huijiao Wang (H. Wang & Wang, 2020)	FFE-CycleGAN VIS ↔ NIR face translation	✓	✓	✓ face matching VIS/NIR	×	✓	×	96.5	Oulu CASIA & WHU VIS-NIR	Limited image fidelity improvements
Nilu R. Salim (Salim et al., 2020)	ResNet-34 (transfer learning) for NIR-VIS Face Matching	✓	✓	✓ face recognition VIS↔NIR	×	✓	×	98.54	Oulu-CASIA, HITSZ	Only heterogeneous FR, not homogeneous
Christopher (Streiffer et al., 2017)	DarNet: CNN + RNN + IMU sensors	✓	✓	✓ 6 driving behaviors	✓ IMU signals	✓	Partial: edge not optimized	87.02% collected, 80% downsampled	Custom driving dataset	Privacy, edge device limitation

CHAPTER 3
RESEARCH METHODOLOGY

3.1 Introduction

This chapter provides a detailed account of the research methodology to ensure a clear understanding of the procedures undertaken in this study. It mainly involves data collection, data processing, modelling of CNN models for distracted behaviors, comparisons of night driving performance against day driving, hardware development, and implementation on the NVIDIA Jetson Nano embedded platform. The NVIDIA Jetson Nano, equipped with a capable GPU and integrated AI features, enables efficient deployment of deep learning models, rendering it well-suited for edge-based applications. It supports frameworks like TensorFlow, PyTorch, and ONNX for model training and inference.

3.2 Research Flowchart

Figure 3.1 illustrates the research flowchart. The first step is to define the research objectives, which establish a clear direction for the study. The objectives focus on improving detection accuracy, selecting the most suitable CNN architecture, and ensuring real-time implementation on an embedded system. The data collection phase follows, involving 44 participants and a structured plan to acquire data on distracted driving behaviors. During this stage, appropriate data preprocessing techniques are also determined to prepare the dataset for model training. Subsequently, the model development phase involves designing CNN architectures, including InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet. These models are trained to recognize different distracted driving behaviors with optimal accuracy and computational efficiency. After the training process, model validation is conducted to evaluate the performance of each CNN model using metrics such as accuracy, precision, recall, and inference speed. This evaluation helps determine the most suitable deployment model. Finally, the selected model is implemented on hardware by deploying it onto the NVIDIA Jetson Nano, enabling efficient real-time processing on an embedded platform suitable for practical driver monitoring applications.

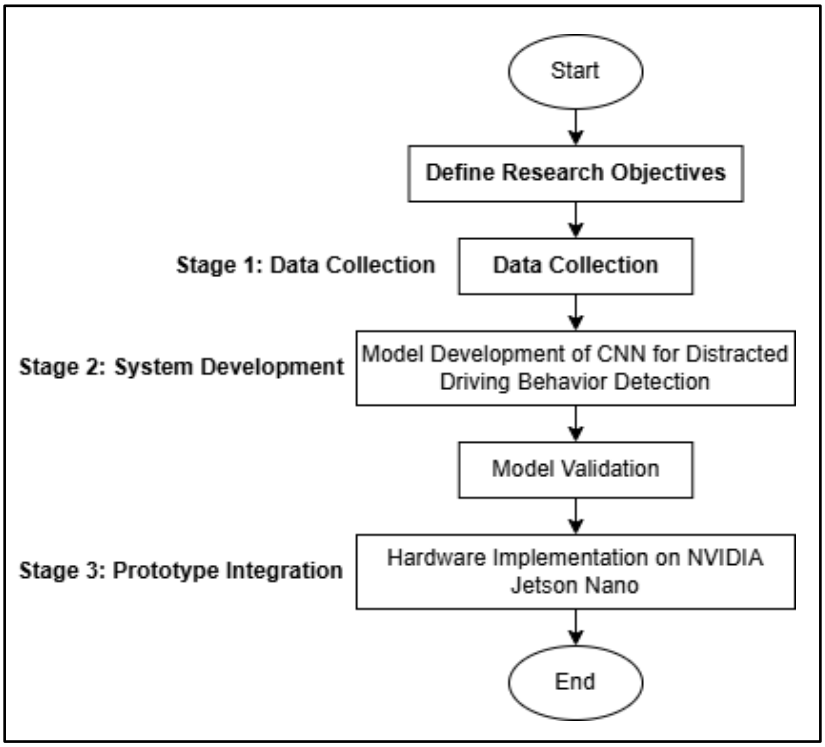


Figure 3.1 Flowchart of Research

3.3 Data Collection

The study was conducted at the Vehicle Intelligence and Telematics Lab, Faculty of Electrical Engineering, Universiti Teknologi MARA (UiTM) Shah Alam. The data collection process involved a total of 44 participants, comprising 30 external individuals who were not part of the research team and 14 internal participants, who included both students and staff. Volunteers were recruited by directly approaching lecturers and students from the same faculty. Participants were selected to represent a mix of gender, age range, and whether they wore spectacles. If a participant expressed interest and agreed to assist, a data collection slot was then reserved for them. Compared to a related study in (Sajid et al., 2021), which involved only 26 subjects, this study utilizes a larger and more diverse dataset. In total, approximately 1.41 Terabytes of storage were used to store 1,104,714 files, consisting of video recordings and extracted image frames across all participants. This volume of data is sufficient to support the training of deep learning models for behavior recognition. The sufficiency of the dataset for deep learning training is determined not only by the number of participants but also by the total number of labeled image frames generated per behavior class. With over one million frames captured across day and night sessions, the dataset provides

substantial intra-class variability such as differences in head movement, lighting conditions, and facial appearance, and inter-class variability among the six distracted driving behaviors. Such scale and diversity are critical for CNN-based supervised learning to achieve stable convergence and generalization performance. Therefore, the dataset size is considered adequate to support robust model training and validation for behavior recognition tasks.

As illustrated in Figure 3.2, the data collection process began with obtaining ethics approval, followed by participant recruitment and informed consent. Each participant was scheduled for 2 driving sessions: one during the day and one at night. Upon arrival, each participant received a 15-minute briefing on the data collection procedure and the 14 target behaviors. Before each session, all equipment and surfaces were sanitized, and hand sanitizer was provided for hygiene. The driving simulator and camera system were then set up and calibrated before recording began.

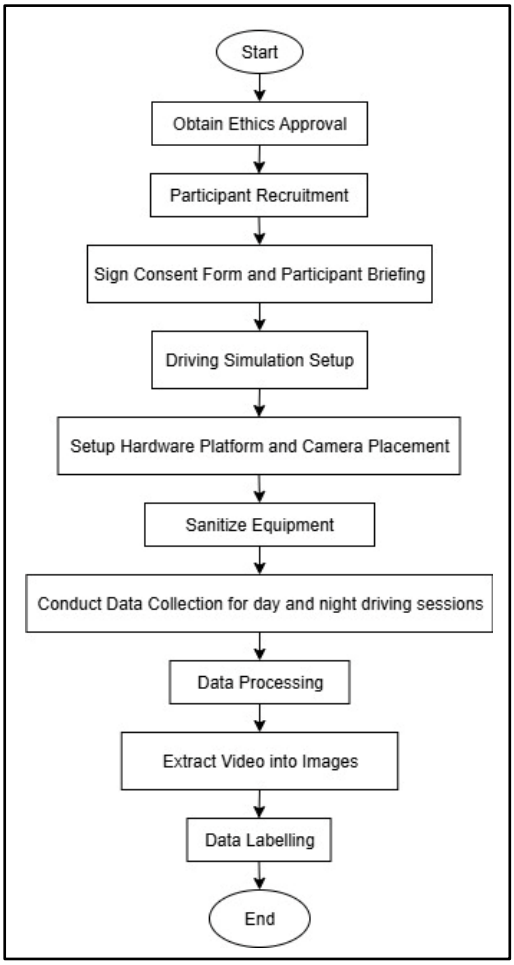


Figure 3.2 Flowchart of Data Collection

3.3.1 Driving Simulator Configurations

The driving simulator used in this research was designed to simulate real-world driving conditions in a controlled environment, enabling consistent and safe data collection for distracted driving behavior analysis. The driving simulator was assembled using essential components such as a steering wheel, accelerator, and brake pedals, a driver seat, a television screen to display the road simulation, and three cameras for capturing driver behaviors from multiple angles. The simulator was configured to provide a natural and immersive driving experience while maintaining the necessary precision for behavioral data acquisition.

Figure 3.3 illustrates the complete setup of the driving simulator used for this study. The positioning of the television screen, camera mounts, seating, and pedals was determined based on established driving simulator configuration practices and ergonomic considerations commonly adopted in driver behavior research. Consistency in hardware placement and field-of-view alignment across participants is important to ensure reliable data collection and behavioral comparability (Aiman Affandi et al., 2026). Table 3.1 presents the detailed hardware platform measurements, which were finalized before conducting the actual data collection. Although minor seat adjustments were permitted to accommodate differences in participant height, the relative positioning between cameras and the display was maintained to ensure consistency in image acquisition (Aiman Affandi et al., 2026). These measurements were critical to ensure that camera angles and field of view remained consistent across all participants. Before initiating full-scale data acquisition, a pre-test was conducted using this configuration to validate the simulator's functionality and to ensure a smooth and effective data collection process.

Figure 3.3 Driving Simulator

Table 3.1
Driving Simulator Setup Measurement

Name of Position	Distance (cm)
Television to floor	61
Camera to floor	111
Left camera to left side of television	16
Right camera to right side of television	16
Mid camera to left and right camera	26
Driver to television	69
Pedal to seat	55
Height of the chair	25

3.3.2 Data Collection Equipment

During the data collection process, a fully equipped simulation platform was designed to ensure that participants encountered an environment comparable to real driving situations. The setup included a large-screen television display mounted to a driving simulator rig equipped with a leather racing seat, steering wheel, gear shifter, and paddle controls. This configuration allowed participants to feel as though they were seated in a real vehicle. A total of 7 cameras were mounted around the display, which was protected by a transparent screen. These included 3 high-definition webcams placed at the left, centre, and right positions of the television, as shown in Figure 3.4, 1 IR camera located above the centre webcam, Figure 3.5, and 3 DVRs also positioned at the left, centre, and right positions, Figure 3.6.

Figure 3.4 High-Definition USB Webcam

Figure 3.5 IR Camera

Figure 3.6 Three DVR Cameras

After the hardware setup, the next step was to determine the most suitable method for presenting the driving context to participants while they performed specific behaviors. Three options were considered: using a racing video game, a physical driving simulator, or playing pre-recorded real-world driving footage. The chosen method involved playing actual dashcam footage of road driving on the television. This approach was selected because it closely mimics the experience of viewing a real windshield, helping participants focus more naturally on executing the required behaviors. In contrast, using a game-based environment might distract participants due to unrealistic visuals or interaction elements.

To prepare the footage, a video editing software was used to combine multiple clips, insert audio cues, add on-screen instructions, and trim unnecessary parts. Two separate instructional videos were created, one for the daytime drive and another for the night drive. These videos guided participants through 14 specific driving behaviors, including complex actions such as yawning and nodding off. These videos guided participants through 14 specific driving behaviors, consisting of seven predefined activities performed during both daytime and nighttime sessions. The seven behaviors included normal driving, yawning, nodding off, texting, calling, talking to passengers, and interacting with car features. However, the behavior “interacting with car features” was excluded from the final experimental analysis due to its limited distinguishable visual patterns under low-light conditions and its lower relevance to high-risk distracted driving classification. Consequently, only six distracted driving behaviors were selected for model training and evaluation in this study, resulting in 12 analyzed behavior categories across day and night conditions. Two weeks were allocated to complete video editing, as the software used required a subscription during the editing phase.

For recording the experiment, a multi-camera recording application was used to simultaneously capture video from all active camera angles. Figure 3.7 shows the multi-camera screen setup, which allowed the configuration of multiple video input sources. Four main camera views were used: left, centre, right, and infrared. The IR camera was

placed directly above the centre camera to capture low-light behavior during night sessions. The recording setup began by creating a new scene named "Data Collection." Each camera feed was added as a source and labelled appropriately (i.e., CAM01 for left, CAM02 for centre, CAM03 for right, and CAM07 for IR). For each source, a recording filter was applied, with the output format set to MP4. The recording filter was configured individually for each source to ensure consistent encoding parameters, uniform video quality, and synchronized frame characteristics across all camera views. This approach minimizes compression variability and ensures that extracted image frames maintain comparable resolution and clarity for subsequent CNN-based behavior recognition. The encoder used was x264, with rate control set to Constant Rate Factor (CRF), CRF value set to 23, and keyframe interval set to 1, ensuring consistent video quality across all recordings.

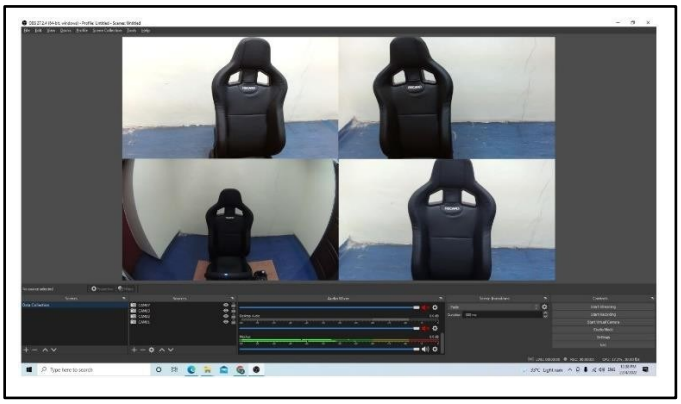


Figure 3.7 Multi-Camera Screen Recording Software

3.3.3 Camera Specifications and Placement

The camera setup includes three Logitech C920 Live Broadcast HD Webcams, placed at the left, middle, and right positions; one ELP IR camera (USBFHD06MT-KL170IR) positioned at the middle; and three DVR cameras located at the left, middle, and right positions. However, for this study, only two cameras are actively used: the middle Logitech C920 webcam as the VIS camera and the IR ELP camera as the IR camera. These two are selected based on their ability to operate under both normal and low-light conditions, which is critical for evaluating driver behavior during night driving. Table 3.2 presents the specifications and purposes of each camera used in the simulator setup.

Table 3.2
Three Types of Camera Specifications

Camera Type	Resolution & FPS	Mega pixel	Placement	Purpose
DVR Camera (VIS)	1920×1080 p, 60 FPS	2.0	Left, Mid, Right	High-quality night recording with built-in night vision capabilities
ELP IR Camera (USBFHD06M T-KL170IR)	1920×1080p, 30 FPS	2.0	Mid	IR imaging for sharp visuals in low-light conditions
Logitech C920 Live Broadcast Webcam (VIS)	1920×1080p, 30 FPS	3.0	Left, Mid, Right	HD auto-focus and light correction for daytime or well-lit conditions

3.3.4 Data Collection Procedure

Figure 3.8 presents the flowchart of the data collection procedure for better visualization and clarity. The procedure begins with participant briefing and simulator setup, followed by the execution of six predefined driving behaviors under both daytime and nighttime driving sessions. Each behavior was performed systematically to ensure consistency across all participants.

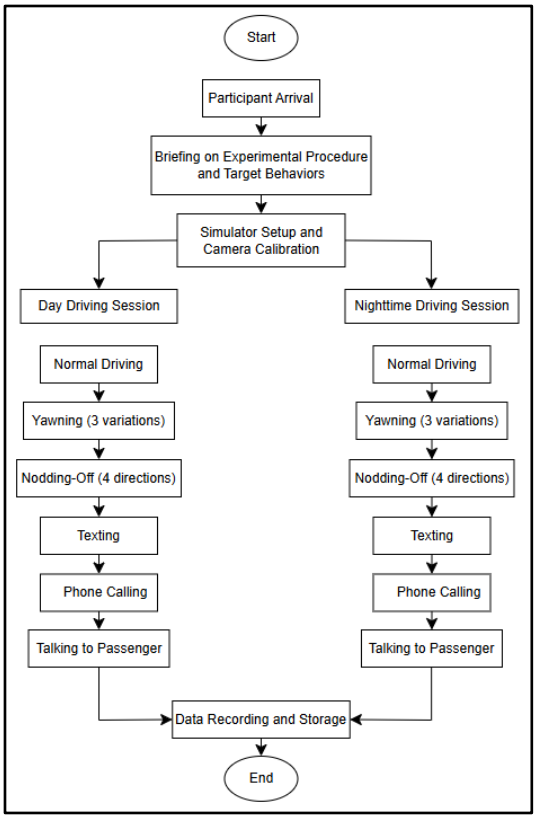


Figure 3.8 Flowchart of Data Collection Procedure

3.3.5 Data Processing

Data processing was conducted after the data collection phase to refine, synchronize, and prepare the recordings for CNN model training. As shown in Figure 3.9, Video Editor was used to cut and trim the raw recordings obtained from 44 participants, as the videos included preparation time and unnecessary segments. Each DVR camera session consisted of 21 one-minute videos with a total duration of 20 minutes and 45 seconds, which were first combined into a single continuous sequence. The beginning of the sequence was then trimmed to ensure that all participants started at the same time, allowing synchronization across the seven cameras, including the Logitech camera. In cases where interruptions occurred during recording, problematic segments were cut, and the affected behaviors were realigned to the start. Following this procedure, each 20-minute and 45-second session was restructured into 14 videos representing the different driving behaviors.

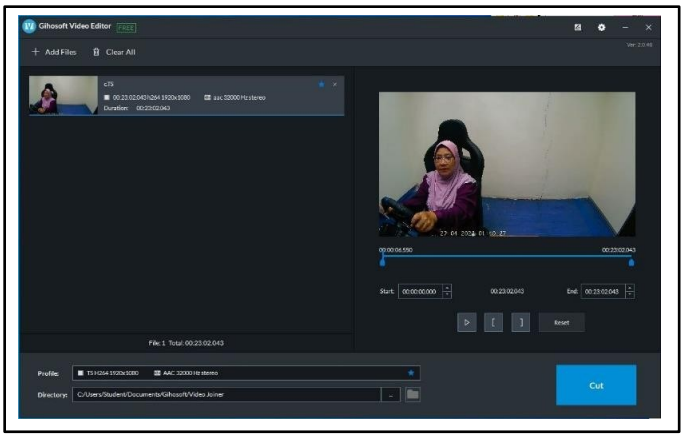


Figure 3.9 Video Editor

After trimming and synchronization, the data processing workflow, as illustrated in Figure 3.10, standardized each behavior to a one-minute duration, focusing on six selected behaviors: Normal Driving, Yawning, Nodding Off, Texting, Calling, and Talking. From 44 participants across two driving sessions (day and night), this process produced a total of 528 trimmed videos (44 participants × 2 sessions × 6 behaviors). Synchronization across multiple cameras was managed using Excel to generate a timing file in CSV format, ensuring that all recordings began at the same timestamp. For this study, only CAM05 (VIS) and CAM07 (IR) were used to provide complementary perspectives from the visible spectrum and infrared imaging. The processed videos were

exported in two resolutions, 1080p and 360p, to balance visual quality with computational efficiency. Labelling was conducted in Excel, where frames were annotated with binary values: “1” indicating the participant was performing the target behavior and “0” indicating otherwise. A Python script was then applied to extract only the frames labelled as “1,” with an under-sampling rate of three frames per second, producing a maximum of 180 images per behavior segment. This systematic workflow ensured that the final dataset of VIS and IR images was synchronized, balanced, and ready for training and evaluation of the CNN models.

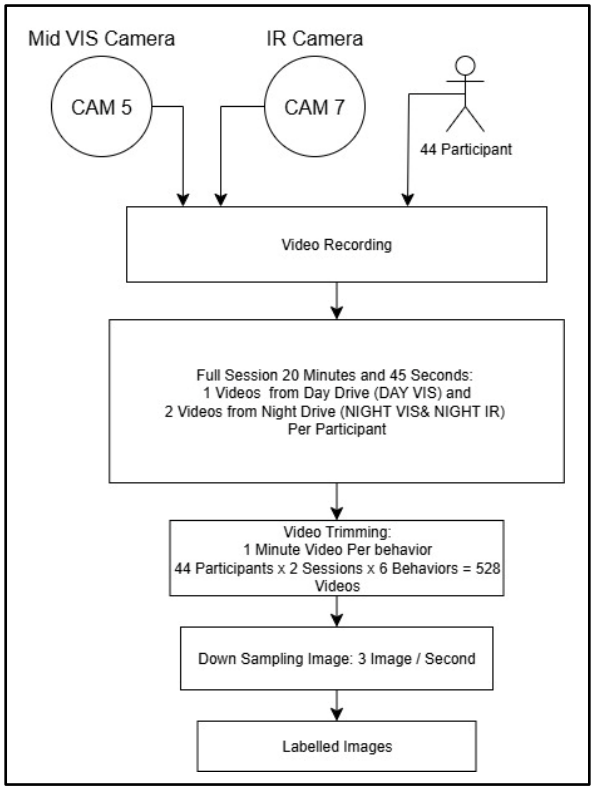


Figure 3.10 Data Processing Workflow for Video Recording and Image Extraction

The datasets used in this research were collected under different lighting conditions and sensor types to evaluate the robustness of the proposed models. Table 3.3 shows the types of image datasets employed in this study, together with their respective camera sources.

Table 3.3
Types of Images Used

Dataset Code	Full Name	Camera	Description
DAY-VIS	Daytime Visible Spectrum Images	CAM 05	Images captured under normal daylight conditions using a visible-light camera.
NIGHT-VIS	Nighttime Visible Spectrum Images	CAM 05	Images captured at night using a visible-light camera without preprocessing.
NIGHT-IR	Nighttime Infrared Spectrum Images	CAM 07	Images captured at night using an infrared camera to handle low-light conditions.
NIGHT-VIS-CLAHE	Nighttime Visible Spectrum Images with CLAHE	CAM 05	Night images enhanced using CLAHE.

As illustrated in Figure 3.11, the camera arrangement on the television consisted of three DVR units (left, mid, and right), three VIS cameras positioned at the left, centre, and right, and one IR camera mounted above the centre VIS camera. This configuration ensured synchronized recording from multiple viewpoints, providing both visible and infrared perspectives for day and night driving scenarios.

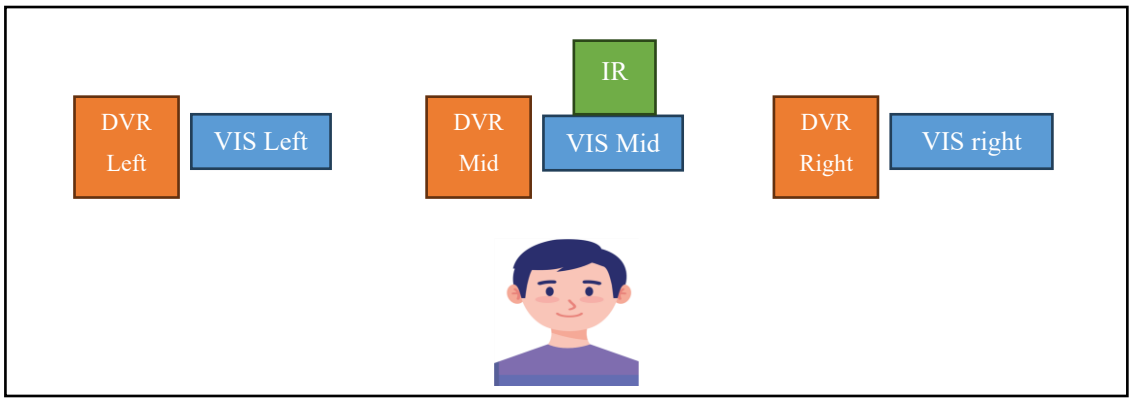


Figure 3.11 Camera Position Diagram on Television

3.3.6 Data Labelling and Annotation

All recorded videos were separated and tagged following data collection and processing to get the dataset ready for training and assessment. Fourteen shorter video clips, each depicting a single driving behavior lasting one minute, were taken from each full-session film, which had originally lasted roughly 20 minutes and 45 seconds. Each behavior could be examined separately and consistently marked because of this

division. During the annotating process, synchronizing video inputs from various camera viewpoints was a crucial step. The video segments captured by the left, center, and right VIS cameras were manually synchronized for each activity to guarantee that the behavior was exhibited from the same timestamp in all three views. Maintaining uniformity among perspectives was crucial, especially for actions like yawning that involved head motions or facial expressions.

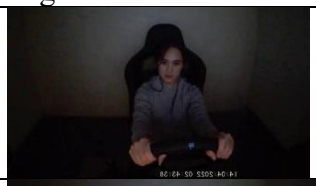
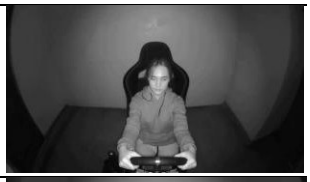
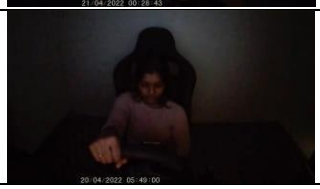
The visible light cameras, which comprised the DVR and USB webcams, were synchronized twice. The IR camera, which was positioned above the central VIS camera, was synchronized once more, especially for night driving. Frame extraction was performed at a rate of three frames per second (FPS) following synchronization, yielding roughly 180 images for every one-minute behavior segment. Instead of using bounding box annotation, a frame-level labeling approach was applied where each extracted image frame was assigned a single label. A binary annotation approach was employed to classify each extracted image based on one of the six predefined distracted driving behaviors. Frames exhibiting the targeted behavior were assigned a label of 1, whereas frames corresponding to neutral or transitional moments were assigned a label of 0. Therefore, each image frame corresponded to one label, resulting in approximately 180 labeled frames for every one-minute behavior segment based on the extraction rate of three frames per second. The labelling procedure was conducted consistently across both daytime and nighttime sessions for all synchronized camera views and the six driver behaviors: normal driving, yawning, nodding off, texting, calling, and talking to passengers. The dataset produced through this annotation process was systematically structured according to behavior category, lighting condition, and camera viewpoint to facilitate the training and evaluation of deep learning models capable of detecting distracted driving behaviors under both visible and infrared conditions. Each of the six distracted driving behaviors was recorded under both daytime and nighttime sessions, and frame extraction was performed at a fixed rate to ensure consistent image representation per behavior segment.

The data annotation process was conducted manually by examining the video recordings frame by frame. All frames were manually labeled one by one using the binary label values (1 and 0) to indicate the presence or absence of the target distracted driving behavior. Each sequence was carefully labelled according to the participants' specific driving behaviors. Six behaviors were considered in this study: normal driving, yawning, nodding off, texting, calling, and conversing with passengers. These

behaviors were recorded during both daytime and nighttime simulator sessions, with additional sequences extracted from the IR camera to facilitate low-light behavior analysis. Manual annotation was employed to maintain consistency and accuracy, with frames selected based on the clear visibility of facial expressions, head movements, and hand gestures relevant to each behavior. To ensure labeling consistency, a predefined annotation guideline was established before the annotation process, clearly specifying the visual criteria for each distracted driving behavior. The annotation was performed systematically by referring to these standardized criteria, and ambiguous or transitional frames were reviewed repeatedly to confirm the correct label assignment. This procedure reduced subjective interpretation and ensured uniform labeling across all behavior classes and lighting conditions. The annotated data were subsequently used to generate the training and validation image sets.

Table 3.4 outlines the availability of each distracted driving behavior under three different recording conditions: daytime, midnight, and infrared camera. This table presents a summary of the annotated image distribution and assists in elucidating the extent of data coverage across the various lighting conditions.

Table 3.4
Six Distracted Behaviors on Day Drive, Night Drive, and IR Camera

Behavior	Day	Night	IR
Normal			
Yawning			
Nodding			
Texting			

Calling			
Talking			

3.4 CNN Modelling for Driving Behavior Detection

Data augmentation was applied to the dataset before and after segmentation to increase the model's generalization capacity. This includes techniques like random cropping, flipping, brightness modification, and image resizing, which artificially raise the dataset and make the model more resilient to changes in input data. The Python programming language was used to train and assess several CNN designs. These architectures include InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet, all of which have demonstrated success in image classification tasks. The training was carried out on two platforms: a high-performance server with a multi-core processor, 64GB RAM, and an NVIDIA RTX A5000 GPU with 24GB memory, and an embedded edge device, the NVIDIA Jetson Nano, to confirm the trained models' real-time applicability on low-power hardware. The training dataset consists of annotated visual sequences obtained from 44 participants, capturing seven diverse driver actions during both day and night sessions. By training and testing the models on both the server and the Jetson Nano, the system's accuracy and efficiency for real-time detection of distracted driving behavior were assessed. Data augmentation was implemented in two steps to improve the model's resilience and alleviate class imbalance, as detailed in the section below.

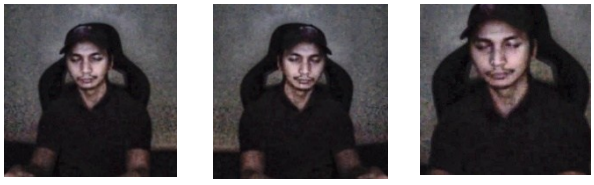
3.4.1 Data Augmentation and Handling Imbalanced Data

Data augmentation played a crucial role in improving both the performance and generalization capability of the CNN models developed in this study. It was applied in two phases: before training to address class imbalance, and during training to prevent overfitting. Initially, the dataset showed a significant imbalance in the distribution of behavior classes. Behaviors such as yawning, nodding, and talking were

underrepresented compared to more common classes like normal driving or texting. To mitigate this issue, augmentation techniques such as horizontal flipping, image cropping, zooming, rotation, and brightness adjustments were applied to the minority classes. Image cropping was used to generate slightly different views of the same image by selecting a smaller region of the original frame. This operation allows the model to learn important visual features from different spatial locations, such as facial expressions, head movements, or hand gestures, which are essential for recognizing distracted driving behaviors. By exposing the CNN to these spatial variations, cropping helps improve the model's robustness to small positional changes and camera viewpoint differences. In addition, image resizing was applied to ensure that all images matched the required input dimensions of the CNN architectures used in this study. Since models such as InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet require fixed input sizes, resizing ensures consistent image dimensions across the dataset and enables efficient batch processing during training. As a result, yawning behavior was augmented to a total of 10,000 samples, as shown in Figure 3.12, nodding behavior to 11,000 samples, as shown in Figure 3.13, and talking behavior to 10,000 samples, as shown in Figure 3.14. This ensured a more balanced distribution across all classes, helping the model learn equally from each category and reducing bias.

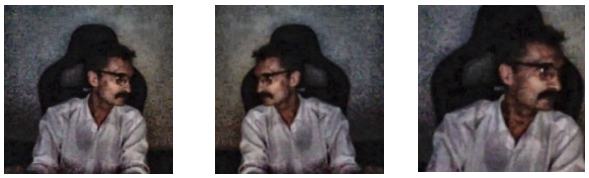
(a) Original (b) Flip (c) Zoom

Figure 3.12 Example of Image Augmentation for Yawning Behavior



(a) Original (b) Flip (c) Zoom

Figure 3.13 Example of Image Augmentation for Nodding Behavior



(a) Original (b) Flip (c) Zoom

Figure 3.14 Example of Image Augmentation for Talking Behavior

Image augmentation was applied dynamically during training. This real-time augmentation involved randomly modifying training images using minor transformations such as rotation, horizontal flipping, image cropping, zooming, brightness adjustments, and image scaling. By introducing these variations at each training epoch, the model was exposed to a wider range of visual representations, enhancing its robustness to new, unseen data and reducing overfitting. These augmentation operations were applied during the training process to improve the model's ability to generalize under different visual conditions, including variations in head orientation, lighting intensity, and driver posture. Image resizing was also performed as part of the preprocessing and augmentation pipeline to ensure that all input images matched the fixed input dimension required by the CNN architectures used in this study. Since models such as InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet require standardized input sizes, resizing ensures consistent image dimensions across the dataset while maintaining compatibility with the network architecture. This step also reduces computational complexity and allows efficient batch processing during model training.

Table 3.5 is divided into two sections: before data augmentation and after data augmentation. Each section shows the number of data samples for different behaviors, categorized as normal, yawning, nodding, talking, texting, and calling, along with the respective ratios within each class. Before data augmentation, there were imbalanced class distributions, particularly for yawning, nodding, and talking behaviors, which contained significantly fewer samples compared to normal, texting, and calling classes. After applying data augmentation, the distribution of these minority classes was increased to achieve a more balanced representation across all behaviors, as reflected in the updated ratios shown in Table 3.5.

Table 3.5
The Ratio of Training Data by Behavior Class Before and After Data Balancing

	Before Data Balancing			**After Data Balancing**		
Behavior	**DAY-VIS (Ratio)**	**NIGHT-VIS, NIGHT-VIS-CLAHE (Ratio)**	**NIGHT-IR (Ratio)**	**DAY-VIS (Ratio)**	**NIGHT-VIS, NIGHT-VIS-CLAHE (Ratio)**	**NIGHT-IR (Ratio)**
Normal	12,279 (0.248)	11,149 (0.229)	10,492 (0.266)	12,279 (0.156)	11,149 (0.138)	10,492 (0.149)
Yawning	2,496 (0.049)	2,637 (0.542)	1,877 (0.048)	10,496 (0.134)	11,637 (0.144)	11,877 (0.168)
Nodding	919 (0.017)	1,042 (0.021)	901 (0.023)	9,919 (0.075)	11,042 (0.137)	11,901 (0.169)
Talking	2,350 (0.047)	2,407 (0.049)	2,261 (0.057)	10,350 (0.132)	11,407 (0.141)	12,261 (0.174)
Texting	12,214 (0.246)	12,28 (0.252)	12,276 (0.311)	12,214 (0.155)	12,287 (0.152)	12,276 (0.174)
Calling	11,891 (0.239)	11,846 (0.243)	11,703 (0.296)	11,891 (0.151)	11,846 (0.146)	11,703 (0.166)

The dataset was divided based on participant identity, with 35 drivers used for training and nine drivers for testing. A subject-independent data splitting strategy was adopted to prevent data leakage and to ensure that the model was evaluated on unseen drivers rather than unseen frames from the same individuals. This approach is widely recommended in driver behavior recognition studies to properly assess model generalization capability across different subjects (Abouelnaga et al., 2018). Selection for the test set was carefully curated to reflect diversity in ethnicity, skin tone, hijab usage, and age, ensuring that the evaluation tested the model's generalization ability across demographic variability. Figures 3.15 and 3.16 illustrate examples of driving behaviors captured under NIGHT-VIS and NIGHT-IR conditions, respectively. Both figures present the six distracted driving behaviors considered in this study: normal driving, yawning, nodding, talking, texting, and calling. These examples highlight the differences in visual quality and feature visibility between VIS and IR imaging during night sessions.

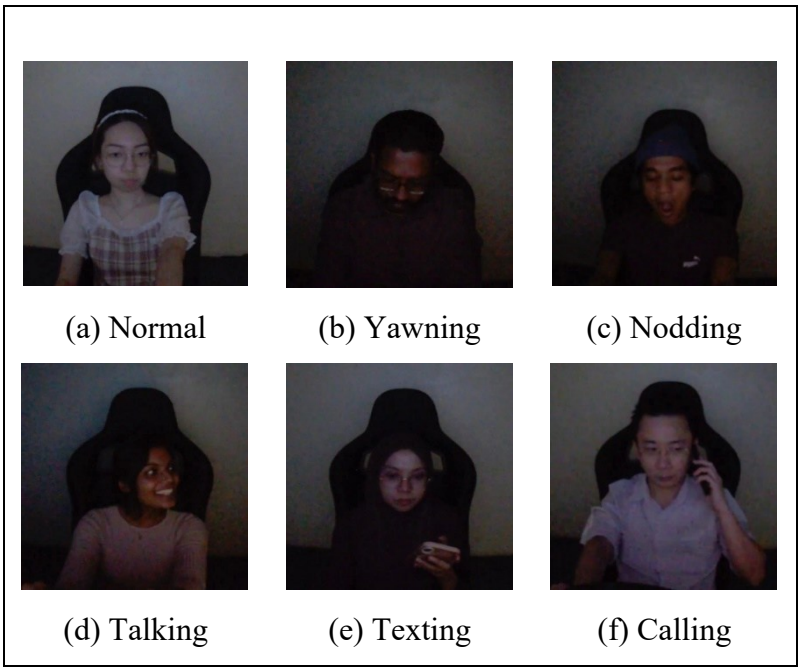


Figure 3.15 Example of Driving Behaviors for NIGHT-VIS Images, Which Includes (a) Normal, (b) Yawning, (c) Nodding, (d) Talking, (e) Texting, and (f) Calling

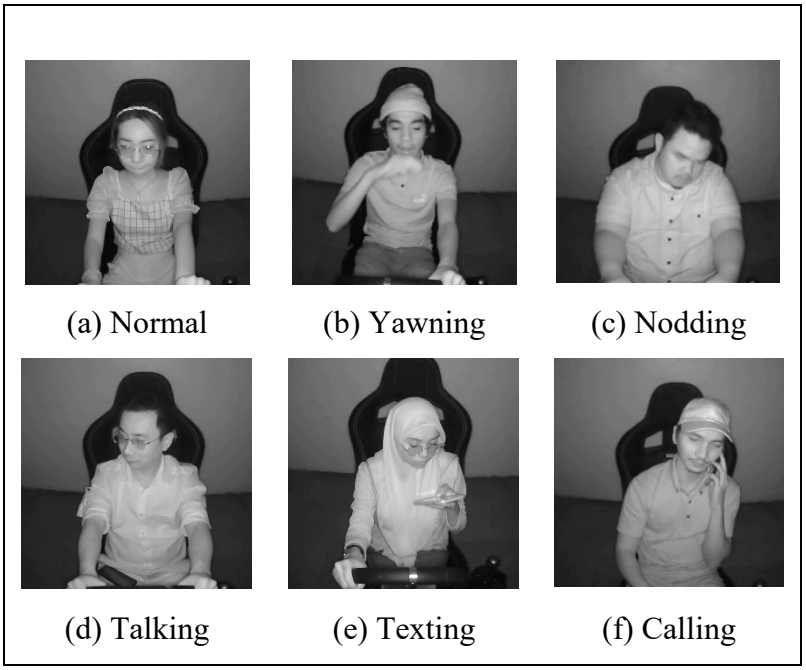


Figure 3.16 Example of Driving Behaviors for NIGHT-IR Images, Which Include (a) Normal, (b) Yawning, (c) Nodding, (d) Talking, (e) Texting, and (f) Calling.

3.4.2 Transfer Learning and Model Architecture

Transfer learning is a powerful technique in deep learning that leverages pre-trained models developed on large-scale datasets, such as ImageNet, and adapts them

to new tasks with comparatively smaller datasets. Instead of training a model from scratch, which requires significant computational resources and large amounts of labeled data, transfer learning allows researchers to repurpose the learned weights of established architectures. In this study, the convolutional base of each pre-trained model MobileNetV2, InceptionV3, ResNet-50, and GoogLeNet was retained as a fixed feature extractor. The top classification layers, which were initially designed to recognize 1,000 object categories in ImageNet, were removed and replaced with new fully connected layers tailored to classify seven distracted driving behaviors.

The main advantage of using transfer learning is that the lower and intermediate layers of pre-trained CNNs already capture generic features such as edges, textures, and shapes, which are applicable across various image recognition tasks. This means only the upper layers need to be trained on the new dataset, drastically reducing training time and improving generalization, especially when data availability is limited. By freezing the early layers and only training the top classification layers (or selectively fine-tuning some deeper layers), the models adapt efficiently to the domain-specific features of distracted driver behavior. This approach not only ensures better performance on smaller datasets but also allows deployment on low-resource platforms like the Jetson Nano without compromising accuracy.

Transfer learning was used by adapting pre-trained CNN models to the behavior classification task. The original top classification layers were removed and replaced with new trainable layers, since the original models were trained on the ImageNet dataset with 1,000 output classes. By fine-tuning these networks with our dataset, the models learned task-specific patterns for distracted driving behaviors.

In this study, four pre-trained CNN architectures were selected and evaluated: InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet. These models are widely used in computer vision tasks due to their proven ability to extract high-level features from image data. Each architecture offers different trade-offs between accuracy, computational efficiency, and memory requirements, which are critical considerations for real-time deployment, especially on edge devices such as the NVIDIA Jetson Nano.

InceptionV3 is a deep architecture comprising approximately 159 layers, known for its use of factorized convolutions and aggressive dimension reduction. These design choices allow it to achieve high accuracy with relatively low computational cost. It uses an input image size of 299 × 299 and contains around 23.9 million parameters. This

makes it suitable for applications that require strong classification accuracy with moderate resources.

MobileNetV2 is designed for mobile and embedded vision applications, especially those requiring low power consumption and high speed. It has 154 total layers with approximately 53 learnable layers and utilizes depth-wise separable convolutions to reduce model size and computation time significantly. It accepts input images of size 224 × 224 and has about 3.5 million parameters, making it highly suitable for edge computing platforms such as the NVIDIA Jetson Nano.

ResNet-50 utilizes residual connections to address the vanishing gradient problem in deep networks, enabling the training of very deep models while preserving high classification performance. It consists of 177 total layers and includes 50 convolutional layers that are trainable. With an input size of 224 × 224, it has approximately 25.6 million parameters. This architecture is suitable for complex classification tasks where model depth contributes to higher accuracy.

GoogLeNet, also referred to as InceptionV1, introduced the inception module, which applies multi-scale convolutions in parallel to extract diverse feature representations. It has 144 total layers, including 22 trainable layers, and accepts input images of 224 × 224 pixels. With roughly 6.8 million parameters, GoogLeNet strikes a good balance between model complexity and computational efficiency.

Table 3.6 presents a comparison of the selected CNN architectures based on several characteristics, including total number of layers, learnable layers, input image size, model size, and the number of parameters. This comparison provides a clearer understanding of the computational complexity and resource demands of each model, guiding their suitability for both server-side training and edge-device inference.

Table 3.6
Characteristics of The Pre-Trained CNN Architectures

Network	Layers	Learnable Layers	Network Size (MB)	Input Image Size	Parameter (Millions)
InceptionV3	159	48	92	299 x 299	23.8
ResNet-50	177	50	98	224 x 224	25.6
GoogLeNet	144	22	27	224 x 224	7
MobileNetV2	154	53	13	224 x 224	3.5

Table 3.7 outlines the training hyperparameters used for model training, including batch size, learning rate, number of epochs, and augmentation configurations for both pixel intensity and image scaling. These parameters were optimized to strike a balance between training efficiency and model performance. Although the number of training epochs was limited to 10, convergence analysis using validation accuracy and loss curves indicated that performance stabilization was achieved before or around the tenth epoch. This observation was consistent across all evaluated CNN architectures, namely InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet, where the validation accuracy curves showed performance stabilization around the tenth epoch. Additional training beyond this point did not produce significant performance improvements and increased the risk of overfitting. Therefore, 10 epochs were considered sufficient for optimal model generalization while maintaining efficient training time.

Table 3.7
Training Hyperparameters

Batch Size	**Learning Rate**	**Epoch**	**Pixel Range Augmentation**	**Scale Range Augmentation**
100	0.0003	10	-30 to 30	- 1.1

3.4.3 Model Development and Deployment

Figure 3.17 presents the block diagram of the distracted driving detection system pipeline, outlining the complete process from data acquisition to final deployment on an embedded hardware platform. The process begins with data acquisition, where image sequences were collected across multiple lighting conditions. This is followed by data preprocessing, which involves resizing and data augmentation to balance classes and improve generalization. Once the data is prepared, the next phase is model training, where four different CNN architectures, InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet, are fine-tuned using the pre-processed dataset. The performance of each model is then assessed during the model testing phase using standard evaluation metrics such as accuracy, precision, recall, F1-score, and confusion matrix.

The best-performing model is selected and moved to the inference and deployment phase, where it is deployed on an NVIDIA Jetson Nano to evaluate real-

time performance. Key considerations at this stage include inference time and hardware efficiency, which are crucial for practical deployment in vehicles.

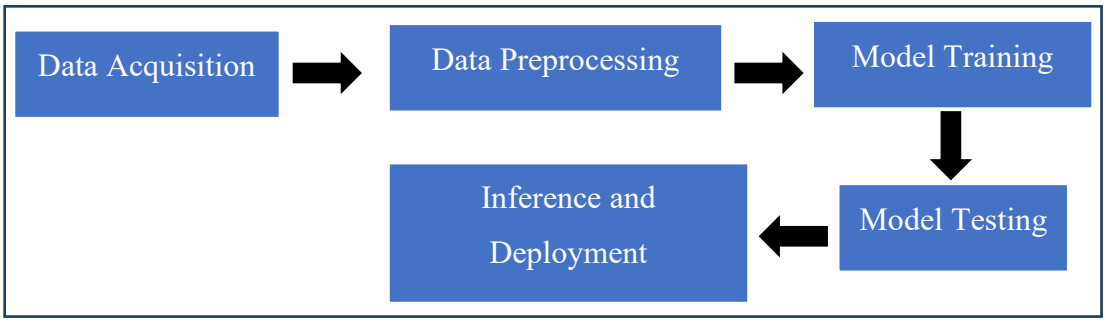


Figure 3.17 Block Diagram of The Distracted Driving Detection System Pipeline from Data Acquisition to Deployment on Jetson Nano

Performance evaluation is an essential part of the modelling process. This includes evaluating accuracy and analysing the confusion matrix, which consists of True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN), as illustrated in Figure 3.18.

Accuracy, one of the most used metrics in multi-class classification, is calculated directly from the confusion matrix. As shown in Equation (3.1), the accuracy formula uses the sum of True Positives and True Negatives in the numerator and the total number of entries in the confusion matrix in the denominator. Equation (3.2) presents the calculation for recall, while Equation (3.3) shows the calculation for precision.

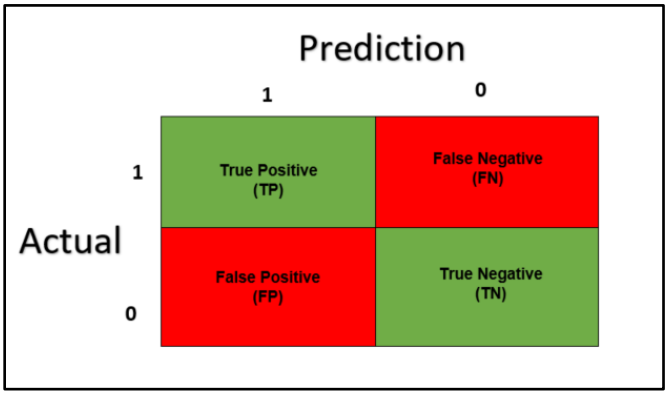


Figure 3.18 Confusion Matrix

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.2}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.3}$$

3.5 Enhancing Model Performance in Night Driving Conditions

In this study, enhancing the performance of distracted driving behavior detection under low-light and nighttime conditions was a primary objective. Night driving presents significant challenges due to poor illumination, shadows, and reduced visibility of facial and hand features, which are critical elements for accurate behavior classification.

To address these limitations, we employed VIS cameras to capture driver behavior across both day and night scenarios. While VIS cameras are commonly used for standard video capture, their performance during night conditions may degrade due to limited low-light sensitivity. Therefore, the IR camera, which provides better low-light capture capabilities, was evaluated for its potential to enhance night drive visibility.

The data collected from VIS cameras were processed into several modalities: DAY-VIS, NIGHT-VIS, and NIGHT-VIS-CLAHE. The CLAHE technique was applied to the NIGHT-VIS dataset to enhance image contrast and make subtle behavioral features more distinguishable. This enhancement technique improved feature clarity in poorly lit frames, aiding the model in learning more robust patterns from nighttime visual inputs.

Additionally, the IR camera was used in parallel with the VIS setup to record thermal images during night driving, creating a NIGHT-IR dataset. The IR modality allowed for behavior detection even in total darkness by capturing heat-based signatures of the driver's face and hands.

Through a combination of camera selection, contrast enhancement, and infrared imaging, the model's ability to detect distracted behaviors during nighttime was significantly improved. These strategies collectively enhanced the diversity and quality of input data, enabling the trained CNN models to achieve stable performance across varying lighting conditions, thus ensuring reliable real-time deployment, including on edge devices like the NVIDIA Jetson Nano.

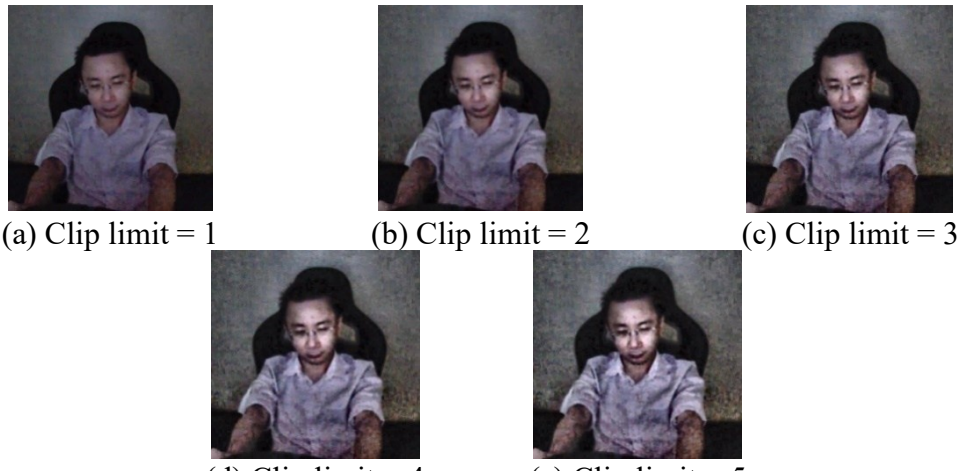
3.6 Contrast-Limited Adaptive Histogram Equalization (CLAHE) Implementation

CLAHE is a technique used to enhance the contrast and visibility of image details, particularly in low-light conditions. Unlike global histogram equalization, CLAHE operates on small regions (tiles) of the image and prevents over-amplification of noise by limiting the histogram contrast. Two key parameters govern its effectiveness: clip limit and tile grid size.

In this study, CLAHE was applied to the NIGHT-VIS image dataset to enhance visual clarity. The implementation process involved testing multiple parameter combinations to determine the most effective settings for this application. As illustrated in Figure 3.19, different clip limits were tested: 1, 2, 3, 4, and 5. The clip limit parameter was selected for tuning because it directly controls the contrast amplification threshold, preventing over-enhancement and excessive noise amplification in low-light images. A clip limit of 3 was chosen as it provided the best balance between contrast enhancement and noise suppression.

Similarly, tile grid sizes of 3×3, 5×5, 8×8, 10×10, and 15×15 was evaluated, as shown in Figure 3.20. The tile grid size was investigated because it determines the spatial region over which histogram equalization is applied, influencing local contrast enhancement and artifact generation. A grid size of 8×8 was found to be optimal, offering improved contrast without introducing visible artifacts. In standard CLAHE implementation, clip limit and tile grid size are the primary tunable parameters, while other parameters, such as interpolation method, remain fixed within the OpenCV framework used in this study. Therefore, parameter optimization focused on these two dominant factors that significantly affect image visibility and noise characteristics. These parameter selections play a critical role in enhancing the visibility of driving behaviors under low illumination. Figure 3.21 shows examples of the six driving behaviors under the original NIGHT-VIS condition before CLAHE enhancement. The images exhibit low contrast and reduced visibility due to limited illumination during nighttime simulation.

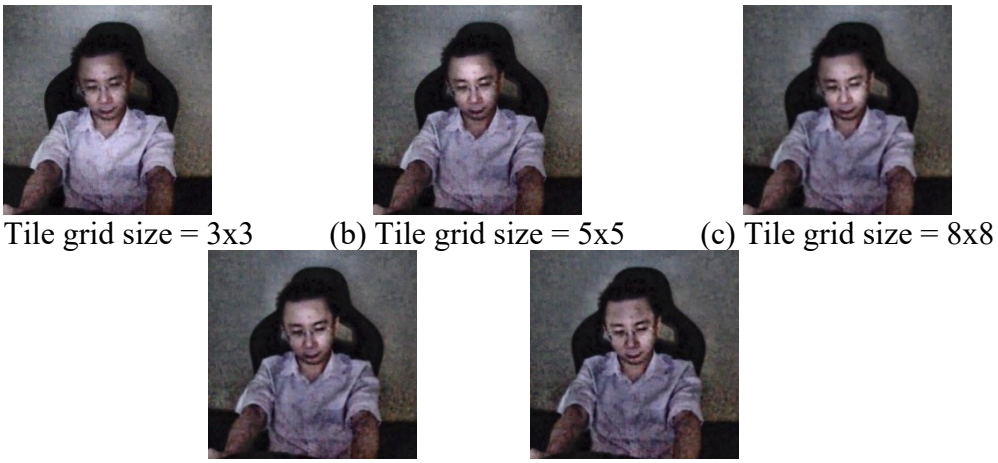
Figure 3.22 presents the six driving behaviors: normal, yawning, nodding, talking, texting, and calling after CLAHE enhancement was applied using the selected parameters. These enhanced images were used in the model training process under NIGHT-VIS conditions.



(a) Clip limit = 1 (b) Clip limit = 2 (c) Clip limit = 3

(d) Clip limit = 4 (e) Clip limit = 5

Figure 3.19 The Effect of Different CLAHE Clip Limits



(a) Tile grid size = 3x3 (b) Tile grid size = 5x5 (c) Tile grid size = 8x8

(d) Tile grid size = 10x10 (e) Tile grid size = 15x15

Figure 3.20 The Effect of Different CLAHE Tile Grid Sizes

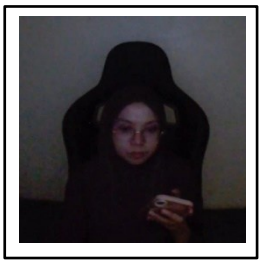


Figure 3.21 NIGHT-VIS (Without CLAHE)

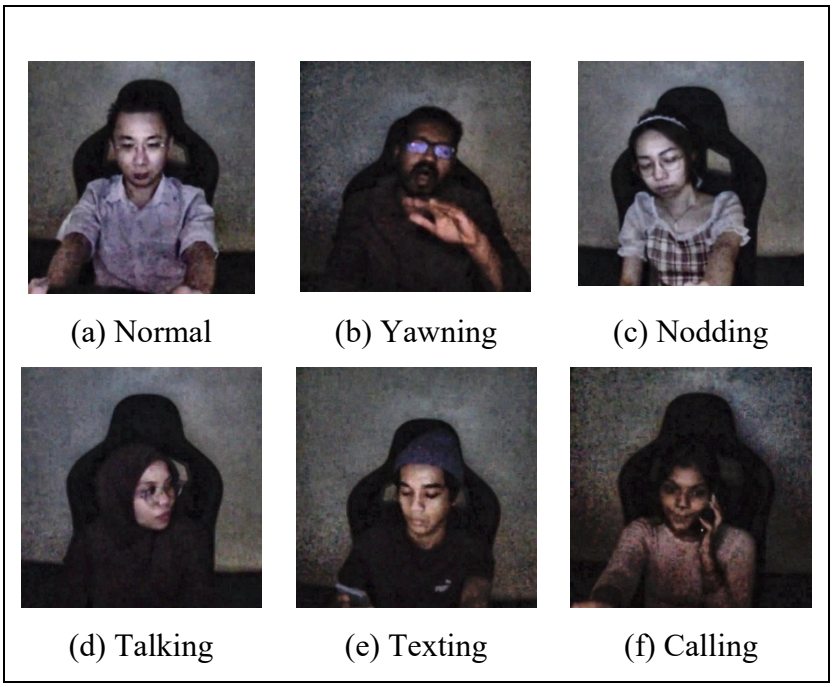


Figure 3.22 Example of Driving Behaviors for NIGHT-VIS CLAHE Images, Which Includes (a) Normal, (b) Yawning, (c) Nodding, (d) Talking, (e) Texting, and (f) Calling

3.7 Hardware Development

The NVIDIA Jetson Nano Developer Kit was chosen as the deployment platform for real-time distracted driving behavior detection due to its suitability for embedded AI applications. Jetson Nano is a compact yet powerful edge-computing device capable of running multiple neural networks in parallel for tasks such as image classification, object detection, and segmentation. It features a Quad-core ARM Cortex-A57 CPU, a 128-core Maxwell GPU, 4GB LPDDR4 memory, and native support for camera and display interfaces making it ideal for low-power, real-time AI inference.

To better illustrate the internal architecture and data flow, Figure 3.23 presents the block diagram of the Jetson Nano Developer Kit. The diagram outlines the board's core subsystems, including the CPU, GPU, RAM, power management unit, and I/O interfaces. It depicts how input from camera sensors is processed through the GPU and CPU, and how outputs are transmitted to peripheral devices such as displays and storage modules. The power supply configuration, which supports both 5W and 10W operation modes, is also highlighted, providing flexibility based on performance demands. With its modular design and support for CSI camera input, USB 3.0, HDMI, and GPIO pins, Jetson Nano offers excellent adaptability for integration into smart vehicle systems and

industrial automation platforms.

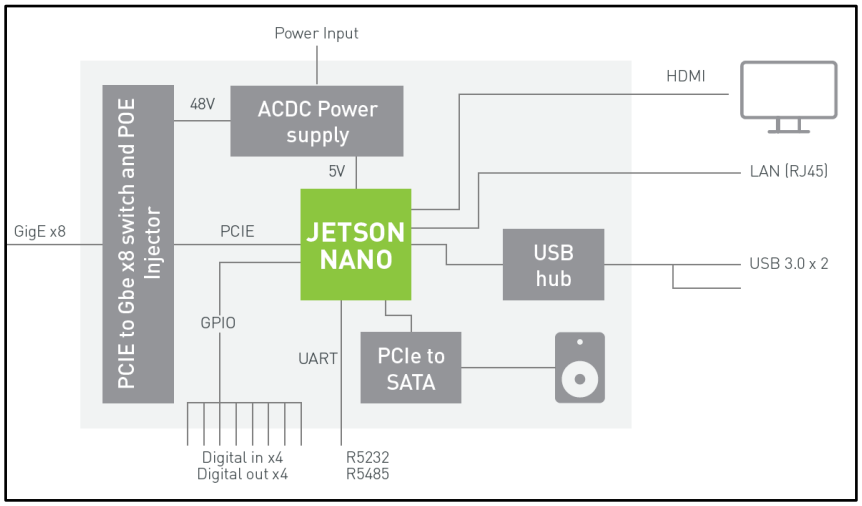


Figure 3.23 Block Diagram of NVIDIA Jetson Nano Developer Kit (NVIDIA, 2019)

One of the key advantages of using Jetson Nano is its ability to execute pre-trained deep learning models directly on the board, thanks to its integrated GPU and compatibility with TensorRT, CUDA, and cuDNN. This allows for efficient real-time inference even in environments with limited computational resources. The board runs on Ubuntu-based Linux OS, and supports frameworks like TensorFlow, PyTorch, and OpenCV, enabling seamless model conversion and deployment from the training environment.

Jetson Nano's low power requirement (5 to 10 watts) also makes it suitable for automotive-grade use, where efficient power usage is essential. It is powered via micro-USB or barrel jack and provides expansion via GPIO, CSI camera ports, and M.2 for storage or connectivity modules. These features make Jetson Nano an ideal candidate for on-board vehicle processing, reducing latency and removing the need for continuous cloud communication.

Figure 3.24 presents the real-time deployment pipeline of the distracted driving behavior detection system on the Jetson Nano platform. This pipeline illustrates the end-to-end stages involved in running the trained CNN models on the embedded hardware for real-world applications. The process begins with video input from either a VIS or IR camera, which is connected directly to the Jetson Nano. The incoming video stream is first processed through a frame extraction module, where individual images are captured for further analysis. These frames undergo preprocessing, including resizing, normalization, and any required augmentation. Once pre-processed, the frames are fed

into the trained CNN model, such as InceptionV3, MobileNetV2, ResNet-50, or GoogLeNet, which has been optimized and deployed onto the Jetson Nano using TensorFlow. The model performs real-time inference to classify driver behaviors like normal, yawning, nodding, texting, talking, or calling. After inference, the output is sent to the post-processing and visualization module, where the detected behavior is either displayed on a monitor or used to trigger further actions alerts, logging, or integration with vehicle safety systems. This real-time pipeline enables low-latency and efficient behavior detection, making it highly suitable for deployment in IDAS or smart vehicles.

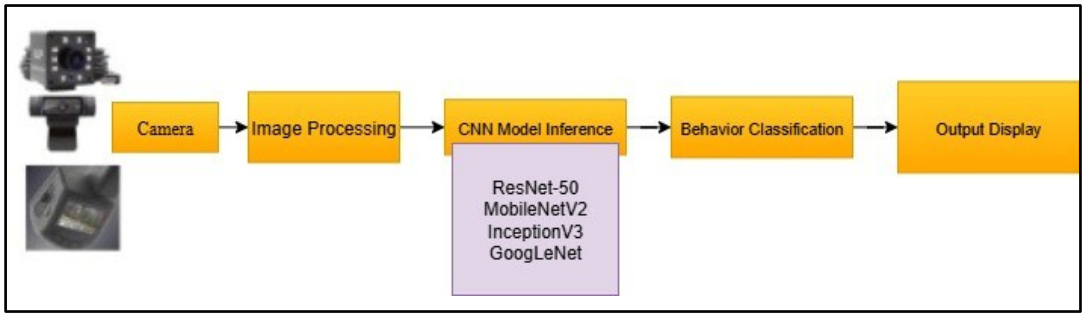


Figure 3.24 Real-Time Deployment Pipeline on Jetson Nano

3.7.1 In-Vehicle Implementation and Validation

To evaluate the practical applicability of the developed distracted driving detection system, an in-vehicle implementation was carried out to verify that the trained models could operate reliably in a real car environment and to measure their inference speed for real-time behavior recognition. Before conducting the in-vehicle test, the models were benchmarked on two hardware platforms: a Lenovo IdeaPad Gaming 3 laptop equipped with an NVIDIA GTX 1650 GPU with 4 GB VRAM and 16 GB RAM, running PyTorch with CUDA acceleration in FP32 precision mode, and an NVIDIA Jetson Nano J1010, running TensorFlow-optimized models in FP16 precision mode for increased throughput.

For the in-vehicle setup, a Logitech C920 HD webcam VIS was mounted at the upper centre of the windshield to capture the driver's facial expressions and upper body posture as shown in Figure 3.25, while the computing device either the Jetson Nano or the laptop was placed securely on the centre console or passenger seat and powered using a car inverter or USB power delivery adapter as shown in Figure 3.26. Cable management was arranged to avoid obstructing the driver's controls or line of sight.

The software stack consisted of Ubuntu OS, OpenCV for video capture, and a multi-threaded inference program, where the capture thread acquired frames at 30 FPS and resized them to 224 × 224 pixels for MobileNetV2, ResNet-50, and GoogLeNet, or 299 × 299 pixels for InceptionV3. The inference thread processed the frames using the GPU, while the post-processing thread displayed the classification result and logged performance data. On the Jetson Nano, inference was accelerated using TensorFlow in FP16 precision, whereas the laptop processed in FP32 mode using CUDA.

Testing was conducted in two phases: a static validation phase, in which the vehicle remained stationary with the engine running while the driver performed the six target behaviors, namely normal driving, yawning, nodding, texting, calling, and talking, and a controlled driving session along a short, low-speed route in a safe environment with a co-pilot monitoring system operation.

Performance evaluation focused on average frames FPS, with acceptance criteria set at a minimum of 10 FPS on the Jetson Nano, median latency below 100 ms, stable operation without crashes, and consistent classification output for at least 80% of the duration of each behavior segment. All results were logged into CSV files for subsequent analysis.

Figure 3.25 In-Vehicle Camera Setup and Computing Device Placement Inside Vehicle

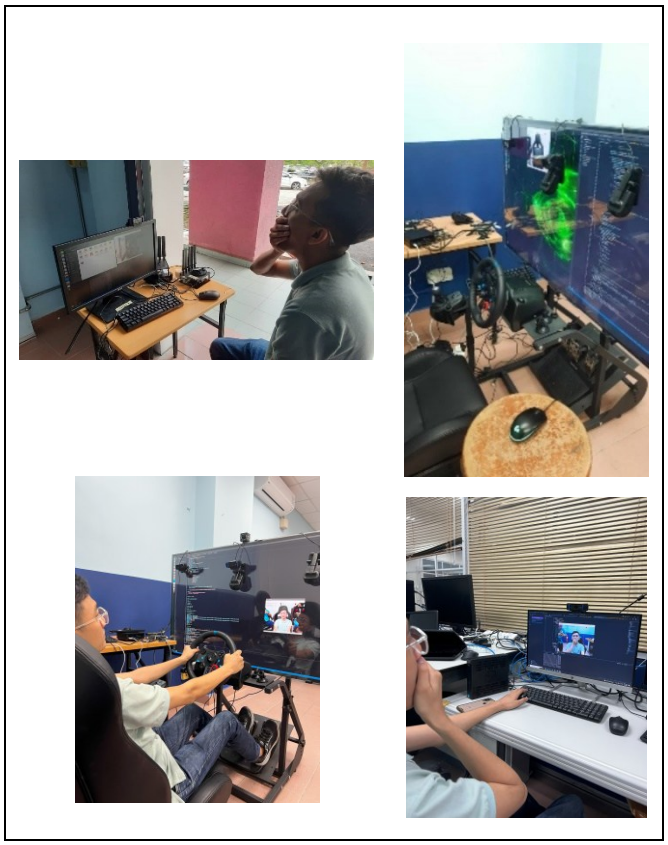


Figure 3.26 Test Setup in Laboratory Environment

3.8 Summary

This chapter describes the overall research methodology adopted in this study, covering the workflow from data acquisition to model deployment on an embedded platform. The research flowchart outlined the main stages, including literature review, objective definition, data collection, data processing, CNN model development, validation, and final deployment. Data collection was conducted at the Vehicle Intelligence and Telematics Lab, UiTM Shah Alam, involving 44 participants. A driving simulator environment and multi-camera setup were used to record driving sessions during both daytime and nighttime conditions. Although seven activities were included in the instructional guidance, only six distracted driving behaviors (normal driving, yawning, nodding off, texting, calling, and talking to passengers) were selected for the final dataset and modelling stage after excluding "interacting with car features." The recorded videos were trimmed, synchronized, and processed into standardized one-minute segments per behavior, followed by frame extraction and manual annotation based on predefined criteria to ensure consistent labeling. The chapter also presented the data augmentation strategy used to address class imbalance and improve

generalization. A subject-independent splitting strategy was applied by separating training and testing sets based on participant identity to prevent data leakage and to evaluate generalization on unseen drivers. Four transfer learning-based CNN architectures (InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet) were trained and evaluated using standard performance metrics such as accuracy, precision, recall, and confusion matrix. To enhance visibility under low-light conditions, CLAHE was applied to the NIGHT-VIS dataset, and its parameter selection process was described. Finally, the hardware deployment process was detailed, where the selected model was implemented on the NVIDIA Jetson Nano to assess real-time inference feasibility for practical in-vehicle applications.

CHAPTER 4
RESULTS AND PERFORMANCE EVALUATION

4.1 Introduction

This chapter presents the experimental results and performance evaluation of the distracted driving behavior detection models, structured according to the three research objectives: (a) to implement a CNN deep learning model for detecting distracted driving behaviors in night driving; (b) to evaluate model performance under different lighting conditions using VIS and IR images; and (c) to validate the prototype implementation on the NVIDIA Jetson Nano.

The CNN models were trained and tested on labelled datasets comprising six distracted driving behaviors (normal driving, yawning, nodding off, texting, calling, and talking to passengers) under varying lighting conditions, including day drive, night drive, and IR, using both VIS and IR cameras. As described in Section 3.3.1, the dataset was annotated with synchronized multi-camera video recordings to ensure consistency. Data augmentation techniques were applied to balance the dataset and improve model generalization. The models were evaluated based on metrics such as accuracy, precision, recall, and F1-score, with confusion matrix analysis to interpret class-wise performance. The model inference speed was measured using the NVIDIA Jetson Nano to assess real-time feasibility.

4.2 Dataset Preparation and Pre-processing Results

This section presents the results of the dataset generation and enhancement process, starting from behavior labelling to image processing, augmentation, CLAHE enhancement, and the final dataset summary. The source data was collected from a 20-minute driving video recording, where each distracted driving behavior was isolated into 60-second segments. A frame extraction rate of 3 fps was applied, resulting in three images captured per second. The dataset comprises six distracted driving behaviors Calling, Nodding, Normal, Talking, Texting, and Yawning recorded using three DVR cameras, three webcam cameras, and one IR camera.

4.2.1 Processed Image Samples

The captured images underwent resizing to fit the CNN input requirement of 224 × 224 pixels while maintaining the aspect ratio. Any excessive background noise was cropped to focus on the driver's upper body and face area, which are crucial for behavior classification. Figure 4.1 presents sample processed images after resizing and cropping for each driving behavior.

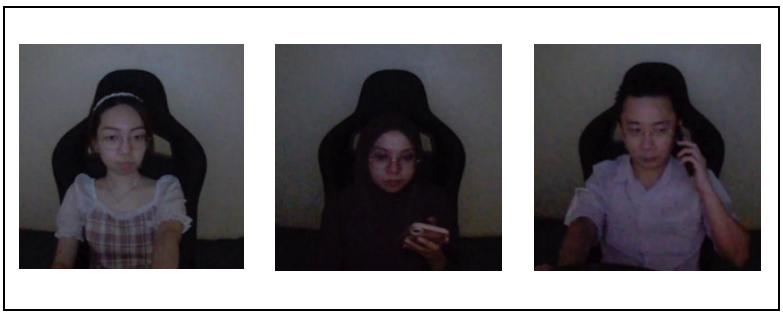


Figure 4.1 Processed Image Samples After Resizing and Cropping

4.2.2 Data Augmentation Outcomes

To enhance dataset diversity and minimize the risk of overfitting, data augmentation was selectively applied to three specific driving behaviors: nodding, talking, and yawning. These behaviors were targeted due to their relatively fewer captured instances under certain lighting conditions. The augmentation process involved random cropping, horizontal flipping, brightness adjustment, and image rotation. These transformations were intended to replicate possible variations in driver posture, lighting environments, and camera viewpoints that may occur in real-world night driving situations. Consequently, each original image for these behaviors produced multiple augmented variants, thereby increasing the dataset size and improving the model's ability to generalize. Figure 4.2 illustrates examples of augmented images for nodding, talking, and yawning behaviors.

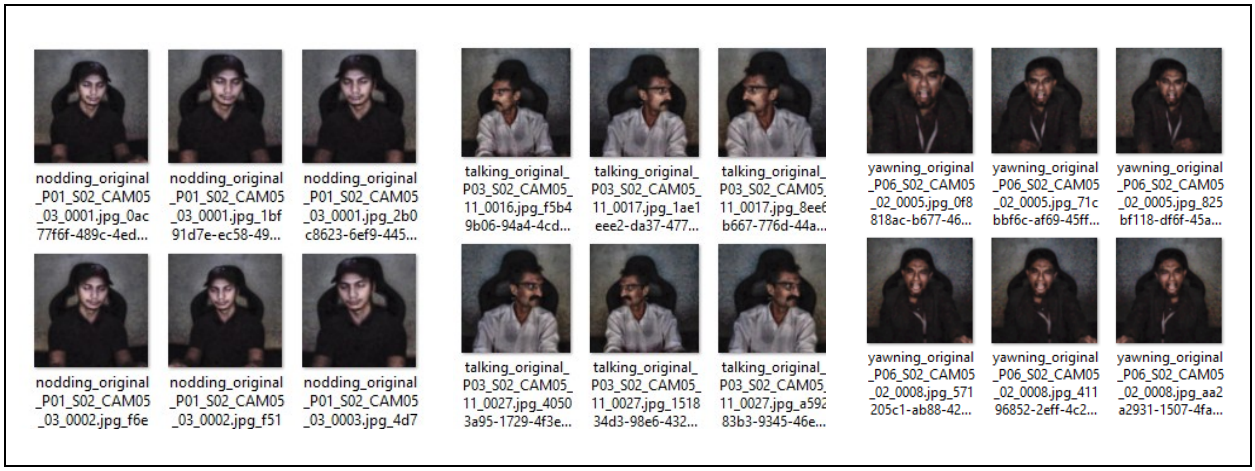


Figure 4.2 Examples of Augmented Images Generated for Nodding, Talking, and Yawning Behaviors

4.2.3 CLAHE Enhancement Results

In addition to data augmentation, CLAHE was applied to selected images, particularly those captured under low-visibility night conditions. The images were selected automatically based on their dataset category, specifically from the NIGHT-VIS dataset, which consists of images captured under nighttime lighting conditions using the VIS camera. Therefore, the selection was not performed manually but determined by the dataset grouping created during the data processing stage. CLAHE enhances local contrast by redistributing pixel intensity within small regions of the image, thereby improving the visibility of critical features such as facial expressions and head posture. This preprocessing technique was particularly effective for IR and VIS images, where standard histogram equalization often causes over-enhancement or noise amplification.

To illustrate the improvement, Figure 4.3 presents a sample of the nodding behavior in two versions: the original image without CLAHE and the corresponding image after CLAHE processing. The comparison highlights the increased contrast and enhanced detail clarity achieved through CLAHE, which can help the model recognize subtle head movements more effectively in night driving scenarios.

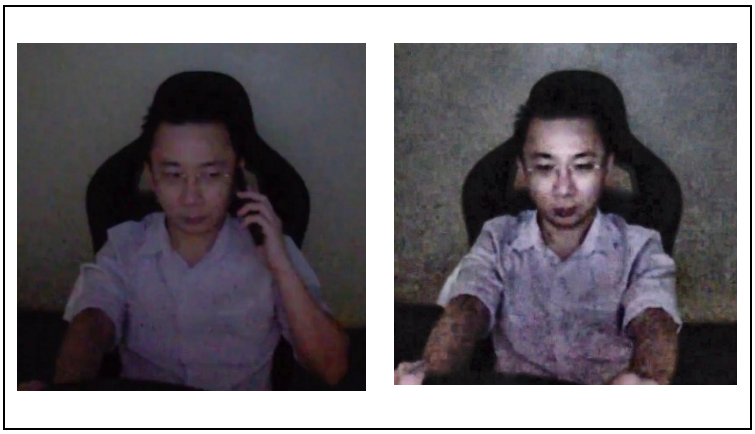


Figure 4.3 Comparison of Original And CLAHE-Enhanced Images for the Nodding Behavior Under Night Driving Conditions

4.3 CNN Model Training and Performance Validation

This section evaluates the performance of the proposed model in detecting distracted driving behavior during night driving compared to daytime driving using IR and VIS images. The analysis focuses on assessing how well the trained CNN models adapt to different lighting and imaging conditions, namely DAY-VIS and three night-time datasets: NIGHT-VIS, NIGHT-VIS-CLAHE, and NIGHT-IR. This evaluation is important to determine whether the models can maintain reliable classification performance when transitioning from well-lit environments to low-light or infrared-based conditions, which is essential for real-time distracted driving detection in both day and night driving scenarios.

Training and validation accuracy and loss graphs are used to evaluate the learning behavior and generalization capability of the CNN models, where training accuracy reflects the model's ability to correctly classify the training data, and validation accuracy measures its ability to correctly classify unseen data. Likewise, training loss represents the prediction error during training, while validation loss indicates the error during model evaluation on the validation set. Ideally, an effective model achieves high accuracy and low loss for both training and validation datasets. Significant differences between training and validation accuracy may indicate overfitting, where the model learns patterns specific to the training set but fails to generalize, whereas low accuracy in both datasets may indicate underfitting, where the model is too simple or insufficiently trained to capture relevant patterns.

In this research, the training and validation performance of four CNN architectures GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50, were examined for all four datasets to compare their convergence behavior, training stability, and ability to avoid overfitting. The accuracy and loss curves are also used to identify how each model responds to the challenges of night driving conditions, where reduced visibility and increased noise can impact performance. The NIGHT-VIS dataset provides performance evaluation in low-light visible-spectrum conditions, NIGHT-VIS-CLAHE applies contrast enhancement preprocessing to improve visibility, and NIGHT-IR assesses performance using infrared imaging, which is less affected by lighting variations. The comparative analysis between these datasets and the DAY-VIS dataset is essential for determining the most effective imaging modality for night distracted driving detection. The following subsections describe the training progress and convergence of each CNN model across the DAY-VIS, NIGHT-VIS, NIGHT-VIS-CLAHE, and NIGHT-IR datasets in detail.

4.3.1 DAY-VIS Images

For the DAY-VIS images, all four CNN models were trained and evaluated using daytime images captured under bright and consistent lighting conditions. This environment provides clear visual features, making it ideal for evaluating baseline model performance. The following subsections present the training and validation accuracy and loss curves for each model.

4.3.1.1 GoogLeNet

Figure 4.4 shows the training and validation performance of GoogLeNet on the DAY-VIS images. The model exhibits stable training behavior, where the training accuracy increases progressively and approaches convergence towards the final epochs. The validation accuracy also improves significantly after the early training stages. However, the validation accuracy does not perfectly match the training accuracy because the model is evaluated on unseen validation samples that contain natural variations in facial appearance, head movement, and lighting conditions. These variations introduce additional complexity during validation, resulting in slight fluctuations between the two curves. Between epoch 7 and epoch 17, the validation

accuracy fluctuates while the training accuracy continues to improve. This behavior is commonly observed when the model begins to specialize to the training data while still maintaining acceptable generalization on unseen data. The fluctuations in validation accuracy indicate that the validation dataset contains more challenging samples compared to the training data, but the overall trend still demonstrates stable learning without severe overfitting. Similarly, the training loss decreases consistently as the model continues to learn the feature representations of distracted driving behaviors. In contrast, the validation loss shows several fluctuations between epoch 7.5 and epoch 17.5. This occurs because the validation dataset contains variations that are not fully represented in the training set, causing temporary increases in prediction error. Such behavior is typical in deep learning training when the model is exposed to different data distributions during validation. Despite these fluctuations, the overall decreasing trend of validation loss indicates that the model continues to generalize effectively without experiencing significant overfitting.

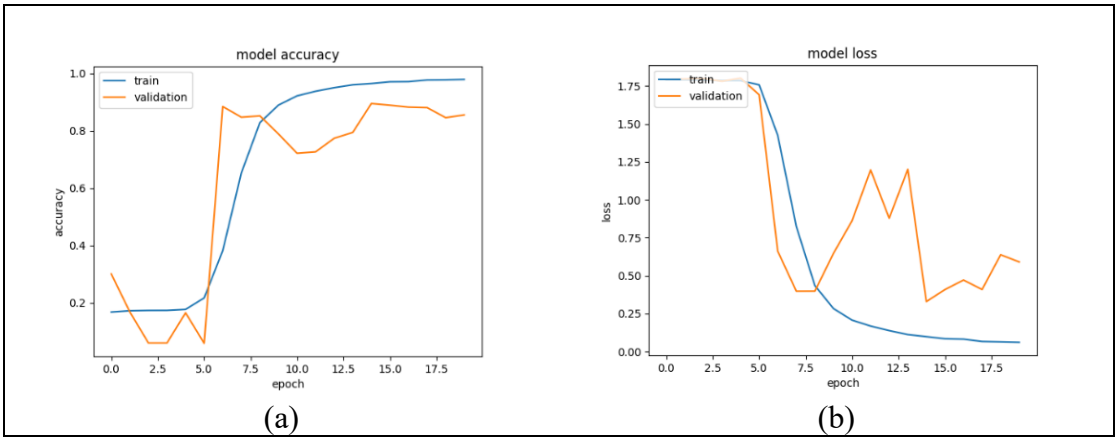


Figure 4.4 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for GoogLeNet Trained On DAY-VIS Images

Although the training was configured for 10 epochs, the plotted curve shows additional intermediate evaluation points recorded during the training process, which results in more detailed visualization of the learning trend across the training iterations.

4.3.1.2 InceptionV3

Figure 4.5 illustrates the training and validation performance of InceptionV3 on the DAY-VIS dataset. As shown in Figure 4.5(a), the training accuracy increases steadily and reaches approximately 0.98, indicating that the model effectively learns discriminative features from the training images. The validation accuracy also improves during training but does not exactly match the training accuracy. This difference occurs because the training dataset is directly used to update the model parameters, whereas the validation dataset is used to evaluate the model on unseen samples. Therefore, slight differences between the training and validation accuracy curves are expected. A noticeable drop in validation accuracy is observed around epoch 18, which may be caused by variations in the validation samples, such as differences in driver posture, illumination, or background conditions. Figure 4.5(b) shows the training and validation loss curves. The training loss decreases steadily, indicating effective optimization during training. In contrast, the validation loss fluctuates across several epochs, reflecting variations in the validation data. Overall, the relatively small gap between the training and validation curves suggests that the InceptionV3 model achieves stable learning behavior and reasonable generalization performance on the DAY-VIS dataset.

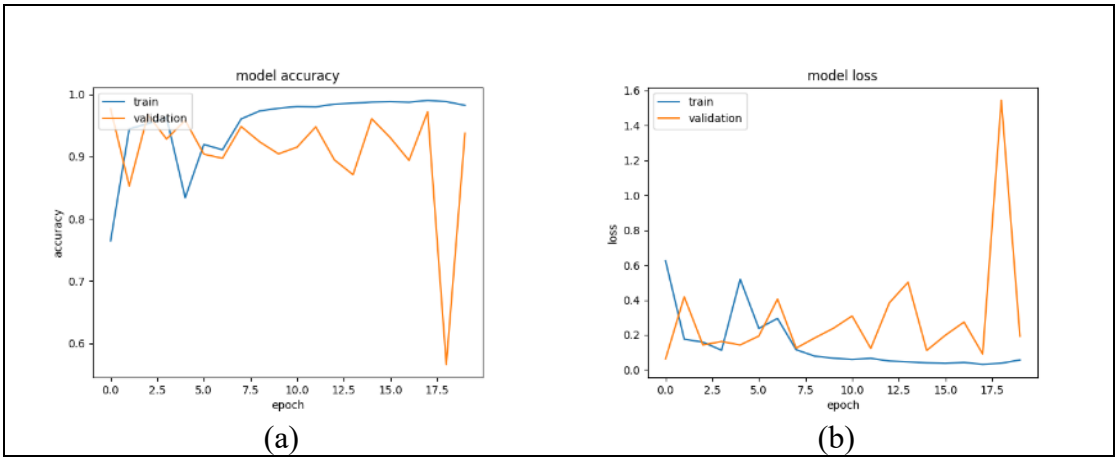


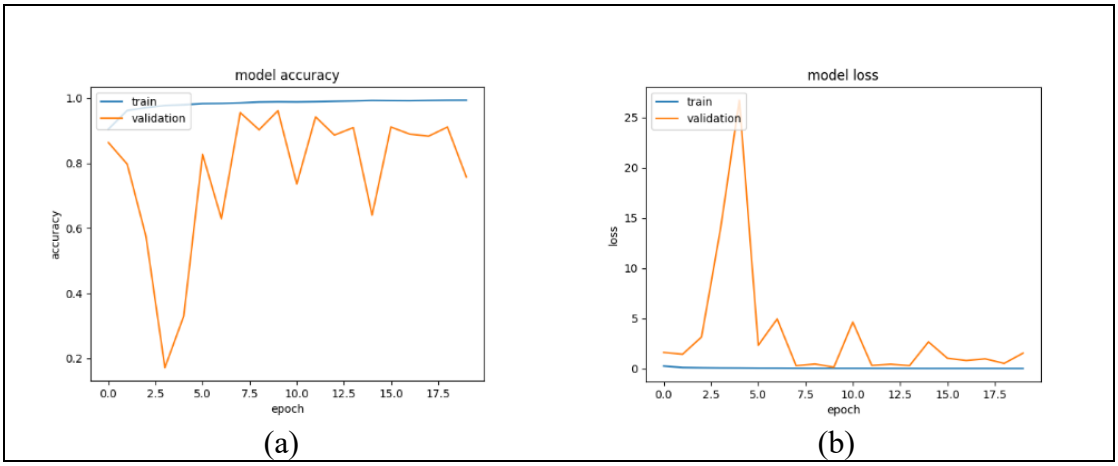
Figure 4.5 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for InceptionV3 Trained On DAY-VIS Images

During the training process, validation was performed at the end of each epoch to evaluate the model performance on unseen validation data. This procedure allows the monitoring of model generalization throughout the training process. Therefore, the

validation accuracy and validation loss curves shown in Figure 4.5 continue across all epochs until the final epoch of training. The validation process does not stop at an intermediate stage, as the evaluation is conducted continuously after each epoch to observe the learning behavior and detect possible overfitting.

4.3.1.3 MobileNetV2

Figure 4.6 presents the training and validation performance of MobileNetV2 on the DAY-VIS dataset. As shown in Figure 4.6(a), the training accuracy rapidly increases and remains close to 1.00, indicating that the model successfully learns the training data. However, the validation accuracy fluctuates significantly across epochs, ranging approximately between 0.2 and 0.95. This difference between training and validation accuracy indicates that the model does not consistently generalize to unseen validation samples. The fluctuations in validation accuracy may be caused by variations in the validation dataset, such as differences in driver posture, illumination conditions, and background complexity. Figure 4.6(b) shows the training and validation loss curves. The training loss decreases steadily and approaches zero, indicating effective optimization during training. In contrast, the validation loss fluctuates and shows a noticeable spike around epoch 4, suggesting instability when predicting certain validation samples. Overall, the fluctuations in validation accuracy and loss indicate that MobileNetV2 demonstrates weaker generalization performance for the DAY-VIS dataset.

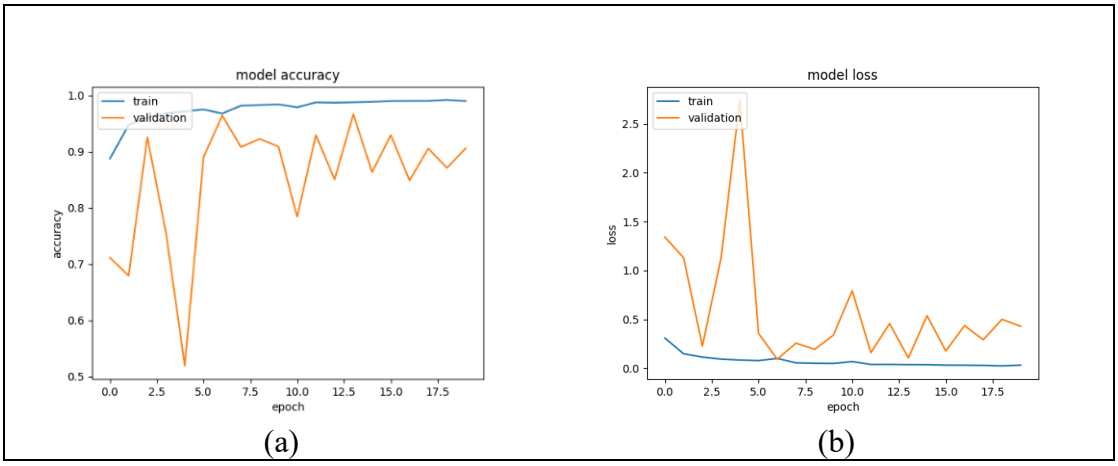


(a) (b)

Figure 4.6 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for MobileNetV2 Trained On DAY-VIS Images

4.3.1.4 ResNet-50

Figure 4.7 illustrates the training and validation performance of ResNet-50 on the DAY-VIS dataset. As shown in Figure 4.7(a), the training accuracy increases steadily and approaches 0.99, indicating that the model effectively learns discriminative features from the training images. The validation accuracy also improves across the epochs but does not exactly match the training accuracy. This difference occurs because the training dataset is used to update the model parameters, while the validation dataset consists of unseen samples used to evaluate the model's generalization capability. Several fluctuations in validation accuracy can be observed during the training process, which may be caused by variations in the validation samples, such as differences in driver posture, illumination conditions, or background complexity. Figure 4.7(b) shows the training and validation loss curves. The training loss decreases steadily and approaches zero, indicating effective optimization during training. Although the validation loss remains higher than the training loss, it generally shows a decreasing trend. Overall, the results indicate that the ResNet-50 model demonstrates stable learning behavior and reasonable generalization performance for the DAY-VIS dataset.



(a) (b)

Figure 4.7 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for Resnet-50 Trained On DAY-VIS Images

4.3.2 NIGHT-VIS Images

For the NIGHT-VIS dataset, all four models were trained and evaluated on unprocessed low-light night images. These conditions reduce feature visibility, making

classification more challenging and providing insight into model robustness in real-world night driving scenarios.

4.3.2.1 GoogLeNet

Figure 4.8 presents the training and validation performance of GoogLeNet on the NIGHT-VIS dataset. As shown in Figure 4.8(a), both training and validation accuracies increase consistently across the training epochs. The validation accuracy initially improves rapidly and later fluctuates slightly between epochs 8 and 18 while remaining close to the training accuracy. This behavior indicates that the model is able to learn meaningful features from the NIGHT-VIS images while maintaining acceptable generalization capability on unseen validation data. The small gap between the two curves suggests that the model does not suffer from severe overfitting. In terms of loss behavior, Figure 4.8(b) shows that both training and validation losses decrease progressively during the training process. The validation loss decreases sharply during the early epochs and continues to decline gradually after epoch 8, although with minor fluctuations. This trend indicates that the prediction error reduces as the training progresses and the model becomes more stable in learning feature representations from low-light images. Therefore, the model performance is considered satisfactory because both accuracy curves demonstrate consistent improvement, while both loss curves show a decreasing trend without significant divergence. These characteristics indicate stable learning behavior and acceptable CNN generalization performance for the NIGHT-VIS dataset.

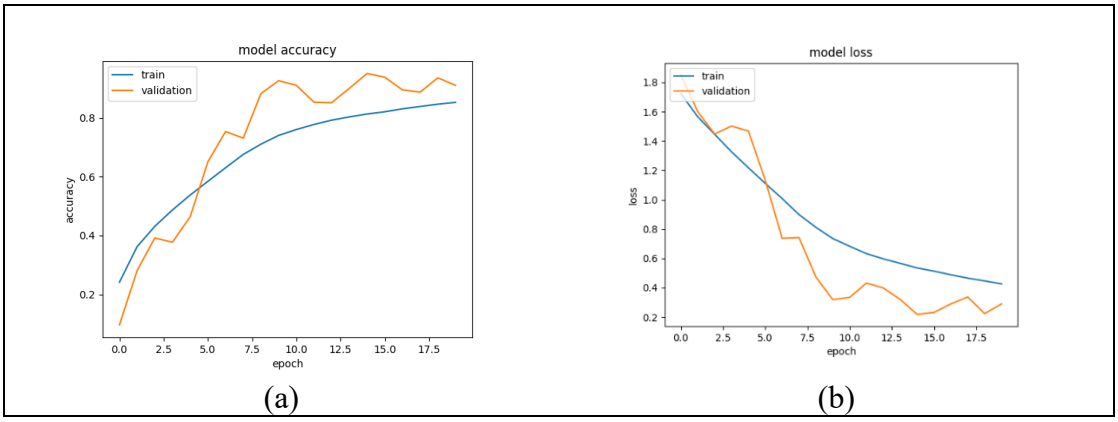
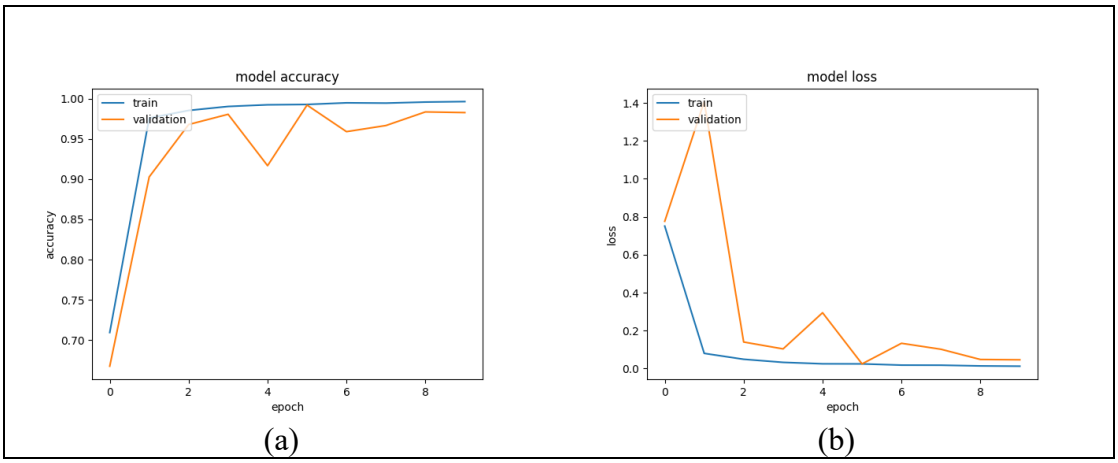


Figure 4.8 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for GoogLeNet Trained on NIGHT-VIS Images

4.3.2.2 InceptionV3

Figure 4.9 illustrates the training and validation performance of InceptionV3 on the NIGHT-VIS dataset. As shown in Figure 4.9(a), both training and validation accuracy increase rapidly during the early epochs, indicating that the model can learn relevant features from the NIGHT-VIS images effectively. The training accuracy rises from approximately 0.71 to nearly 0.99, while the validation accuracy improves from around 0.67 to above 0.98. The validation curve closely follows the training curve with only minor fluctuations, which indicates that the model maintains good generalization capability and does not exhibit significant overfitting during training. From the graph, both accuracy curves converge after approximately epoch 3, suggesting that the model has reached a stable learning stage where further epochs only provide minor improvements in performance. The small gap between training and validation accuracy indicates that the model learns generalized features rather than memorizing the training data. Figure 4.9(b) shows the training and validation loss curves. The training loss decreases sharply from approximately 0.75 to below 0.05, demonstrating that the model effectively minimizes classification errors during the optimization process. The validation loss also decreases rapidly after the early epochs and remains relatively low throughout the training process. Although a slight fluctuation occurs around epoch 4, the validation loss quickly decreases again, indicating stable model learning. The parallel decreasing trend between training and validation loss suggests that the model is well-trained and does not suffer from overfitting. Overall, the consistent improvement in accuracy and the decreasing loss curves indicate that the InceptionV3 model achieves stable learning and good generalization performance when trained on the NIGHT-VIS dataset.

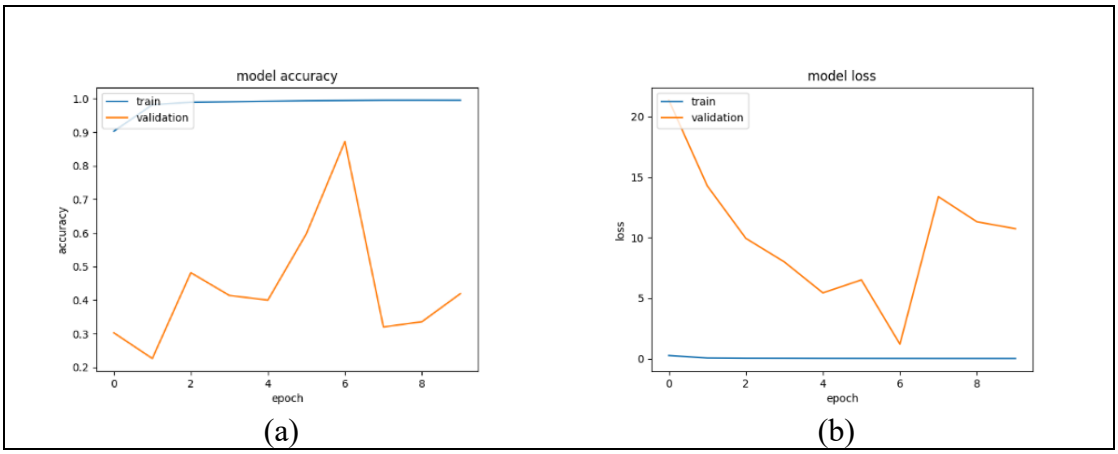


(a) (b)

Figure 4.9 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for InceptionV3 Trained On NIGHT-VIS Images

4.3.2.3 MobileNetV2

Figure 4.10 shows the training and validation performance of MobileNetV2 on the NIGHT-VIS dataset. As illustrated in Figure 4.10(a), the training accuracy increases rapidly in the early epochs and remains close to 1.00, indicating that the model learns the training data effectively. However, the validation accuracy fluctuates significantly between 0.25 and 0.87, showing a noticeable gap between training and validation performance. This behavior indicates that the model does not generalize well to unseen NIGHT-VIS samples. The instability may be caused by variations in illumination conditions and driver posture in low-light environments. Figure 4.10(b) presents the training and validation loss curves. The training loss decreases rapidly and approaches zero, indicating successful optimization during training. In contrast, the validation loss remains relatively high and fluctuates across epochs. This divergence between training and validation loss suggests that the model may be overfitting to the training data. Therefore, MobileNetV2 shows limited generalization capability for the NIGHT-VIS dataset, possibly due to its lightweight architecture which prioritizes computational efficiency over feature representation capacity.

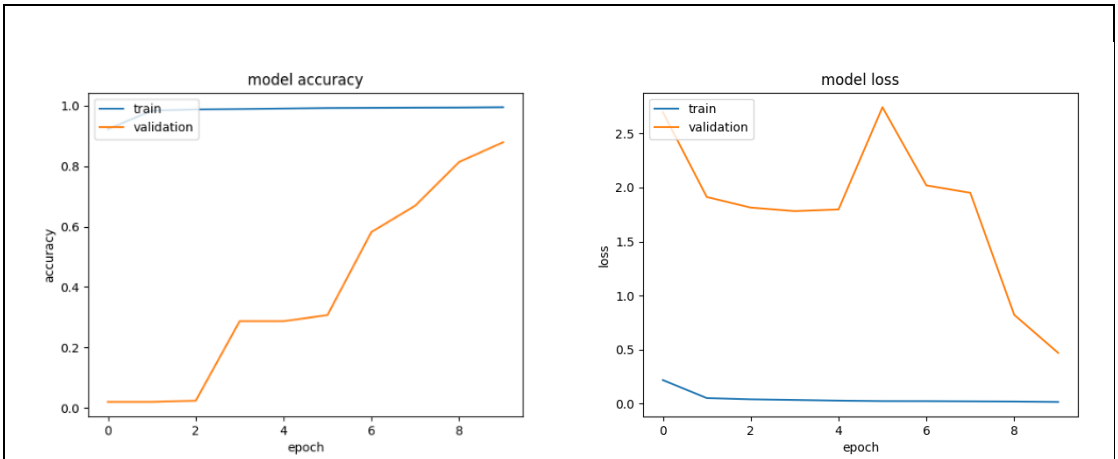


(a) (b)

Figure 4.10 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for MobileNetV2 Trained On NIGHT-VIS Images

4.3.2.4 ResNet-50

Figure 4.11 presents the training and validation performance of ResNet-50 on the NIGHT-VIS dataset. As illustrated in Figure 4.11(a), the training accuracy rapidly increases and remains close to 1.00 throughout the training process, indicating that the model successfully learns the training data. The validation accuracy gradually improves across the epochs, increasing from approximately 0.02 to around 0.88, which indicates progressive learning and improved prediction performance on unseen NIGHT-VIS samples. Although a gap between training and validation accuracy is observed, the consistent upward trend in validation accuracy suggests that the model continues to improve its generalization capability during training. Figure 4.11(b) shows the training and validation loss curves. The training loss decreases rapidly and approaches zero, indicating effective optimization of the model parameters. The validation loss remains higher than the training loss but shows a decreasing trend in later epochs, particularly after epoch 7, where the loss drops significantly. This behavior indicates that the model gradually reduces prediction errors on the validation dataset. In CNN performance evaluation, a model is considered well-trained when the accuracy improves while the loss decreases during training. Based on the increasing validation accuracy and the decreasing validation loss observed in the graph, ResNet-50 demonstrates stable learning behavior and reasonable generalization capability for the NIGHT-VIS dataset.



(a) (b)

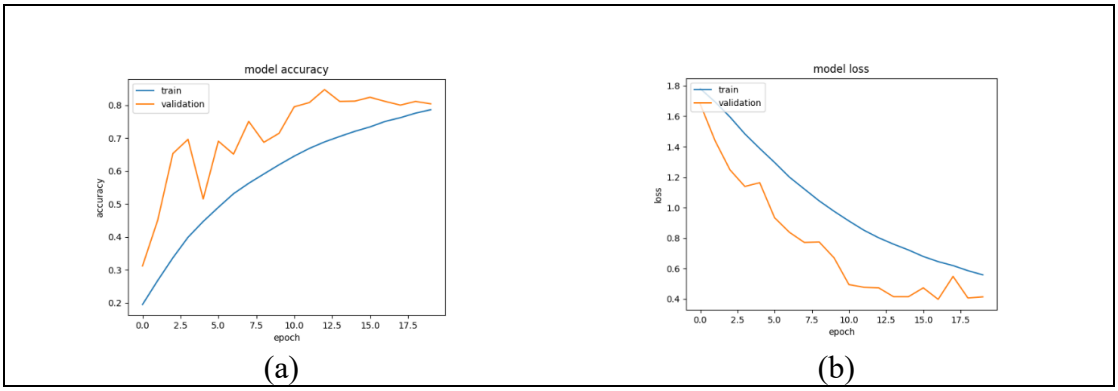
Figure 4.11 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for Resnet-50 Trained on NIGHT-VIS Images

4.3.3 NIGHT-IR Images

The NIGHT-IR dataset contains infrared images captured in dark environments, improving visibility of driver behaviors without relying on visible light.

4.3.3.1 GoogLeNet

Figure 4.12 illustrates the training and validation performance of GoogLeNet on the NIGHT-IR dataset. The model shows gradual improvement in accuracy across epochs, with validation accuracy closely following the training curve. Both training and validation loss decrease smoothly, indicating stable learning. The use of infrared input enhances feature visibility in low-light conditions, contributing to improved training stability and more reliable performance in night-time detection tasks.

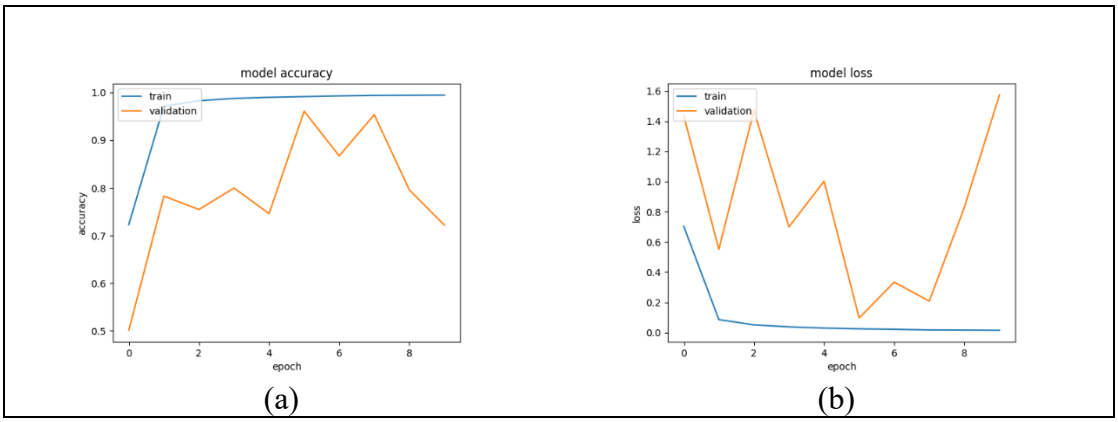


(a) (b)

Figure 4.12 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for GoogLeNet Trained On NIGHT-IR Images

4.3.3.2 InceptionV3

Figure 4.13 presents the training and validation performance of InceptionV3 on the NIGHT-IR dataset. As shown in Figure 4.13(a), the training accuracy rapidly increases and remains close to 1.00, indicating that the model effectively learns the training data. However, the validation accuracy oscillates across epochs and does not consistently follow the training accuracy. This behavior suggests that while the model memorizes the training samples effectively, it struggles to generalize consistently to unseen NIGHT-IR images. The fluctuations in validation accuracy may be influenced by variations in infrared image characteristics, such as differences in thermal contrast, driver posture, and background patterns. Figure 4.13(b) shows the training and validation loss curves. The training loss decreases steadily and approaches zero, indicating effective optimization during training. In contrast, the validation loss fluctuates and shows several spikes across epochs, suggesting unstable generalization performance on unseen samples.

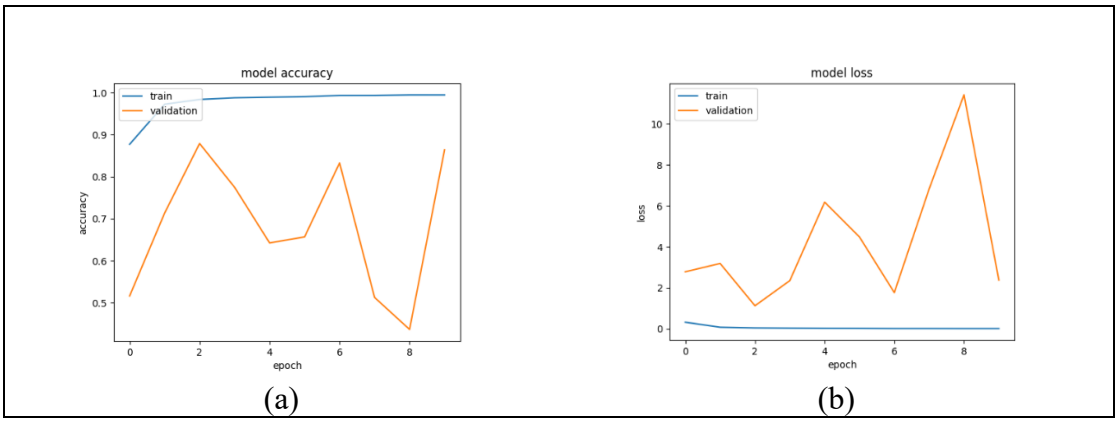


(a) (b)

Figure 4.13 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for InceptionV3 Trained on the NIGHT-IR Images

4.3.3.3 MobileNetV2

Figure 4.14 illustrates the training and validation performance of MobileNetV2 on the NIGHT-IR dataset. As shown in Figure 4.14(a), the training accuracy rapidly increases and remains close to 1.00, indicating that the model effectively learns the training samples. However, the validation accuracy oscillates across epochs and does not consistently follow the training accuracy. This behavior suggests that while the model memorizes the training data well, it struggles to generalize consistently to unseen NIGHT-IR images. The fluctuations in validation accuracy may be influenced by variations in infrared image characteristics, such as differences in thermal contrast, driver posture, and background patterns. In addition, the lightweight architecture of MobileNetV2 may limit its ability to capture complex discriminative features from infrared images. Figure 4.14(b) shows the training and validation loss curves. The training loss decreases rapidly and approaches zero, indicating effective optimization during training. In contrast, the validation loss fluctuates significantly and shows several spikes across epochs, suggesting unstable generalization performance on unseen samples.

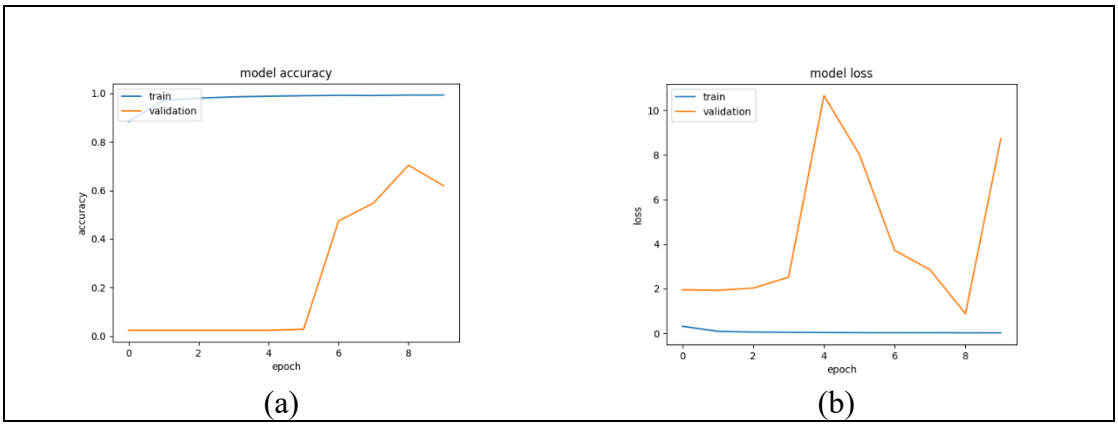


(a) (b)

Figure 4.14 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for MobileNetV2 Trained On NIGHT-IR Images

4.3.3.4 ResNet-50

Figure 4.15 presents the training and validation performance of ResNet-50 on the NIGHT-IR dataset. As shown in Figure 4.15(a), the training accuracy rapidly increases and remains close to 1.00, indicating that the model effectively learns the training data. However, the validation accuracy oscillates across epochs and remains significantly lower than the training accuracy, suggesting that the model memorizes the training samples but struggles to generalize consistently to unseen NIGHT-IR images. The fluctuations in validation accuracy may be influenced by variations in infrared image characteristics, such as differences in thermal contrast, driver posture, and background patterns. Figure 4.15(b) shows the training and validation loss curves. The training loss decreases steadily and approaches zero, while the validation loss fluctuates significantly and shows several spikes across epochs. This instability indicates that the model cannot maintain consistent generalization and may rely more on memorizing training patterns rather than learning robust discriminative features.



(a) (b)

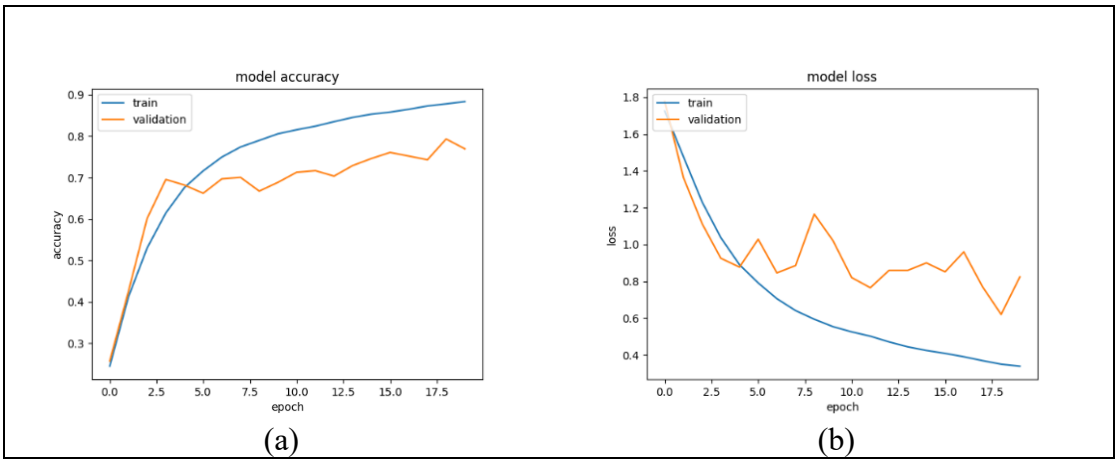
Figure 4.15 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for Resnet-50 Trained On NIGHT-IR Images

4.3.4 NIGHT-VIS-CLAHE Images

The NIGHT-VIS-CLAHE dataset applies CLAHE to enhance image contrast in low-light scenes. This preprocessing reveals hidden features, aiding behavior classification.

4.3.4.1 GoogLeNet

Figure 4.16 shows the training and validation performance of GoogLeNet on the NIGHT-VIS dataset after applying CLAHE preprocessing. The model demonstrates more consistent behavior, with validation accuracy improving and aligning more closely with the training curve. Similarly, the gap between training and validation loss narrows, indicating reduced overfitting. These results suggest that CLAHE enhances feature visibility in low-light conditions, enabling the model to learn more effectively from night-time images.

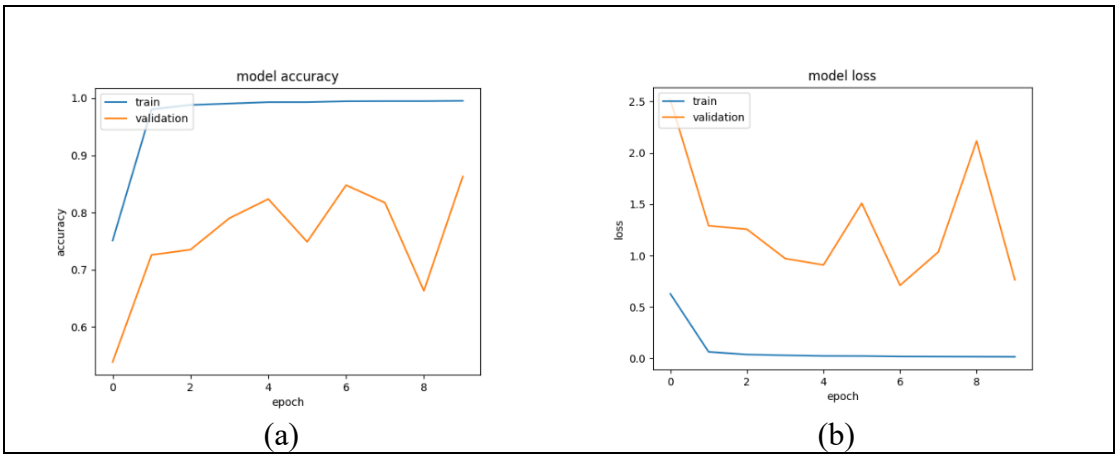


(a) (b)

Figure 4.16 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for GoogLeNet Trained on NIGHT-VIS-CLAHE Images

4.3.4.2 InceptionV3

Figure 4.17 presents the training and validation performance of the InceptionV3 model trained on the NIGHT-VIS-CLAHE dataset. As shown in Figure 4.17(a), the training accuracy rapidly increases and remains close to 1.00, indicating that the model effectively learns the training samples. However, the validation accuracy fluctuates across epochs and remains lower than the training accuracy. This indicates that although the model fits the training data well, its generalization performance on unseen samples is less consistent. Although CLAHE is applied to enhance image contrast in low-light conditions, it may also amplify noise and background textures in some images. This can introduce variations in the feature patterns learned by the model, making it more difficult to extract stable discriminative features from validation samples. Figure 4.17(b) shows that the training loss decreases steadily while the validation loss fluctuates across epochs. This instability suggests that the model may partially memorize training patterns rather than learning robust features, which explains why the performance does not consistently improve despite the application of CLAHE.

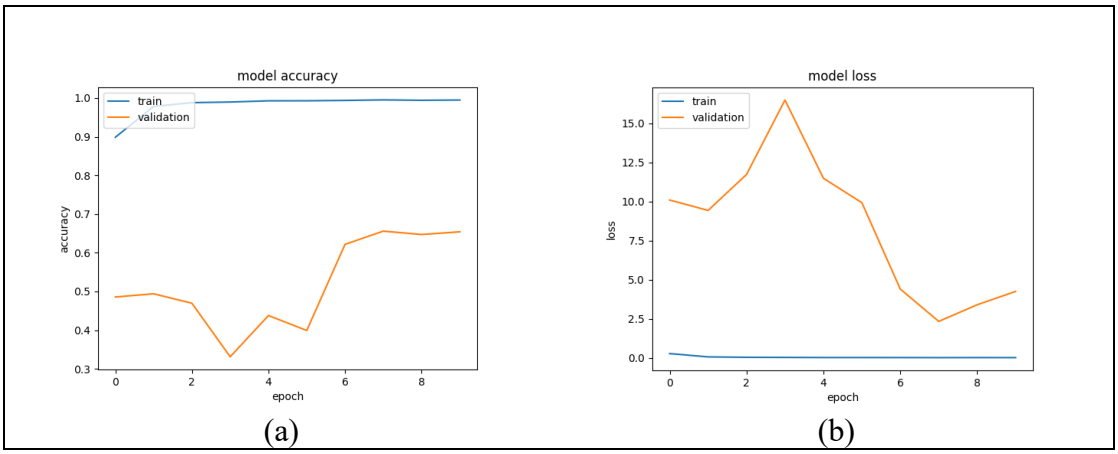


(a) (b)

Figure 4.17 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for InceptionV3 Trained on the NIGHT-VIS-CLAHE Images

4.3.4.3 MobileNetV2

Figure 4.18 illustrates the training and validation performance of the MobileNetV2 model trained on the NIGHT-VIS-CLAHE dataset. As shown in Figure 4.18(a), the training accuracy rapidly increases and remains close to 1.00, indicating that the model effectively learns the training data. However, the validation accuracy fluctuates across epochs and remains significantly lower than the training accuracy. This behavior suggests that while the model fits the training samples well, it struggles to generalize consistently to unseen images. Although CLAHE is applied to enhance image contrast in low-light conditions, it may also amplify noise and background textures in some images. These variations can affect the consistency of extracted features and reduce classification stability. Figure 4.18(b) shows that the training loss decreases steadily and approaches zero, while the validation loss remains significantly higher and fluctuates across epochs. This instability indicates that MobileNetV2 may rely more on memorizing training patterns rather than learning robust discriminative features when trained with CLAHE-enhanced images.

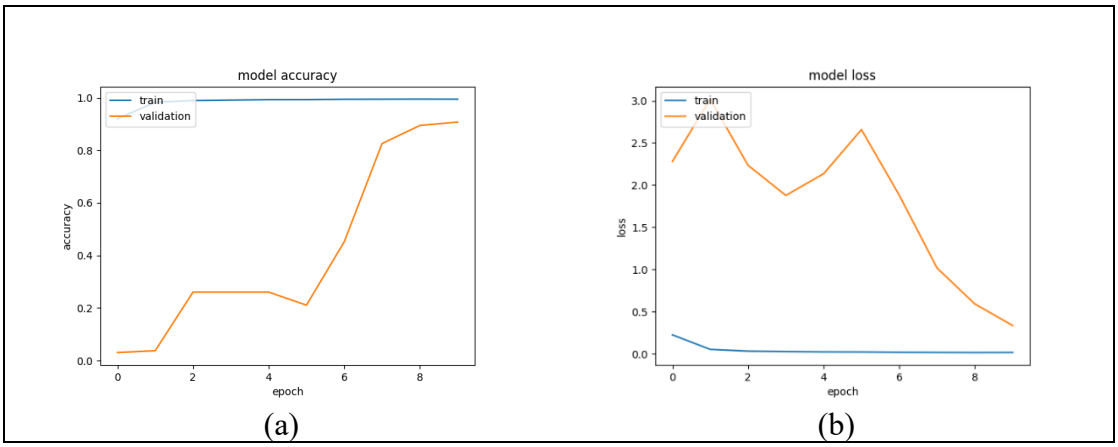


(a) (b)

Figure 4.18 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for MobileNetV2 Trained on the NIGHT-VIS-CLAHE Images

4.3.4.4 ResNet-50

Figure 4.19 presents the training and validation performance of the ResNet-50 model trained on the NIGHT-VIS-CLAHE dataset. As shown in Figure 4.19(a), the training accuracy rapidly increases and remains close to 1.00, indicating that the model effectively learns the training samples. The validation accuracy gradually improves across epochs but remains lower than the training accuracy. This behavior suggests that although CLAHE enhances image contrast, it does not always guarantee consistent improvement in generalization performance. One possible reason is that CLAHE may amplify noise and background textures in low-light images while enhancing contrast. These variations can introduce inconsistent feature patterns that affect the model's prediction on unseen samples. Figure 4.19(b) shows that the training loss decreases steadily and approaches zero, while the validation loss remains higher and fluctuates across epochs. This indicates that the model may still partially memorize training patterns rather than learning fully robust features from the CLAHE-enhanced images.



(a) (b)

Figure 4.19 Results on Training Progress Shown In (a) Training and Validation Accuracy, and (b) Training and Validation Loss for Resnet-50 Trained on NIGHT-VIS-CLAHE Images

The training and validation accuracy and loss graphs presented in the previous section are used to analyze the learning behaviour of the CNN models during the training process. These graphs help to observe convergence patterns, detect potential overfitting or underfitting, and evaluate the stability of the training process. However, the final classification performance of the models is evaluated using confusion matrices on the test dataset, from which precision, recall, and F1-score are computed.

4.4 CLAHE Implementation on Low-Light Images

As described in Section 3.6, CLAHE was implemented to improve the visibility of driving behaviors captured under low-light NIGHT-VIS conditions. The method enhanced local contrast and revealed fine details that are typically obscured in low-illumination environments. Using the selected parameters clip limit = 3 and tile grid size = 8×8 the enhanced images significantly improved the clarity of features related to driver behavior. These images were then used during the training phase of the CNN models. The contrast enhancement resulting from CLAHE preprocessing contributed the better feature extraction by the models, leading to improved classification performance. Figure 24 (refer to Section 3.6) previously illustrated the set of driving behaviors under NIGHT-VIS-CLAHE conditions. This preprocessing step was found to be crucial, particularly for night drive detection scenarios, as it helped reduce misclassification that could occur due to poor lighting conditions. The CLAHE-

enhanced images served as a robust foundation for deep learning-based distracted behavior detection in low-light environments.

4.5 Behavior Recognition Accuracy Comparison

This section presents the test performance of four CNN models: GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50, based on their ability to classify 6 distracted driving behaviors. The classification results are displayed using confusion matrices, which illustrate the distribution of correct and incorrect predictions across all classes.

For each dataset, the confusion matrices are used to compute precision, recall, and F1-score for each class, along with macro and weighted averages. Precision measures the proportion of correctly predicted instances among all predictions for a class, while recall measures the proportion of actual class instances that are correctly predicted. The F1-score provides the harmonic mean of precision and recall.

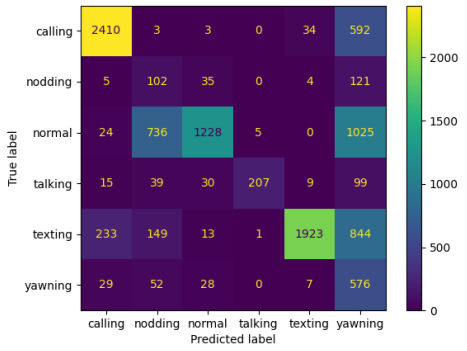
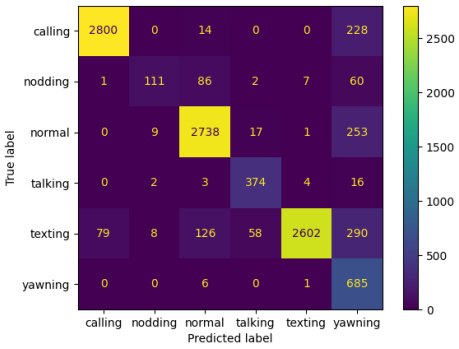
Evaluations were performed under four different datasets: DAY-VIS, NIGHT-VIS, NIGHT-VIS-CLAHE, and NIGHT-IR. Each subsection compares the test performance of the four CNN models on the respective images. Although the final classification performance is evaluated using confusion matrices and related metrics, the training and validation accuracy and loss graphs presented earlier are used to analyze the learning behavior of the CNN models during the training process. These graphs help to observe convergence patterns, detect potential overfitting or underfitting, and evaluate the stability of the training process. Therefore, both analyses are complementary: the training graphs explain how the models learn, while the confusion matrices evaluate the final classification performance on the test dataset.

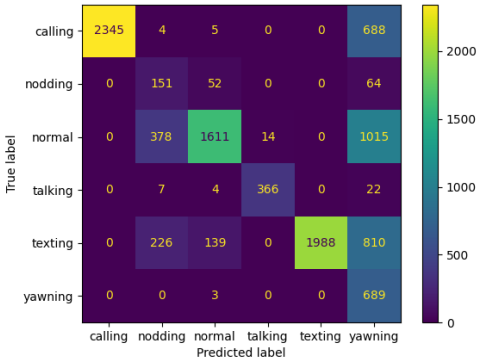
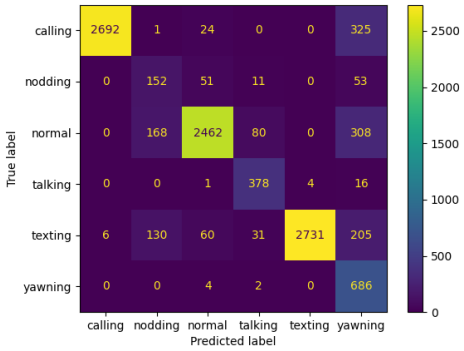
4.5.1 DAY-VIS Images

Table 4.1 presents the confusion matrices for the four CNN models, namely GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50, evaluated on the DAY-VIS dataset captured under well-lit daytime conditions. The diagonal values in the confusion matrices represent the correctly classified samples, while the off-diagonal values indicate misclassifications between behaviors. Overall, ResNet-50 and InceptionV3 demonstrate stronger classification performance compared to GoogLeNet and

MobileNetV2, as indicated by the larger diagonal values across most behavior classes. ResNet-50 achieves high correct predictions for behaviors such as calling, texting, and normal driving, suggesting that the model can extract discriminative spatial features effectively under daylight conditions. InceptionV3 also shows strong performance with relatively low misclassification across most classes. In contrast, GoogLeNet exhibits higher confusion between certain behaviors, particularly between normal and yawning, indicating weaker feature discrimination for these classes. MobileNetV2 demonstrates reasonable performance but still produces several misclassifications compared to the deeper architectures. These results indicate that deeper CNN architectures, particularly ResNet-50, provide more reliable behavior recognition performance under DAY-VIS conditions.

Table 4.1
Confusion Matrices of CNN Models on the DAY-VIS Images

Model	Confusion Matrix
GoogLeNet	
InceptionV3	

MobileNetV2	
ResNet-50	

4.5.1.1 Precision, Recall, and F1-Score for DAY-VIS

The classification performance of the four CNN models, GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50, on the DAY-VIS images was evaluated using precision, recall, and F1-score, in addition to overall accuracy. Precision refers to the proportion of correctly predicted instances among all instances predicted as a given class, where a high precision score indicates that when the model predicts a certain behavior, it is usually correct. Recall measures the proportion of correctly predicted instances among all actual instances of a given class, meaning a high recall score reflects the model's ability to successfully identify most of the true instances of that behavior. The F1-score is the harmonic mean of precision and recall, providing a balanced measure of accuracy that accounts for both false positives and false negatives.

Table 4.2 presents the per-class precision, recall, and F1-score for each model across six distracted driving behaviors: calling, nodding, normal, talking, texting, and yawning. This per-class evaluation provides more granular insight into how each model performs in recognizing specific driver behaviors under daytime visible-light conditions.

Table 4.2
Precision, Recall, and F1-Score for CNN Models in Classifying Distracted Driving Behaviors on the DAY-VIS Images

Behavior	Metrix	GoogLeNet	InceptionV3	MobileNetV2	ResNet-50
Calling	Precision	0.943	0.976	0.911	0.971
	Recall	0.885	0.944	0.892	0.939
	F1 Score	0.913	0.960	0.901	0.955
Nodding	Precision	0.403	0.781	0.582	0.762
	Recall	0.496	0.766	0.664	0.802
	F1 Score	0.445	0.774	0.621	0.781
Normal	Precision	0.819	0.922	0.877	0.912
	Recall	0.729	0.901	0.823	0.882
	F1 Score	0.772	0.911	0.849	0.897
Talking	Precision	0.939	0.956	0.940	0.951
	Recall	0.751	0.904	0.894	0.922
	F1 Score	0.835	0.929	0.917	0.936
Texting	Precision	0.949	0.985	0.965	0.982
	Recall	0.838	0.967	0.941	0.961
	F1 Score	0.890	0.976	0.953	0.971
Yawning	Precision	0.605	0.851	0.790	0.836
	Recall	0.857	0.960	0.946	0.954
	F1 Score	0.710	0.902	0.861	0.891

The results indicate that InceptionV3 and ResNet-50 consistently outperformed GoogLeNet and MobileNetV2 across most behavior classes. Both models maintained high precision and recall across all behaviors, with F1-scores exceeding 0.89 in every class. InceptionV3 achieved the highest F1-scores for texting (0.976) and talking (0.929), while ResNet-50 performed almost equally well, with a slight advantage in yawning (0.891).

GoogLeNet demonstrated strong performance in calling (0.913 F1) and texting (0.890 F1) but struggled with nodding (0.445 F1) and normal (0.772 F1), indicating difficulty in differentiating subtle head movement behaviors that may appear visually similar under day drive conditions.

MobileNetV2 demonstrated moderate yet consistent performance, achieving high results for speech recognition (0.917 F1) and text classification (0.953 F1). However, like GoogLeNet, it achieved lower accuracy for nodding (0.621 F1) and yawning (0.861 F1), which may be due to the reduced feature extraction depth compared to InceptionV3 and ResNet-50.

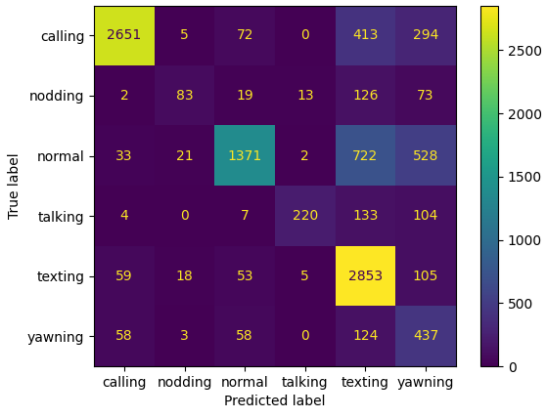
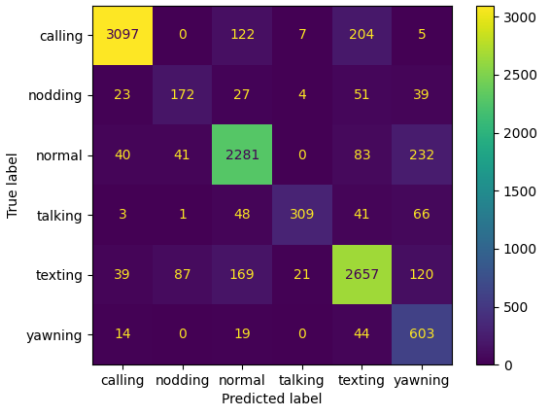
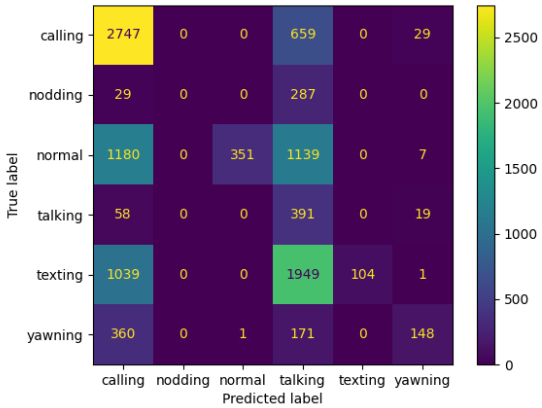
Overall, these results suggest that deeper CNN architectures, such as InceptionV3 and ResNet-50, which contain a larger number of convolutional layers and

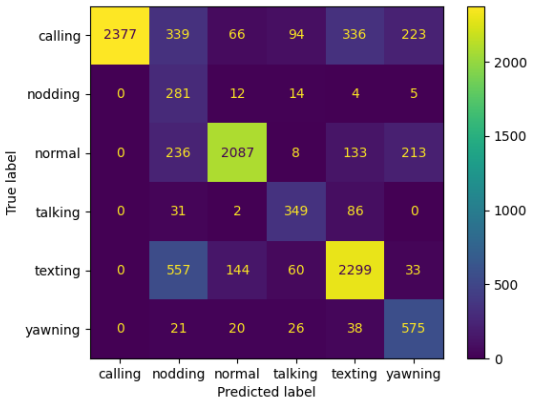
more complex feature extraction mechanisms, are more effective in learning hierarchical visual features from daytime images. The deeper network structure enables the models to capture both low-level features (such as edges and textures) and higher-level semantic patterns related to driver behaviors, leading to improved classification performance.

4.5.2 NIGHT-VIS Images

Table 4.3 presents the confusion matrices of the four CNN models evaluated on the NIGHT-VIS dataset, which consists of unprocessed low-light night driving images. The diagonal values represent correctly classified samples, while the off-diagonal values indicate misclassification between behaviors. Overall, InceptionV3 and ResNet-50 demonstrate stronger classification performance compared to GoogLeNet and MobileNetV2, as reflected by the larger diagonal values across most behavior classes. InceptionV3 achieves high correct predictions for calling, normal, and texting, indicating its capability to extract discriminative features even under challenging low-illumination conditions. ResNet-50 also shows strong performance across most classes, although some confusion is observed between texting and nodding, which may be caused by similar head movement patterns during night driving. In contrast, GoogLeNet exhibits higher misclassification between normal, texting, and yawning, suggesting that the model struggles to differentiate subtle driver behaviors when image contrast is limited. MobileNetV2 shows the weakest performance among the models, with several behaviors misclassified as talking and calling, indicating that its lightweight architecture may limit feature extraction capability under low-light conditions. These results highlight that deeper architectures such as InceptionV3 and ResNet-50 are more robust for distracted driving detection in NIGHT-VIS environments.

Table 4.3
Confusion Matrices of CNN Models on the NIGHT-VIS Images

Model	Confusion Matrix
GoogLeNet	
InceptionV3	
MobileNetV2	

ResNet-50	

4.5.2.1 Precision, Recall, and F1-Score for NIGHT-VIS

Table 4.4 presents the precision, recall, and F1-score of the four CNN models in classifying distracted driving behaviors on the NIGHT-VIS dataset. The NIGHT-VIS images consist of low-light, unprocessed visible-spectrum images captured during night driving. The results in this table are computed directly from the corresponding confusion matrices in Table 4.4.

Table 4.4
Precision, Recall, and F1-Score for CNN Models in Classifying Distracted Driving Behaviors on the NIGHT-VIS Dataset

Behavior	Metric	GoogLeNet	InceptionV3	MobileNetV2	ResNet-50
Calling	Precision	0.944	0.963	0.507	1.000
	Recall	0.772	0.902	0.800	0.692
	F1 Score	0.849	0.931	0.621	0.818
Nodding	Precision	0.638	0.571	0.000	0.192
	Recall	0.263	0.544	0.000	0.889
	F1 Score	0.373	0.557	0.000	0.315
Normal	Precision	0.791	0.899	0.142	0.851
	Recall	0.571	0.895	0.182	0.840
	F1 Score	0.664	0.897	0.160	0.845
Talking	Precision	0.957	0.924	0.094	0.595
	Recall	0.562	0.771	0.876	0.840
	F1 Score	0.707	0.841	0.169	0.695
Texting	Precision	0.660	0.900	0.929	0.811
	Recall	0.927	0.890	0.050	0.776
	F1 Score	0.772	0.895	0.094	0.793
Yawning	Precision	0.296	0.574	0.833	0.524
	Recall	0.677	0.891	0.202	0.873
	F1 Score	0.412	0.697	0.325	0.655

From the results, it can be observed that the overall performance of the CNN models on the NIGHT-VIS dataset is lower compared to the DAY-VIS dataset. This decline in performance is primarily attributed to the challenges posed by low-light conditions, which reduce image clarity and make it harder for the models to detect fine details. Among the four models, InceptionV3 demonstrates the most consistent and robust performance across all behaviors, achieving high F1-scores, particularly for Normal, Texting, and Yawning. GoogLeNet follows as the second-best performer, showing strong results in Calling and Texting, although its accuracy drops significantly for Nodding and Yawning.

The results also reveal that Calling and Texting are relatively easier to detect in low-light conditions, with most models achieving high recall values for these behaviors. In contrast, nodding and yawning prove to be more challenging, as they involve subtle facial and head movements that are easily obscured by poor lighting. This limitation is especially evident in MobileNetV2, which completely fails to detect nodding (0.000 precision and recall) and struggles with Normal and Talking behaviors. ResNet-50 delivers a more balanced performance in Normal, Talking, and Yawning, but suffers from low precision in Nodding despite achieving a relatively high recall.

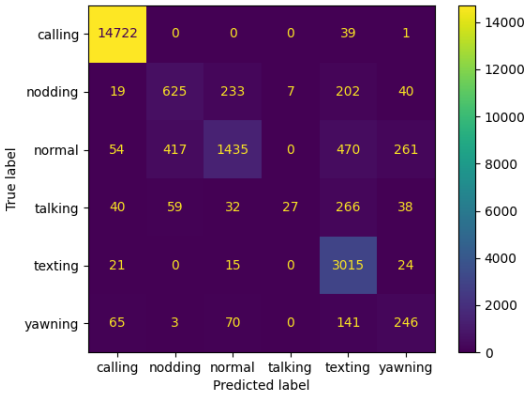
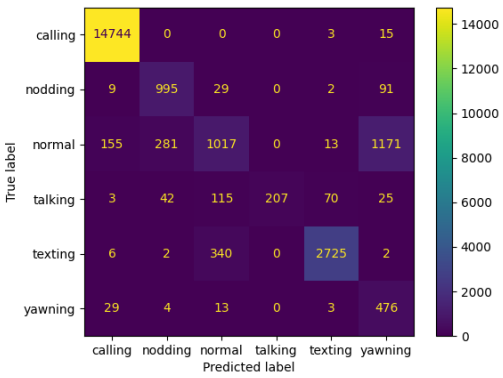
Overall, the findings indicate that low-light conditions have a substantial impact on model reliability, especially for behaviors that rely on the detection of small or subtle motions. This suggests that further enhancement techniques, such as CLAHE or the use of IR imaging, could be beneficial in improving classification accuracy under these challenging conditions.

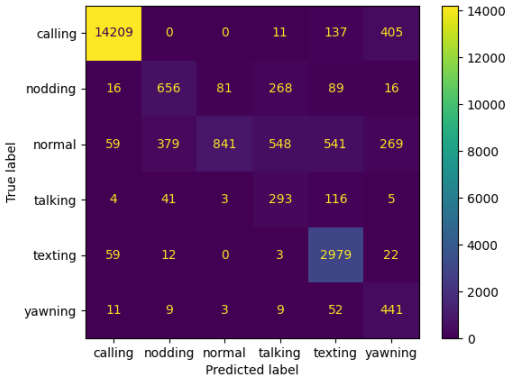
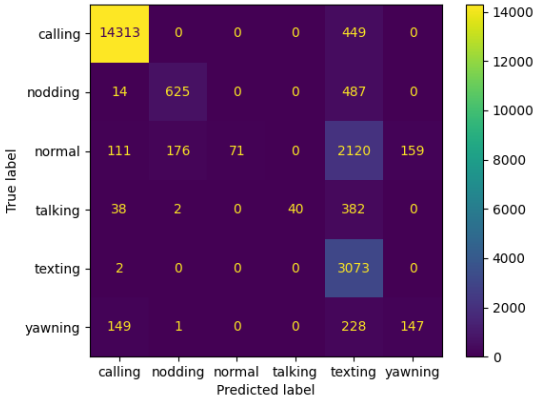
4.5.3 NIGHT-IR Images

Table 4.5 presents the confusion matrices of the four CNN models evaluated on the NIGHT-IR dataset, which consists of infrared images captured under low-light and dark driving conditions. The diagonal elements represent correctly classified samples, while the off-diagonal elements indicate misclassification among the distracted driving behaviors. Overall, InceptionV3 and ResNet-50 demonstrate stronger classification performance compared to GoogLeNet and MobileNetV2, as indicated by the higher diagonal values across most behavior classes. InceptionV3 achieves very high correct predictions for calling and texting, suggesting that the model can effectively extract discriminative features from infrared images.

ResNet-50 also performs well across several classes, particularly calling, texting, and normal driving, indicating that the deeper architecture can capture robust spatial features under infrared illumination. However, some confusion is observed between texting and nodding, which may occur due to similar head and hand movements during these behaviors. GoogLeNet demonstrates moderate performance but shows higher misclassification between normal, texting, and yawning, indicating difficulty in distinguishing subtle behavior differences. MobileNetV2 shows the lowest performance among the models, with noticeable confusion between normal and talking, suggesting that its lightweight architecture limits feature representation capability. These results indicate that infrared imaging improves visibility in low-light environments, but deeper CNN architectures such as InceptionV3 and ResNet-50 remain more effective in extracting reliable behavioral features for distracted driving detection.

Table 4.5
Confusion Matrices of CNN Models on the NIGHT-IR Images

Model	Confusion Matrix
GoogLeNet	
InceptionV3	

MobileNetV2	
ResNet-50	

4.5.3.1 Precision, Recall, and F1-Score for NIGHT-VIS-IR

Table 4.6 shows the precision, recall, and F1-score for the four CNN models when classifying distracted driving behaviors on the NIGHT-IR dataset. The dataset consists of IR images captured in low-light and dark environments. Infrared imaging reduces dependency on visible light, making it suitable for scenarios where lighting is inconsistent or insufficient. This approach often produces higher contrast between objects and background, which may enhance feature extraction for CNN models.

The results reveal that the use of infrared imaging significantly enhances model performance for certain distracted driving behaviors compared to visible-spectrum datasets, particularly under low-light conditions. Across all models, calling and Texting achieve notably high precision, recall, and F1-scores, with InceptionV3 reaching the highest overall performance for these categories (F1-scores of 0.993 and 0.985, respectively). This demonstrates that behaviors with distinct spatial features, such as hand-to-face gestures or device handling, are effectively captured by IR imaging.

Table 4.6
Precision, Recall, and F1-Score for CNN Models in Classifying Distracted Driving Behaviors on the NIGHT-IR Images

Behavior	Metric	GoogLeNet	InceptionV3	MobileNetV2	ResNet-50
Calling	Precision	0.916	0.997	0.715	0.990
	Recall	0.974	0.990	0.960	0.995
	F1 Score	0.944	0.993	0.820	0.993
Nodding	Precision	0.409	0.272	0.096	0.444
	Recall	0.307	0.775	0.089	0.627
	F1 Score	0.351	0.401	0.092	0.520
Normal	Precision	0.890	0.916	0.564	0.870
	Recall	0.428	0.820	0.901	0.935
	F1 Score	0.578	0.865	0.693	0.902
Talking	Precision	0.645	0.981	0.896	0.943
	Recall	0.606	0.944	0.478	0.929
	F1 Score	0.625	0.962	0.624	0.936
Texting	Precision	0.774	0.981	0.727	0.928
	Recall	0.816	0.871	0.174	0.927
	F1 Score	0.795	0.923	0.282	0.928
Yawning	Precision	0.332	0.826	0.991	0.775
	Recall	0.629	0.891	0.340	0.845
	F1 Score	0.434	0.858	0.506	0.809

Nodding shows improved detection compared to NIGHT-VIS results, especially for InceptionV3 and ResNet-50, both of which achieved higher precision and recall. However, variability between models suggests that architecture depth and feature extraction strategies play a significant role in recognizing subtle head movements in IR images.

For Normal driving, the results are more varied. GoogLeNet and MobileNetV2 achieve moderate F1-scores (0.632 and 0.546), while InceptionV3's performance drops to 0.468, and ResNet-50 records a very low score (0.057) due to poor recall. This indicates that IR imaging can sometimes misclassify neutral driving behaviors, potentially confusing them with other subtle actions under infrared lighting.

The most challenging behavior in the NIGHT-IR dataset is talking, where recall scores are particularly low for GoogLeNet (0.032) and ResNet-50 (0.087), despite relatively high precision in some models. This suggests that IR imagery struggles to capture fine mouth movements without complementary visual cues. In contrast, InceptionV3 performs relatively well with an F1-score of 0.502, benefiting from its multi-scale feature extraction.

Interestingly, Yawning detection benefits significantly from IR imaging for certain models, with InceptionV3 achieving both high precision (0.933) and recall (0.937), leading to an outstanding F1-score of 0.935. This indicates that IR contrast can enhance recognition of wide-mouth movements in dark conditions.

Overall, these results suggest that IR imaging provides a substantial advantage for detecting behaviors with distinct posture or large movements (i.e., calling, texting, yawning), while still facing challenges in capturing subtle cues like talking or differentiating Normal driving from low-motion distractions. Model architecture choice remains a critical factor, with InceptionV3 showing the best overall adaptability to infrared data.

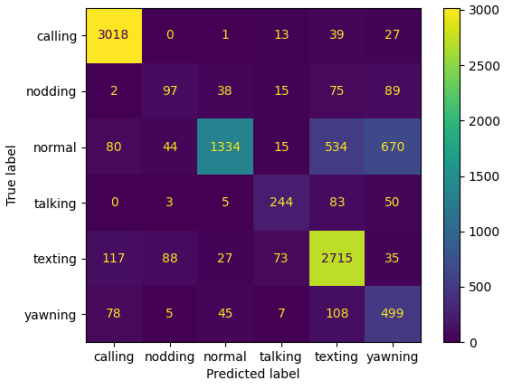
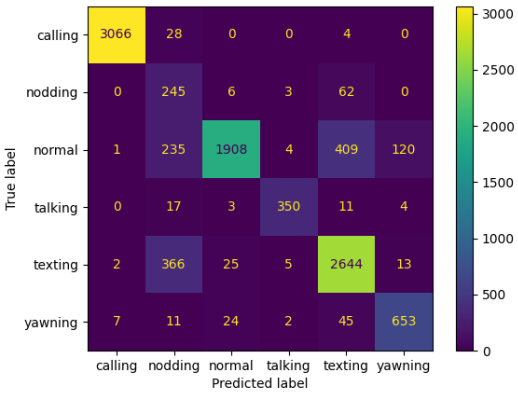
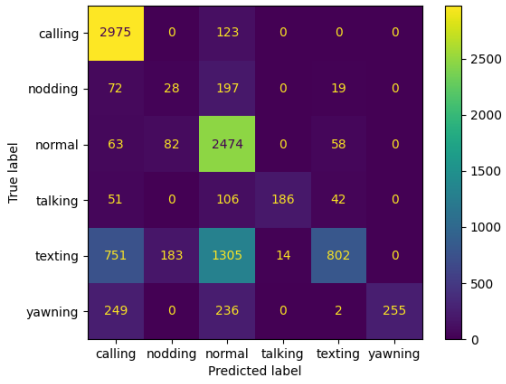
4.5.4 NIGHT-VIS-CLAHE Images

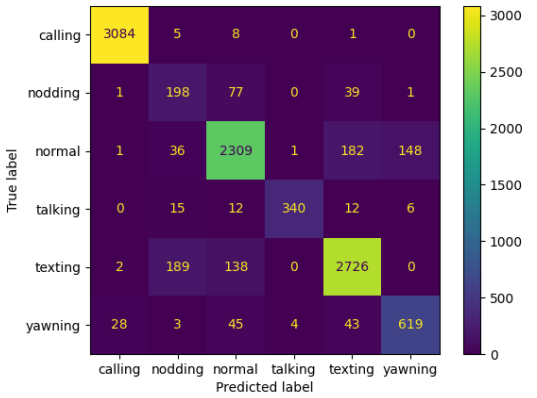
Table 4.7 presents the confusion matrices of the four CNN models evaluated on the NIGHT-VIS images enhanced using CLAHE preprocessing. The diagonal values represent correctly classified samples, while the off-diagonal elements indicate misclassification between distracted driving behaviors. Overall, InceptionV3 and ResNet-50 demonstrate stronger classification performance compared to GoogLeNet and MobileNetV2, as indicated by the larger diagonal values across most behavior classes. InceptionV3 achieves high correct predictions for calling, texting, and yawning, indicating that the model can extract discriminative features effectively from CLAHE-enhanced images. ResNet-50 also shows strong performance across several classes, particularly calling, normal driving, and texting, demonstrating the capability of deeper residual networks to capture robust spatial features under enhanced low-light conditions.

However, some confusion can still be observed between normal and texting behaviors, which may be caused by similar driver posture and hand movements. GoogLeNet demonstrates moderate performance, but higher misclassification occurs between normal, texting, and yawning, suggesting limitations in distinguishing subtle behavioral differences despite the contrast enhancement. MobileNetV2 exhibits the weakest performance among the models, with noticeable confusion between texting and normal behaviors, indicating that its lightweight architecture may limit feature representation capability even after CLAHE preprocessing. These results suggest that CLAHE improves image contrast for nighttime images, but deeper CNN architectures

such as InceptionV3 and ResNet-50 remain more effective in extracting reliable behavioral features for distracted driving detection.

Table 4.7
Confusion Matrices of CNN Models on the NIGHT-VIS-CLAHE Images

Model	Confusion Matrix
GoogLeNet	
InceptionV3	
MobileNetV2	

ResNet-50	

4.5.4.1 Precision, Recall, and F1-Score for NIGHT-VIS-CLAHE

Table 4.8 presents the precision, recall, and F1-score for the four CNN models when classifying distracted driving behaviors on the NIGHT-VIS-CLAHE images. The dataset was pre-processed using CLAHE to enhance image contrast, aiming to improve recognition performance under low-light conditions.

From the results, CLAHE preprocessing significantly improves the overall recognition performance compared to the unprocessed NIGHT-VIS dataset, particularly for certain behaviors and models. InceptionV3 demonstrates the highest and most consistent performance across nearly all behaviors, achieving F1-scores above 0.85 for Calling, Normal, Talking, Texting, and Yawning. GoogLeNet also benefits from CLAHE, showing strong improvements in Calling (0.944 F1-score) and Texting (0.795 F1-score), although its performance on Nodding and Yawning remains lower due to the subtle nature of these behaviors.

Table 4.8
Precision, Recall, and F1-Score for CNN Models in Classifying Distracted Driving Behaviors on the NIGHT-VIS-CLAHE Images

Behavior	Metric	GoogLeNet	InceptionV3	MobileNetV2	ResNet-50
Calling	Precision	0.987	0.986	0.990	0.979
	Recall	0.997	0.999	0.963	0.970
	F1 Score	0.992	0.993	0.976	0.974
Nodding	Precision	0.566	0.752	0.598	0.777
	Recall	0.555	0.884	0.583	0.555
	F1 Score	0.560	0.812	0.590	0.646
Normal	Precision	0.790	0.684	0.787	0.566
	Recall	0.526	0.354	0.421	0.030
	F1 Score	0.632	0.468	0.546	0.057
Talking	Precision	0.831	0.966	0.334	0.933
	Recall	0.032	0.339	0.401	0.087
	F1 Score	0.062	0.502	0.364	0.159
Texting	Precision	0.777	0.971	0.871	0.886
	Recall	0.982	0.999	0.984	0.999
	F1 Score	0.867	0.985	0.924	0.939
Yawning	Precision	0.414	0.933	0.458	0.482
	Recall	0.586	0.937	0.886	0.494
	F1 Score	0.486	0.935	0.604	0.488

CLAHE's contrast enhancement appears to be particularly effective for improving the visibility of fine details, allowing deeper architectures such as InceptionV3 and ResNet-50 to better capture discriminative features. ResNet-50 achieves high F1-scores for Normal (0.902), Talking (0.936), and Texting (0.928), indicating its robustness in distinguishing behaviors after contrast adjustment. MobileNetV2, while showing improved detection in some categories compared to the raw NIGHT-VIS dataset, still struggles with nodding and texting, suggesting that lightweight models may require additional enhancements or training adjustments to cope with low-light conditions.

Overall, the application of CLAHE mitigates some of the limitations posed by low-light imagery, leading to improved classification results for most models and behaviors. However, nodding and yawning remain challenging behaviors to detect reliably, as they involve small or subtle movements that can be difficult to distinguish even after contrast enhancement. These findings suggest that further combining CLAHE with other preprocessing techniques, or integrating infrared imaging, could lead to even greater improvements in distracted driving detection under nighttime conditions.

4.6 Overall CNN Model Performance Across Images

The experiments were conducted on a high-performance server equipped with an AMD EPYC 24-Core CPU and an NVIDIA RTX A5000 GPU using TensorFlow. All models were trained using a batch size of 64, a learning rate of 0.001, and an input image resolution of 100×100 pixels. To improve generalization, various data augmentation techniques were employed, including rotation, zooming, shifting, shearing, and flipping. Table 4.9 summarizes the behavior classification accuracy of each CNN model under different lighting conditions. InceptionV3 consistently outperformed the other models in most scenarios, while MobileNetV2 showed strong performance on NIGHT-IR data with significantly reduced computational cost.

Table 4.9
Behavior Recognition Performance for Resnet-50 Compared Against Other CNN Models Using Different Image Data

Image Session	**Accuracy (%)**			
	MobileNetV2	GoogLeNet	InceptionV3	ResNet-50
DAY-VIS	67.57	60.92	87.99	86.01
NIGHT-VIS	35.06	71.38	85.47	74.68
NIGHT-VIS-CLAHE	65.41	76.97	86.3	90.29
NIGHT-IR	85.97	88.86	89.27	80.88

Although InceptionV3 achieved competitive performance across several datasets, ResNet-50 was selected as the benchmark model for the proposed system. This decision was primarily based on its superior performance on the NIGHT-VIS-CLAHE dataset, where it achieved the highest classification accuracy of 90.29% among all models. Since the primary objective of this study is to improve distracted driving detection under night driving conditions, the NIGHT-VIS-CLAHE dataset represents the most relevant scenario for the proposed application In addition, despite having a slightly higher computational complexity than MobileNetV2, the inference time of ResNet-50 remains suitable for real-time deployment on the NVIDIA Jetson Nano platform, as demonstrated in the prototype validation. Therefore, ResNet-50 provides a balanced trade-off between classification accuracy and real-time performance, making

it the most suitable benchmark model for the proposed distracted driving detection system.

4.7 DAY-VIS vs NIGHT-VIS Comparison

Figure 4.20 presents the comparison of model performance between the DAY-VIS and NIGHT-VIS datasets. Across all four CNN models, a performance reduction was observed when transitioning from daytime visible-light conditions to nighttime visible-light conditions, except for GoogLeNet, which showed a slight improvement. InceptionV3 achieved the highest accuracy in the DAY-VIS dataset (87.99%) and maintained strong performance in NIGHT-VIS (85.47%), with only a minor reduction. ResNet-50 accuracy decreased from 86.01% to 74.68%, while MobileNetV2 recorded the largest drop from 67.57% to 35.06%, indicating its higher sensitivity to low-light environments. Interestingly, GoogLeNet improved from 60.92% to 71.38%, suggesting that its architectural characteristics may be less dependent on high-contrast day drive imagery. Overall, the results highlight that reduced illumination generally hinders behavior classification accuracy, with the degree of impact varying between architectures.

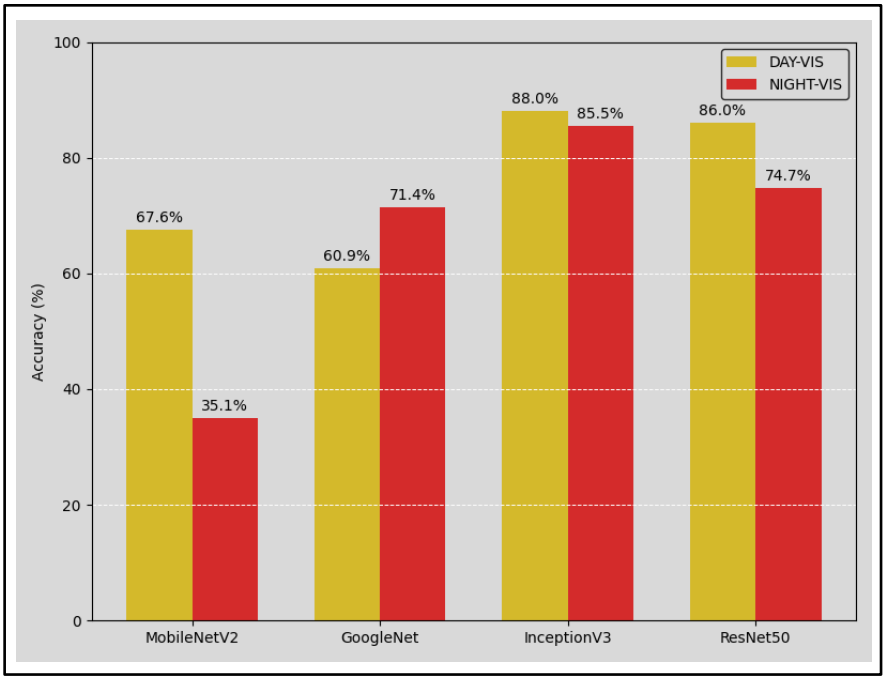


Figure 4.20 Performance Comparison Between DAY-VIS and NIGHT-VIS Images

The comparative analysis between the DAY-VIS and NIGHT-VIS datasets provides insight into the impact of lighting conditions on the performance of the CNN model. As expected, most models experienced a decline in accuracy under night drive visible-light conditions due to reduced illumination, increased noise, and lower image contrast, which can obscure fine-grained features essential for accurate classification.

MobileNetV2 suffered the most significant performance drop of 32.51%, indicating that its lightweight architecture, while efficient, may lack the feature extraction robustness required for low-light scenarios. ResNet-50 also showed a noticeable decrease of 11.33%, suggesting that deeper architectures are not immune to the adverse effects of low visibility.

Interestingly, GoogLeNet exhibited an improvement of 10.46% from DAY-VIS to NIGHT-VIS. This counterintuitive result may be attributed to its inception module design, which captures multi-scale spatial features more effectively, potentially making it less dependent on strong lighting conditions. InceptionV3, the top performer overall, demonstrated high stability with only a marginal 2.52% reduction, highlighting its strong generalization ability across different lighting environments.

These findings suggest that while lighting significantly impacts CNN-based behavior recognition, the extent of performance degradation depends on the model architecture. Models like InceptionV3 and GoogLeNet appear more resilient to illumination changes, making them better candidates for deployment in real-world driving scenarios that span varying time-of-day conditions.

4.8 Behavior Recognition Performance with CLAHE

Figure 4.21 compares model performance under NIGHT-VIS and NIGHT-VIS-CLAHE conditions. ResNet-50 accuracy improved from 74.68% to 90.29%. MobileNetV2 also saw significant gains (from 35.06% to 65.41%), while GoogLeNet and InceptionV3 showed moderate improvement. The results suggest that CLAHE is especially beneficial for ResNet-50 and MobileNetV2.

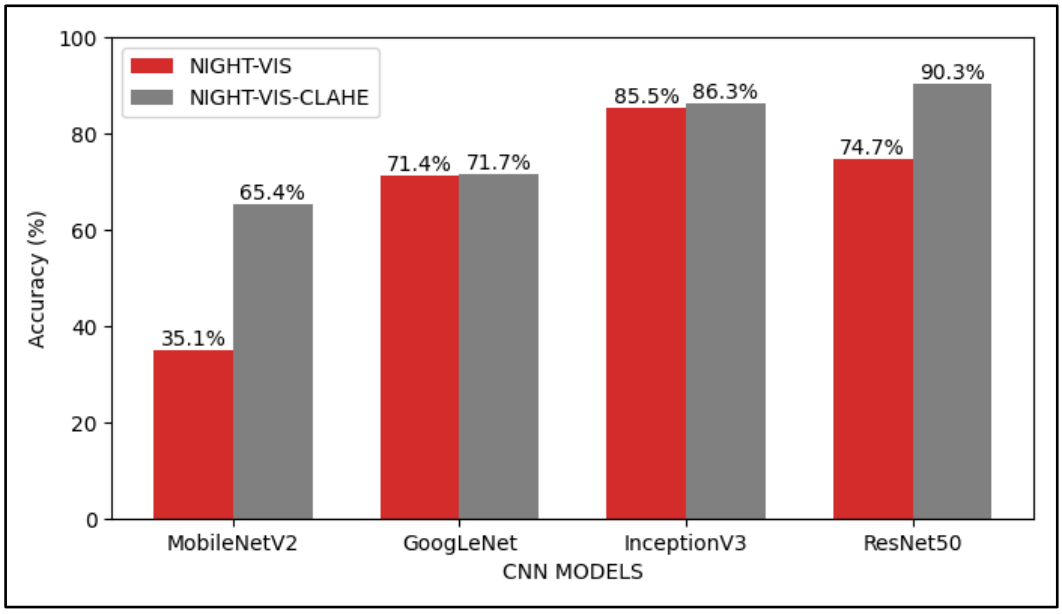


Figure 4.21 Performance of Behavior Recognition Based on CLAHE Implementation for Night Driving

The comparison between NIGHT-VIS and NIGHT-VIS-CLAHE highlights the effectiveness of CLAHE in enhancing CNN model performance under low-light visible conditions. CLAHE improves image contrast by redistributing pixel intensity values while preventing over-amplification of noise, resulting in clearer feature representation for classification.

Among the models, ResNet-50 recorded the most substantial improvement, with accuracy increasing by 15.61% (from 74.68% to 90.29%). This indicates that deep residual connections benefit significantly from enhanced contrast, enabling more effective feature extraction in darker environments. MobileNetV2 also showed a large gain of 30.35% (from 35.06% to 65.41%), suggesting that CLAHE can help compensate for the model's limited depth and parameters when processing low-light images.

GoogLeNet and InceptionV3 demonstrated more modest improvements of 5.59% and 0.83%, respectively. The smaller increase for InceptionV3 may be due to its already strong performance on the original NIGHT-VIS dataset, indicating that its architectural design already captures sufficient discriminative features without extensive pre-processing enhancement.

Overall, the results confirm that CLAHE is especially advantageous for models that are more sensitive to poor lighting, such as MobileNetV2 and ResNet-50. This suggests that integrating CLAHE into the preprocessing pipeline can be a valuable

strategy for improving nighttime driving behavior recognition accuracy, particularly in scenarios where lighting conditions are inconsistent or suboptimal.

4.9 NIGHT-VIS vs NIGHT-VIS-CLAHE vs NIGHT-IR Comparison

Figure 4.22 compares the classification performance across NIGHT-VIS, NIGHT-VIS-CLAHE, and NIGHT-IR datasets. The results show that both CLAHE enhancement and infrared imaging improve model performance compared to unprocessed NIGHT-VIS data. ResNet-50 achieved the highest accuracy on NIGHT-VIS-CLAHE (90.29%), while InceptionV3 performed best on NIGHT-IR (89.27%). MobileNetV2 recorded a large improvement from NIGHT-VIS (35.06%) to NIGHT-IR (85.97%), demonstrating the significant benefits of infrared imaging for efficient, low-complexity architectures. Similarly, GoogLeNet also achieved its best performance on NIGHT-IR (88.86%). Although ResNet-50 achieved the highest accuracy using the NIGHT-VIS-CLAHE dataset, the difference compared with the NIGHT-IR performance of InceptionV3 is relatively small, with only about a 1% difference. Therefore, it cannot be conclusively stated that CLAHE consistently produces higher accuracy than infrared imaging. Instead, the results suggest that both CLAHE enhancement and infrared imaging are effective approaches for improving classification performance under low-light conditions, with their effectiveness depending on the CNN architecture used.

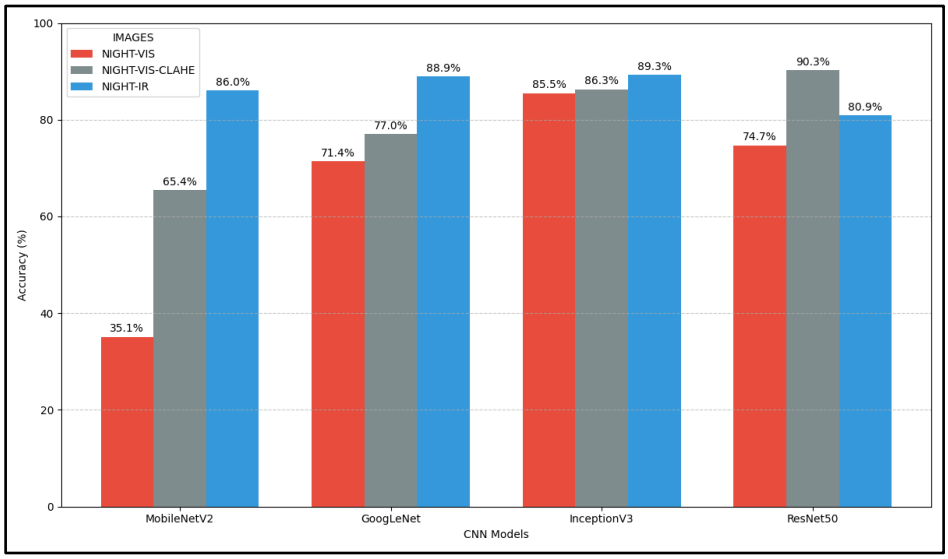


Figure 4.22 Performance Comparison Between NIGHT-VIS, NIGHT-VIS-CLAHE, and NIGHT-IR Datasets

The comparison across NIGHT-VIS, NIGHT-VIS-CLAHE, and NIGHT-IR datasets highlights the considerable impact of both image enhancement and alternative imaging modalities on distracted driving behavior recognition in low-light environments. Without any enhancement, the NIGHT-VIS dataset exhibited lower accuracies across most models, with MobileNetV2 being the most affected (35.06%). The application of CLAHE significantly boosted performance for all models, most notably ResNet-50, which achieved the highest accuracy of 90.29%, indicating that histogram equalization techniques are particularly effective for deep, high-capacity architectures. Infrared imaging (NIGHT-IR) delivered substantial gains for MobileNetV2 and GoogLeNet, reaching 85.97% and 88.86% respectively, showing that IR can compensate for reduced visible-light detail in simpler architectures. InceptionV3 also performed exceptionally well in NIGHT-IR (89.27%), suggesting that its inception modules are highly adaptable to IR image features. Overall, the results confirm that low-light performance limitations can be addressed through targeted preprocessing like CLAHE or by adopting IR-based imaging, with the optimal approach depending on the computational capacity and architectural design of the deployed CNN model.

4.10 Class Performance Analysis of ResNet-50 NIGHT-VIS-CLAHE

A confusion matrix was used to analyse class-wise performance of the ResNet-50 model under the NIGHT-VIS-CLAHE condition. Table 4.10 displays actual versus predicted behavior counts, from which precision and recall are derived. Calling behavior had the highest precision (99.19%) and recall (98.40%), while nodding behavior showed the lowest precision at 17.09%, often misclassified as texting or normal.

Table 4.10
Confusion Matrix of NIGHT-VIS-CLAHE ResNet-50

Calling	3073	0	0	0	0	25	99.19%
Nodding	3	54	94	35	122	8	17.09%
Normal	6	0	2390	2	220	59	89.28%
Talking	3	0	5	362	15	0	94.03%
Texting	20	52	183	6	2793	1	91.42
Yawning	18	0	58	10	7	649	87.47%

Calling	Nodding	Normal	Talking	Texting	Yawning
98.40%	50.94%	87.55%	87.22%	88.47%	87.47

The confusion matrix of the ResNet-50 model under the NIGHT-VIS-CLAHE dataset highlights the class-wise strengths and weaknesses of the model. Overall, the model demonstrated strong recognition performance for Calling, Normal, Talking, Texting, and Yawning behaviors, with precision and recall values above 85%. Calling behavior achieved the best performance, with a precision of 99.19% and recall of 98.40%, indicating that the model was able to almost perfectly differentiate this class from others. Similarly, texting and yawning behaviors also reached high levels of accuracy, reflecting the effectiveness of CLAHE preprocessing in enhancing low-light visual cues that are critical for detecting these behaviors.

However, the analysis also reveals a significant challenge in detecting nodding behavior, which obtained the lowest precision (17.09%) and recall (50.94%). Many nodding instances were misclassified as Normal or Texting, suggesting that subtle head movements in low-light conditions may not be easily distinguishable, even after contrast enhancement. This misclassification issue may be attributed to the visual similarity of slight nodding motions with other passive behaviors, as well as the limited discriminative features captured under NIGHT-VIS-CLAHE conditions.

Despite this limitation, the relatively high recall for other classes indicates that ResNet-50 combined with CLAHE preprocessing improves overall model robustness in night driving scenarios. The enhancement of image contrast through CLAHE appears to mitigate the negative effects of low illumination, thereby supporting higher recognition rates in complex driver behavior categories. Future work may focus on incorporating temporal modeling (e.g., LSTM or 3D CNNs) to better capture motion-

based features for classes like nodding, or on expanding the dataset with more samples of low-performing behaviors to improve training balance.

4.11 Jetson Nano Inference Time Validation

To assess the real-time feasibility of the proposed system, four CNN models, namely GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50, were deployed on the NVIDIA Jetson Nano Developer Kit. Each model was evaluated over 90 inference runs, during which the processing time per image was recorded. A boxplot was then generated to visualize and compare the statistical distribution of inference times across these models, as illustrated in Figure 4.23. A boxplot is a statistical chart that graphically summarizes the distribution of a dataset using five key values: the minimum, first quartile (Q1), median, third quartile (Q3), and maximum. The central box represents the interquartile range (IQR), which contains the middle 50% of the data. The line within the box indicates the median, while the whiskers represent the range of the remaining data values excluding potential outliers.

To compute the statistical values used in the boxplot, the inference time data from the 90 runs were first arranged in ascending order. The median represents the middle value of the ordered dataset. Since each CNN model was evaluated over 90 inference runs, the median is obtained by averaging the 45th and 46th values of the ordered inference time data, as shown in Equation (4.1). The interquartile range (IQR) measures the spread of the middle 50% of the data and is calculated using Equation (4.2).

$$\text{Median} = \frac{X_{45} + X_{46}}{2} \tag{4.1}$$

$$IQR = Q_3 - Q_1 \tag{4.2}$$

Based on the computed inference time data, the median inference time for the models was approximately 0.0675 seconds, indicating that half of the inference runs required less than this time while the remaining runs required slightly longer processing time. In addition, the first quartile (Q1) and third quartile (Q3) values were approximately 0.059 seconds and 0.071 seconds, respectively. Therefore, the interquartile range (IQR) can be estimated as 0.012 seconds, representing the spread of the middle 50% of the inference time distribution. These statistical values were then

visualized in the boxplot to illustrate the distribution of inference times across the CNN models. The boxplot analysis reveals that MobileNetV2 achieved the lowest median inference time and exhibited the tightest IQR, indicating that it is the fastest and most stable model in terms of runtime performance. This result is consistent with the lightweight architecture of MobileNetV2, which is specifically designed for efficient computation on embedded systems.

GoogLeNet also demonstrates relatively low inference latency, although its inference time distribution shows slightly higher variation compared to MobileNetV2. InceptionV3, on the other hand, presents moderate inference times with a wider spread, indicating greater variability in processing time across different inference runs. ResNet-50, while achieving the highest classification accuracy in earlier performance evaluations, exhibits the highest median inference time and the largest spread among the four models. This indicates that ResNet-50 requires more computational resources due to its deeper network structure and larger number of convolutional layers. Despite this higher computational cost, the inference time of ResNet-50 remains within an acceptable range for real-time deployment in non-critical driver monitoring applications. Therefore, although MobileNetV2 provides the fastest inference speed, ResNet-50 offers a better balance between classification accuracy and runtime performance, making it suitable for implementation in the proposed distracted driving detection system on the NVIDIA Jetson Nano platform.

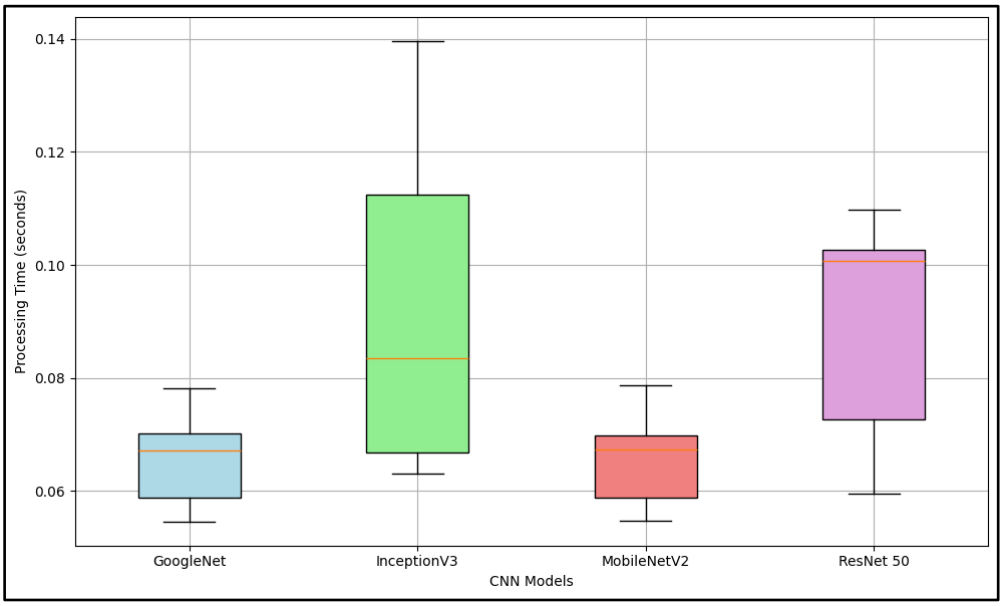


Figure 4.23 CNN Model Boxplot of Inference Times for GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50 on NVIDIA Jetson Nano

4.12 Lenovo IdeaPad Gaming 3 Inference Time Validation

To further evaluate inference performance under higher computational capacity, the trained CNN models were also executed on a Lenovo IdeaPad Gaming 3 laptop. This platform provides significantly greater processing power than the NVIDIA Jetson Nano, enabling a comparative evaluation between embedded system performance and high-performance laptop hardware. As with the Jetson Nano evaluation, each CNN model was tested over 90 inference runs, and the processing time per image was recorded. The inference time distribution for the four CNN models is illustrated using a boxplot in Figure 4.24. A boxplot summarizes the statistical distribution of a dataset using five key values: the minimum, first quartile (Q1), median, third quartile (Q3), and maximum. The box represents the interquartile range (IQR), which contains the middle 50% of the data, while the horizontal line inside the box indicates the median inference time.To compute the statistical values shown in the boxplot, the recorded inference time data from the 90 runs were first arranged in ascending order. Since the dataset consists of an even number of observations, the median is obtained by averaging the 45th and 46th values of the ordered inference time data, as shown in Equation (4.1). The interquartile range (IQR) represents the spread of the middle 50% of the data and is calculated as the difference between the third quartile (Q3) and the first quartile (Q1), as shown in Equation (4.2). Based on the computed inference time data, the median processing time was approximately 0.067 seconds, while the first quartile (Q1) and third quartile (Q3) values were approximately 0.059 seconds and 0.071 seconds, respectively. Therefore, the interquartile range (IQR) can be estimated as 0.012 seconds, representing the spread of the middle 50% of the inference time distribution.

The boxplot analysis in Figure 4.24 shows that MobileNetV2 achieved the lowest median inference time and the narrowest IQR, indicating that it remains the fastest and most consistent model in terms of runtime performance on the laptop platform. GoogLeNet also demonstrates very low inference latency with minimal variation between runs. InceptionV3 shows improved inference speed compared to the Jetson Nano implementation, although the spread of inference times remains slightly wider than that of MobileNetV2, indicating moderate variability during repeated inference runs. ResNet-50, despite having the deepest architecture and highest computational complexity, still produces relatively stable inference times on the laptop platform. However, it remains the slowest among the four models due to its deeper

network structure and larger number of convolutional layers. These results confirm that all evaluated CNN models are capable of performing real-time inference with very low latency when executed on a higher-performance computing platform, further supporting the feasibility of the proposed distracted driving detection system for practical deployment.

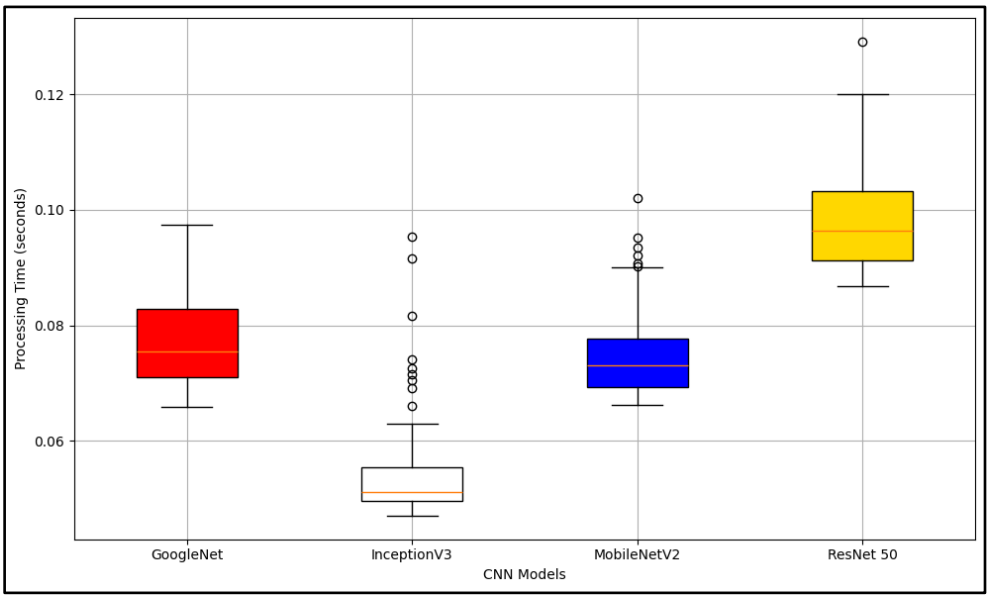
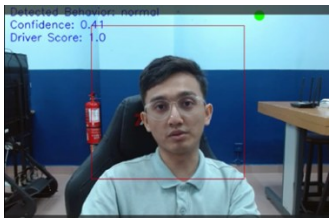


Figure 4.24 CNN Model Boxplot of Inference Times for GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50 on Lenovo IdeaPad Gaming 3

Table 4.11 presents the average inference time and corresponding frames per second (FPS) for the four CNN models, namely GoogLeNet, InceptionV3, MobileNetV2, and ResNet-50, evaluated on both the NVIDIA Jetson Nano and a laptop platform. The results indicate that lightweight architectures such as GoogLeNet and MobileNetV2 achieved the fastest inference speeds on the Jetson Nano, recording approximately 15 FPS, while heavier models such as InceptionV3 and ResNet-50 operated at around 11 FPS. On the laptop platform, InceptionV3 achieved the highest FPS at approximately 18.54 FPS, followed by MobileNetV2 and GoogLeNet at around 13 FPS, whereas ResNet-50 remained the slowest with approximately 10.17 FPS. The difference in inference speed between the models is mainly attributed to their architectural complexity, where deeper models with more convolutional layers require higher computational resources during inference. Despite these differences in processing speed, model selection in this study prioritizes classification accuracy rather than inference speed. ResNet-50 consistently achieved the highest recognition accuracy

across multiple datasets, particularly under challenging night driving conditions. Therefore, although it is not the fastest model in terms of FPS, ResNet-50 is considered the most reliable architecture for distracted driving behavior detection due to its superior classification capability and robust feature extraction performance. In addition to the inference time evaluation, the laptop platform was also used to demonstrate the real-time behavior detection capability of the proposed system. Figure 4.25 illustrates examples of real-time detection results obtained on the Lenovo IdeaPad Gaming 3 for six driving behaviors, namely (a) Normal Driving, (b) Nodding, (c) Yawning, (d) Talking, (e) Calling, and (f) Texting. Each frame shows the predicted behavior label together with the corresponding confidence score, demonstrating that the proposed CNN-based system is capable of performing real-time driver behavior recognition with minimal latency on a desktop-class computing platform. In addition, Figure 4.25(g) presents the No Driver condition, where no driver is detected inside the vehicle cabin.



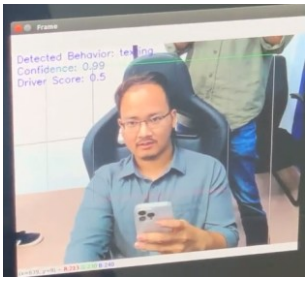
(a) Normal Driving

(b) Nodding

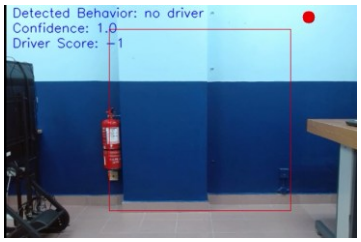
(c) Yawning

(d) Talking

(e) Calling



(f) Texting



(g) No Driver

Figure 4.25 Real-Time Detection Results on Lenovo Ideapad Gaming 3 Showing Classification Outputs for Six Driving Behaviors: (a) Normal Driving, (b) Nodding, (c) Yawning, (d) Talking, (e) Calling, (f) Texting, and (g) No Driver.

Table 4.11
Average Inference Time and Corresponding FPS for CNN Models on Jetson Nano and Laptop

CNN Models	**FPS**			
Jetson	GoogLeNet	InceptionV3	MobileNetV2	ResNet-50
Nano	15.47	11.27	15.39	11.01
Laptop	12.96	18.54	13.39	10.17

4.13 In-Vehicle Real-Time Prototype Validation

The system, powered by the NVIDIA Jetson Nano, was mounted securely on the vehicle's dashboard in a position that enabled continuous monitoring of the driver's facial region and upper body. This configuration replicated the intended real-world deployment scenario, ensuring that the detection module operated in natural driving conditions. The primary aim of this test was not to re-evaluate classification accuracy already established in previous controlled experiments, but to confirm the stability of inference speed and system responsiveness in a moving vehicle.

During the on-road trials, the prototype was tested in both urban and highway conditions under varying lighting environments, including day and night driving. The system ran continuously, processing input from the camera feed and classifying the driver's behavior in real time. As illustrated in Figure 4.26, the interface displayed the detected behavior, confidence score, and driver performance score alongside a live video feed. Concurrently, GPS coordinates and inference processing times per frame were logged to assess system responsiveness. The recorded inference speeds remained consistent with the results obtained in the controlled Jetson Nano test (Section 4.9), with processing times averaging between 147 ms and 156 ms per frame, confirming the prototype's ability to sustain real-time performance even under vibration, changing lighting, and dynamic backgrounds.

The successful completion of in-vehicle testing validates the system's feasibility for operational deployment in real driving environments. The findings confirm that the proposed CNN models, particularly MobileNetV2, due to their high efficiency and low latency, can be effectively integrated into embedded hardware for real-time distracted driving detection without significant degradation in performance. This in-field implementation completes the validation process for Objective 3, demonstrating that the system is not only functional in laboratory conditions but also robust in actual driving scenarios.

Figure 4.26 Prototype in Real-Time Operation Inside a Vehicle, Showing Live Behavior Detection with Confidence Score, Driver Score, GPS Logging, and Inference Time Display.

4.14 Summary

This chapter has comprehensively evaluated the performance of CNN models in detecting distracted driving behaviors under varying lighting conditions, particularly during night driving. The first objective, which aimed to implement deep learning models specifically for night driving scenarios, was achieved through the training and validation of four CNN architectures: InceptionV3, MobileNetV2, ResNet-50, and GoogLeNet using NIGHT-VIS and NIGHT-IR image datasets. The models demonstrated effective convergence within 10 epochs, with MobileNetV2 and ResNet-50 showing strong training stability. Data augmentation and CLAHE preprocessing further enhanced the models' learning capability in low-light environments.

The second objective focused on evaluating the model's performance under different lighting conditions, including day drive, night drive, and infrared settings. Results revealed that InceptionV3 and ResNet-50 consistently achieved high classification accuracy across all conditions, with ResNet-50 achieving the highest performance on NIGHT-VIS-CLAHE images at 90.29%. The CLAHE enhancement process played a significant role in improving accuracy, particularly for MobileNetV2, which demonstrated a substantial improvement from 35.06% to 65.41% when trained on NIGHT-VIS-CLAHE data. The comparison between NIGHT-IR and NIGHT-VIS-CLAHE further highlights the effectiveness of CLAHE preprocessing as a viable alternative to IR imaging, especially when IR cameras are unavailable. Class-wise

analysis using a confusion matrix showed that calling behavior was the most accurately classified, while nodding was the most frequently misclassified behavior.

The final objective aimed to validate the practical implementation of the trained models on an embedded platform, specifically the NVIDIA Jetson Nano. To this end, all four models were deployed and tested using 90 input samples each, and their processing times were analysed through boxplot visualizations. The results confirmed that MobileNetV2 had the shortest and most consistent inference time, making it highly suitable for real-time applications. While ResNet-50 had slightly longer processing times, its superior accuracy and robustness across all lighting conditions demonstrated that it can also be considered for real-time deployment when slightly higher latency is acceptable.

In conclusion, this study successfully met all three research objectives by implementing, evaluating, and validating deep learning models for distracted driving detection. The combination of CLAHE preprocessing, model selection, and embedded deployment ensures that the proposed system is both accurate and efficient for real-world, low light driving scenarios.

CHAPTER 5
CONCLUSION

5.1 Conclusion

This research proposed a deep learning-based system for detecting distracted driving behaviors during night driving using visible (VIS) and infrared (IR) images. The study addressed the challenge of recognizing driver behaviors under low-light conditions by integrating data augmentation, CLAHE-based contrast enhancement, and convolutional neural network (CNN) models. Four CNN architectures, namely GoogLeNet, MobileNetV2, InceptionV3, and ResNet-50, were evaluated across different image datasets including DAY-VIS, NIGHT-VIS, NIGHT-VIS-CLAHE, and NIGHT-IR. The results showed that image enhancement and infrared imaging significantly improved behavior recognition performance compared to raw night-time images. Among the evaluated models, ResNet-50 combined with CLAHE-enhanced NIGHT-VIS images achieved the highest accuracy of 90.29%, demonstrating its effectiveness for low-light distracted driving detection. Although MobileNetV2 achieved the fastest inference speed of 15.39 FPS on the NVIDIA Jetson Nano, ResNet-50 provided the best balance between accuracy and computational efficiency, achieving 11.01 FPS while maintaining superior recognition performance. The successful deployment of the models on the Jetson Nano confirms the feasibility of implementing the proposed system for real-time driver monitoring applications. Overall, the findings demonstrate that integrating CNN-based models with image enhancement techniques can significantly improve distracted driving behavior detection under challenging night driving conditions.

5.2 Future Work

Future research may focus on incorporating temporal information from video sequences to further improve the detection of distracted driving behaviors. Instead of relying solely on static images, advanced deep learning models such as Vision Transformers (ViT), 3D Convolutional Neural Networks (3D-CNN), or hybrid CNN-LSTM architectures could be explored to capture both spatial and temporal features of

driver behaviors. These approaches can model subtle motion patterns such as gradual head movements, blinking, or micro-expressions that may not be fully captured in single-frame analysis. In addition, future studies may consider integrating multimodal sensing technologies, including depth cameras, eye-tracking systems, or thermal imaging sensors, to improve robustness under challenging lighting conditions. Such developments would further enhance the reliability and practical deployment of intelligent driver monitoring systems for advanced driver assistance systems (ADAS) and road safety applications.

REFERENCES

(NHTSA), N. H. T. S. A. (2018). National Center for Statistics and Analysis: Distracted Driving 2016. *NHTSA's National Center for Statistics and Analysis*, *April 2018*, 1–6.

Abouelnaga, Y., Eraqi, H. M., & Moustafa, M. N. (2018). *Real-time Distracted Driver Posture Classification*. *Nips*. http://arxiv.org/abs/1706.09498

Affanni, A., & Najafi, T. A. (2022). Drivers' Attention Assessment by Blink Rate Measurement from EEG Signals. *2022 IEEE International Workshop on Metrology for Automotive, MetroAutomotive 2022 - Proceedings*, 128–132. https://doi.org/10.1109/MetroAutomotive54295.2022.9855098

Agarwal, A., Patni, K., & Rajeswari, D. (2021). Lung Cancer Detection and Classification Based on Alexnet CNN. *Proceedings of the 6th International Conference on Communication and Electronics Systems, ICCES 2021*, 1390–1397. https://doi.org/10.1109/ICCES51350.2021.9489033

Aiman Affandi, M., Hazwani Mokhtar, N., Zahir Hassan, M., Faradila Paiman, N., Mohd Jawi, Z., & Yoshifusa, M. (2026). Accelerating Insight: A Thorough Systematic Review on Driving Simulators and Their Transformative Effect on Driver Behaviour. *Journal of Advanced Research Design Journal Homepage*, *140*(1), 39–55. https://akademiabaru.com/submit/index.php/ard

Ali, L., Alnajjar, F., Jassmi, H. Al, Gochoo, M., Khan, W., & Serhani, M. A. (2021). Performance evaluation of deep CNN-based crack detection and localization techniques for concrete structures. *Sensors*, *21*(5), 1–22. https://doi.org/10.3390/s21051688

Ang, K., Lim, W. T., Loh, Y. P., & Ong, S. (2022). Noise-Aware Zero-Reference Low-light Image Enhancement for Object Detection. *2022 International Symposium on Intelligent Signal Processing and Communication Systems, ISPACS 2022*, 1–4. https://doi.org/10.1109/ISPACS57703.2022.10082804

Bagaskara, A., & Suryanegara, M. (2021). Evaluation of VGG-16 and VGG-19 Deep Learning Architecture for Classifying Dementia People. *Proceedings - 2021 4th International Conference on Computer and Informatics Engineering: IT-Based Digital Industrial Innovation for the Welfare of Society, IC2IE 2021*, 1–4. https://doi.org/10.1109/IC2IE53219.2021.9649132

Balakrishnan, R., Rawat, R., Kannan, S., Patel, W., Patil, H., & Rajkumar, S. (2024).

Image Recognition in Low-Light Conditions with Deep Learning Model. 1479–1485. https://doi.org/10.1109/IDCIoT59759.2024.10467387

Basulto-Lantsova, A., Padilla-Medina, J. A., Perez-Pinal, F. J., & Barranco-Gutierrez, A. I. (2020). Performance comparative of OpenCV Template Matching method on Jetson TX2 and Jetson Nano developer kits. *2020 10th Annual Computing and Communication Workshop and Conference, CCWC 2020*, 812–816. https://doi.org/10.1109/CCWC47524.2020.9031166

Batzolis, E., Vrochidou, E., & Papakostas, G. A. (2023). Machine Learning in Embedded Systems : Limitations , Solutions and Future Challenges. *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, 345–350. https://doi.org/10.1109/CCWC57344.2023.10099348

Bhaskar, A. (2018). EyeAwake: A cost effective drowsy driver alert and vehicle correction system. *Proceedings of 2017 International Conference on Innovations in Information, Embedded and Communication Systems, ICIIECS 2017*, *2018-Janua*, 1–6. https://doi.org/10.1109/ICIIECS.2017.8276114

Bhattacharya, S., & Bernadin, S. (2020). Correlation Studies on the Cognitive Responses of Distracted Drivers during Conversational Tasks. *Conference Proceedings - IEEE SOUTHEASTCON*, *2020-March*, 1–5. https://doi.org/10.1109/SoutheastCon44009.2020.9249678

Chen, H. Y., Li, Y. J., & Fan, Y. C. (2018). Luminance compensation design using modified histogram equalization for driving assistance system under night vision. *Proceedings - 2018 7th International Symposium on Next-Generation Electronics, ISNE 2018*, *Isne*, 1–2. https://doi.org/10.1109/ISNE.2018.8394660

Contrast, N. A. (2007). *Passenger Vehicle Occupant Fatalities by*. *May*.

Darapaneni, N., Parikh, B., Paduri, A. R., Kumar, S., Beedkar, T., Narayanan, A., Tripathi, N., & Khoche, T. (2022). Distracted Driver Monitoring System Using AI. *2022 International Conference on Interdisciplinary Research in Technology and Management, IRTM 2022 - Proceedings*, 1–8. https://doi.org/10.1109/IRTM54583.2022.9791545

Darvish Rouhani, B., Mirhoseini, A., & Koushanfar, F. (2017). TinyDL: Just-in-time deep learning solution for constrained embedded systems. *Proceedings - IEEE International Symposium on Circuits and Systems*. https://doi.org/10.1109/ISCAS.2017.8050343

Deep Learning - MATLAB & Simulink. (n.d.). Retrieved September 24, 2024, from

https://www.mathworks.com/discovery/deep-learning.html

Dehzangi, O., & Galster, S. (2018). Multi-modal system to detect on-the-road driver distraction. *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2191–2196.

Delivery services a booming industry. (n.d.). Retrieved September 21, 2024, from https://themalaysianreserve.com/2021/01/25/delivery-services-a-booming-industry/

Deshmukh, S. V., & Dehzangi, O. (2017). ECG-Based Driver Distraction Identification Using Wavelet Packet Transform and Discriminative Kernel-Based Features. *2017 IEEE International Conference on Smart Computing, SMARTCOMP 2017*. https://doi.org/10.1109/SMARTCOMP.2017.7947003

Distracted Driving Dangers and Statistics | NHTSA. (n.d.). Retrieved September 20, 2024, from https://www.nhtsa.gov/risky-driving/distracted-driving

Driving at Night - National Safety Council. (n.d.). Retrieved September 21, 2024, from https://www.nsc.org/road/safety-topics/driving-at-night

Ewecker, L., Schiffel, F., Schwager, R., Br, T., Sohn, T. S., & Villmann, T. (2024). *PVDN-URBAN – A DATASET FOR PROVIDENT VEHICLE DETECTION AT NIGHT IN URBAN SCENARIOS*. 34–40. https://doi.org/10.1109/ICIP51287.2024.10647873

Faraji, F., Lotfi, F., Khorramdel, J., Najafi, A., & Ghaffari, A. (2021). *Drowsiness Detection Based On Driver Temporal Behavior Using a New Developed Dataset*. 1–7. http://arxiv.org/abs/2104.00125

Feng, J. (2021). Fatigue driving detection system based on EEG Signal. *4th IEEE International Conference on Automation, Electronics and Electrical Engineering, AUTEEE 2021*, 299–302. https://doi.org/10.1109/AUTEEE52864.2021.9668667

Gadde, S., Lakshmanarao, A., & Satyanarayana, S. (2021). SMS Spam Detection using Machine Learning and Deep Learning Techniques. *2021 7th International Conference on Advanced Computing and Communication Systems, ICACCS 2021*, 358–362. https://doi.org/10.1109/ICACCS51430.2021.9441783

Gao, Y., Gao, L., & Li, X. (2021). A Generative Adversarial Network Based Deep Learning Method for Low-Quality Defect Image Reconstruction and Recognition. *IEEE Transactions on Industrial Informatics*, *17*(5), 3231–3240. https://doi.org/10.1109/TII.2020.3008703

Gargouri, A., & Haddeji, F. (n.d.). Implementation of a Photovoltaic Default Detection System using Deep learning. *2024 IEEE International Conference on Advanced Systems and Emergent Technologies (IC_ASET)*, 1–5.

Global status report on road safety 2023. (n.d.). Retrieved September 20, 2024, from https://www.who.int/teams/social-determinants-of-health/safety-and-mobility/global-status-report-on-road-safety-2023

Goel, L., Chennamaneni, S., Golla, A., Puchhakayala, S., & Rudraram, A. (2023). Transfer Learning-based Driver Distraction Detection. *2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, 58–63. https://doi.org/10.1109/ICSCDS56580.2023.10104662

Hsiao, S. H., & Jang, J. S. R. (2019). Improving ResNet-based feature extractor for face recognition via re-ranking and approximate nearest neighbor. *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2019*. https://doi.org/10.1109/AVSS.2019.8909884

Huang, C., Huang, C., Wang, X., Cao, J., Wang, S., Wang, S., Zhang, Y., & Zhang, Y. (2020). HCF: A Hybrid CNN Framework for Behavior Detection of Distracted Drivers. *IEEE Access*, *8*, 109335–109349. https://doi.org/10.1109/ACCESS.2020.3001159

Hwang, S., Lee, P., Park, S., & Byun, H. (2021). Learning Subject-independent Representation for EEG-based Drowsy Driving Detection. *9th IEEE International Winter Conference on Brain-Computer Interface, BCI 2021*, 3–5. https://doi.org/10.1109/BCI51272.2021.9385364

Jia, J., Zhang, C., & Xu, J. (2020). Centrality fluctuations and decorrelations in heavy-ion collisions in a Glauber model. *Physical Review Research*, *2*(2), 1–18. https://doi.org/10.1103/PhysRevResearch.2.023319

Jiang, S., Long, H., & Yang, T. (2021). A Real-time Embedded Target Tracking System Based on Deep Learning Model. *2021 7th International Conference on Computer and Communications (ICCC)*, 757–762. https://doi.org/10.1109/ICCC54389.2021.9674462

Jing, C., Cao, S., Shen, Y., & Wang, S. (2021). GoogLeNet-like Model for Pedestrian Attribute Detection in Surveillance Environment. *2021 7th International Conference on Computer and Communications, ICCC 2021*, 787–791. https://doi.org/10.1109/ICCC54389.2021.9674698

Ju, J., & Li, H. (2024). A Survey of EEG-Based Driver State and Behavior Detection

for Intelligent Vehicles. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, *6*(3), 420–434. https://doi.org/10.1109/TBIOM.2024.3400866

Kaur, A. (2024). An Efficient Fine-tuned GoogleNet Model for Multiclass Classification of Blood Cell Cancer. *2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, *2*, 1–5. https://doi.org/10.1109/IATMSI60426.2024.10502531

Kulkarni, U., Gurlahosur, S. V, Babar, P., Muttagi, S. I., Soumya, N., Jadekar, P. A., & Meena, S. M. (2023). Facial Key points Detection using MobileNetV2 Architecture. *2023 IEEE 8th International Conference for Convergence in Technology (I2CT)*, 1–6. https://doi.org/10.1109/I2CT57861.2023.10126381

Kumar, N., Chandarana, Y., Anand, K., & Singh, M. (2017). Using social media for word-of-mouth marketing. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10440 LNCS*, 391–406. https://doi.org/10.1007/978-3-319-64283-3_29

Kurian, A. T., & Kumar Soori, P. (2023). AI-Based Driver Drowsiness and Distraction Detection in Real-Time. *Proceedings of 3rd IEEE International Conference on Computational Intelligence and Knowledge Economy, ICCIKE 2023*, 13–18. https://doi.org/10.1109/ICCIKE58312.2023.10131730

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Li, G., Lee, B. L., & Chung, W. Y. (2015). Smartwatch-Based Wearable EEG System for Driver Drowsiness Detection. *IEEE Sensors Journal*, *15*(12), 7169–7180. https://doi.org/10.1109/JSEN.2015.2473679

Lionardi, S. J., Fransen, M., & Handojo, A. (2017). Night to day algorithm for video camera. *Proceedings - 2017 International Conference on Soft Computing, Intelligent System and Information Technology: Building Intelligence Through IOT and Big Data, ICSIIT 2017*, *2018-Janua*, 62–65. https://doi.org/10.1109/ICSIIT.2017.51

Liu, A., Tang, C., & Qiu, H. (2024). A Computer Vision System for Defect Contrast Enhancement on Transparent Materials with Structured Light and Neural Network. *2024 8th International Conference on Robotics, Control and Automation, ICRCA 2024*, 243–247.

https://doi.org/10.1109/ICRCA60878.2024.10649307

Malaysia has 32.3 million motor vehicles, 15.8 million drivers. (n.d.). Retrieved September 20, 2024, from https://www.nst.com.my/news/nation/2021/01/654712/malaysia-has-323-million-motor-vehicles-158-million-drivers

Ministry of Transport Malaysia. (n.d.). Retrieved September 20, 2024, from https://www.mot.gov.my/en/land/safety/malaysia-road-fatalities-index

NHTSA. (2025). *Distracted Driving in 2023*. *April*, 1–10. https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813703%0Ahttps://crashstats.nhtsa.dot.gov/#!/PublicationList/41

Nighttime Driving Accident Statistics in the U.S. | Injury Claim Coach. (n.d.). Retrieved September 21, 2024, from https://www.injuryclaimcoach.com/night-time-driving-accident-statistics.html

No Title. (n.d.). *38*(2), 1527–1536. https://doi.org/10.5281/zenodo.776772

Noh, Y., Kim, S., & Yoon, Y. (2019). Evaluation on Diversity of Drivers' Cognitive Stress Response using EEG and ECG Signals during Real-Traffic Experiment with an Electric Vehicle ∗. *2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019*, 3987–3992. https://doi.org/10.1109/ITSC.2019.8916844

NVIDIA. (2019). *Jetson Nano Developer Kit User Guide*. 24. https://developer.download.nvidia.com/embedded/L4T/r32-3-1_Release_v1.0/Jetson_Nano_Developer_Kit_User_Guide.pdf

Obreja, M., & Dobrea, D. (2024). Modified ResNet-50 for Training the Neural Network in Pothole Detection using Deep Learning in Matlab. *2024 16th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 1–4. https://doi.org/10.1109/ECAI61503.2024.10607466

Oldenziel, E., Ohnemus, L., & Saralajew, S. (2020). *Provident Detection of Vehicles at Night*. *Iv*. https://doi.org/10.1109/IV47402.2020.9304752

Omerustaoglu, F., Sakar, C. O., & Kar, G. (2020). Distracted driver detection by combining in-vehicle and image data using deep learning. *Applied Soft Computing*, *96*, 106657. https://doi.org/10.1016/J.ASOC.2020.106657

P, D. P. (2022). Multifunctional Night Patrolling Robot Based on Rocker-Bogie Mechanism. *2022 IEEE Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, 1–5. https://doi.org/10.1109/IATMSI56455.2022.10119262

Pandit, A., & Parihar, A. S. (2022). Detail Decomposition for Low Light Image Enhancement. *Proceedings - 2022 4th International Conference on Advances in Computing, Communication Control and Networking, ICAC3N 2022*, 2320–2326. https://doi.org/10.1109/ICAC3N56670.2022.10074576

Perera, D., Wang, Y. K., Lin, C. T., Zheng, J., Nguyen, H. T., & Chai, R. (2020). Statistical Analysis of Brain Connectivity Estimators during Distracted Driving. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, *2020-July*, 3208–3211. https://doi.org/10.1109/EMBC44109.2020.9176240

Pinto, A., Bhasi, M., Bhalekar, D., Hegde, P., & Koolagudi, S. G. (2019). A deep learning approach to detect drowsy drivers in real time. *2019 IEEE 16th India Council International Conference, INDICON 2019 - Symposium Proceedings*, 31–34. https://doi.org/10.1109/INDICON47234.2019.9030305

Prakash, J., Nashita, V., S, S. S., V, S. S., Sivesh, K., & Ashwin, V. (2024). Key-Point Driven Animation of Stick Figures and Children ' s Drawings using ResNet -50 and Shape Manipulation. *2024 International Conference on Smart Systems for Electrical, Electronics, Communication and Computer Engineering (ICSSEECC)*, *Icsseecc*, 369–374. https://doi.org/10.1109/ICSSEECC61126.2024.10649440

Quach, L. (2024). *Using SURF to Improve ResNet-50 Model for Poultry Disease Recognition Algorithm*. *October 2020*, 8–9. https://doi.org/10.1109/ICCI51257.2020.9247698

Raj, P. (2020). *An Embedded Deep Learning Based Traffic Advisory System*. *2020*. https://doi.org/10.1109/ICRAIE51050.2020.9358364

Rajput, P., Nag, S., & Mittal, S. (2020). Detecting Usage of Mobile Phones using Deep Learning Technique. *ACM International Conference Proceeding Series*, 96–101. https://doi.org/10.1145/3411170.3411275

Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. http://arxiv.org/abs/1804.02767

Sajid, F., Javed, A. R., Basharat, A., Kryvinska, N., Afzal, A., & Rizwan, M. (2021). An Efficient Deep Learning Framework for Distracted Driver Detection. *IEEE Access*, *9*, 169270–169280. https://doi.org/10.1109/ACCESS.2021.3138137

Salim, N. R., Jayaraman, U., & Srinath, V. (2020). Face recognition in the dark: A unified approach for NIR- VIS and VIS- NIR face matching. *4th IEEE*

Conference on Information and Communication Technology, CICT 2020. https://doi.org/10.1109/CICT51604.2020.9312106

Saralajew, S., Ohnemus, L., Ewecker, L., Asan, E., Isele, S., & Roos, S. (2021). A Dataset for Provident Vehicle Detection at Night. *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 9750–9757. https://doi.org/10.1109/IROS51168.2021.9636162

Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Computer Science*, *2*(6), 1–20. https://doi.org/10.1007/s42979-021-00815-1

Shen, Z., Zhang, R., Dell, M., Lee, B. C. G., Carlson, J., & Li, W. (2021). LayoutParser: A Unified Toolkit for Deep Learning Based Document Image Analysis. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12821 LNCS*(Dl), 131–146. https://doi.org/10.1007/978-3-030-86549-8_9

Streiffer, C., Raghavendra, R., Benson, T., & Srivatsa, M. (2017). DarNet: A deep learning solution for distracted driving detection. *Middleware 2017 - Proceedings of the 2017 International Middleware Conference (Industrial Track)*, 22–28. https://doi.org/10.1145/3154448.3154452

Suzen, A. A., Duman, B., & Sen, B. (2020). Benchmark Analysis of Jetson TX2, Jetson Nano and Raspberry PI using Deep-CNN. *HORA 2020 - 2nd International Congress on Human-Computer Interaction, Optimization and Robotic Applications, Proceedings*, 3–7. https://doi.org/10.1109/HORA49412.2020.9152915

Systems, D. (2022). *A Review of Recent Developments in Driver Drowsiness*. 1–41.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 2818–2826. https://doi.org/10.1109/CVPR.2016.308

Tarabin, M., Alketbi, M. M., Alfalasi, H. R., Alsmirat, M., & Sharrab, Y. (2023). Detecting Distracted Drivers Using Convolutional Neural Networks. *2023 International Conference on Intelligent Data Science Technologies and Applications, IDSTA 2023*, 59–66. https://doi.org/10.1109/IDSTA58916.2023.10317853

Thalanki, V. (2023). Voice - based Image Captioning using Inception - V3 Transfer

Learning Model. *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)*, *Icoei*, 51–57. https://doi.org/10.1109/ICOEI56765.2023.10125754

Tirumalasetty, M. H. (2025). *Driver Drowsiness Detection Using Convolutional Neural Networks (CNN)*. *7*(5), 1–6.

Tsai, T. (2019). A Sketch Classifier Technique with Deep Learning Models Realized in an Embedded System. *2019 IEEE 22nd International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS)*, 1–4.

Verma, S., & Bhoomi, D. (2023). Detection of Traffic Sign using Inception V3 in Comparison with VGG-19 to Measure Accuracy. *2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 730–733. https://doi.org/10.1109/ICACITE57410.2023.10183243

Wang, F., Zheng, R., Li, P., Song, H., Du, D., & Sun, J. (2021). Face recognition on Raspberry Pi based on MobileNetV2. *Proceedings - 2021 International Symposium on Artificial Intelligence and Its Application on Media, ISAIAM 2021*, 116–120. https://doi.org/10.1109/ISAIAM53259.2021.00031

Wang, H., & Wang, L. (2020). FACIAL FEATURE EMBEDDED CYCLEGAN FOR VIS-NIR TRANSLATION Haijian Zhang Lei Yu School of Electronic Information , Wuhan University , China Agency for Science , Technology and Research , Singapore. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1903–1907.

Wang, Y. F., Liu, H. M., & Fu, Z. W. (2019). Low-Light Image Enhancement via the Absorption Light Scattering Model. *IEEE Transactions on Image Processing : A Publication of the IEEE Signal Processing Society*, *28*(11), 5679–5690. https://doi.org/10.1109/TIP.2019.2922106

Wareechol, T., & Chiracharit, W. (2021). Recognition of Similar Gait Pattern Using Transfer Learning DarkNet. *Proceeding of the 2021 9th International Electrical Engineering Congress, IEECON 2021*, 381–384. https://doi.org/10.1109/iEECON51072.2021.9440386

Wu, W., Fan, B., He, Y., Li, W., Li, W., Liu, Z., Wan, W., & Xie, Y. (2022). Design of Autonomous Driving Verification Platform Based on Udacity Vehicle Simulator. *2022 IEEE Smartworld, Ubiquitous Intelligence & Computing, Scalable Computing & Communications, Digital Twin, Privacy Computing, Metaverse, Autonomous & Trusted Vehicles*

(SmartWorld/UIC/ScalCom/DigitalTwin/PriComp/Meta), 732–738. https://doi.org/10.1109/SmartWorld-UIC-ATC-ScalCom-DigitalTwin-PriComp-Metaverse56740.2022.00114

Wu, X., Yu, W., & Cang, N. (2020). *Detection and Research on Unsafe Driving of Taxi*. 176–182. https://doi.org/10.1109/TOCS50858.2020.9339721

Xing, Y., Lv, C., Wang, H., Cao, D., Velenis, E., & Wang, F. (2019). Driver Activity Recognition for Intelligent Vehicles : A Deep Learning Approach. *IEEE Transactions on Vehicular Technology*, *68*(6), 5379–5390. https://doi.org/10.1109/TVT.2019.2908425

Xiong, Q., Lin, J., Yue, W., Liu, S., Liu, Y., & DIng, C. (2019). A deep learning approach to driver distraction detection of using mobile phone. *2019 IEEE Vehicle Power and Propulsion Conference, VPPC 2019 - Proceedings*, 2–6. https://doi.org/10.1109/VPPC46532.2019.8952474

Yang, C., Lin, Y., & Peng, S. (2017). Develop a video monitoring system for dairy estrus detection at night. *2017 International Conference on Applied System Innovation (ICASI)*, 1900–1903. https://doi.org/10.1109/ICASI.2017.7988321

Ye, L., & Ma, Z. (2023). LLOD:A Object Detection Method under Low-Light Condition by Feature Enhancement and Fusion. *2023 4th International Seminar on Artificial Intelligence, Networking and Information Technology, AINIT 2023*, 659–662. https://doi.org/10.1109/AINIT59027.2023.10212748

Zhang, C., Wu, X., Zheng, X., & Yu, S. (2019). Driver drowsiness detection using multi-channel second order blind identifications. *IEEE Access*, *7*, 11829–11843. https://doi.org/10.1109/ACCESS.2019.2891971

Zhao, Y., Xie, K., Zou, Z., & He, J. B. (2020). Intelligent Recognition of Fatigue and Sleepiness Based on InceptionV3-LSTM via Multi-Feature Fusion. *IEEE Access*, *8*, 144205–144217. https://doi.org/10.1109/ACCESS.2020.3014508

Zhonglin, H., Lei, N., Ni, C., Xinpeng, C., Fangde, L., Guowei, W., Libin, X., & Kaiyu, T. (2023). IMPROVED GOOGLENET MODEL FOR TIRED DRIVING DETECTION. *2023 20th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, 1–7. https://doi.org/10.1109/ICCWAMTIP60502.2023.10387133

AUTHOR'S PROFILE

Muhammad Firdaus Ishak is an aspiring graduate student currently pursuing a Master of Science in Electrical Engineering (Research) at Universiti Teknologi MARA, Malaysia. Dedicated to the field of Vehicle Intelligence and Telematics, he actively contributes as a member of the esteemed VITAL team at the College of Engineering, Universiti Teknologi MARA. Focused on the implementation of deep learning networks, his research primarily centers around detecting distracted driving behaviors in night driving scenarios, emphasizing the critical domain of image processing. With a solid academic background, including a Bachelor of Electrical Engineering (Hons) in Electronics Engineering and a Diploma of Electrical Engineering (Hons) in Electronics Engineering, he recently showcased his expertise as a presenter at the 11th International Conference on Information and Communication Technology.

LIST OF PUBLICATION

Bin Ishak, M. F., Zaman, F. H. K., Mun, N. K., & Makhtar, A. K. (2023). Day Driving and Night Driving Behavior Detection Using Deep Learning Models. *2023 11th International Conference on Information and Communication Technology, ICoICT 2023*, *2023-Augus*, 463–468. https://doi.org/10.1109/ICoICT58202.2023.10262745

Ishak, M. F., Zaman, F. H. K., Mun, N. K., Abdullah, S. A. C., & Makhtar, A. K. (2024). Improving night driving behavior recognition with ResNet50. *Indonesian Journal of Electrical Engineering and Computer Science*, *33*(3), 1974–1988. https://doi.org/10.11591/ijeecs.v33.i3.pp1974-1988