CRANFIELD UNIVERSITY

LICHAO YANG

DRIVER BEHAVIOUR CHARACTERIZATION USING ARTIFICIAL INTELLIGENCE TECHNIQUES IN LEVEL 3 AUTOMATED VEHICLE

SCHOOL OF AEROSPACE TRANSPORT AND MANUFACTURING

PhD
Academic Year: 2018 - 2021

Supervisor: Dr. Yifan Zhao
Associate Supervisor: Prof. James Brighton
September 2021

CRANFIELD UNIVERSITY

SCHOOL OF AEROSPACE TRANSPORT AND MANUFACTURING

PhD

Academic Year 2018 - 2021

LICHAO YANG

Driver Behaviour Characterization using Artificial Intelligence Techniques in Level 3 Automated Vehicle

Supervisor: Dr. Yifan Zhao
Associate Supervisor: Prof. James Brighton
September 2021

This thesis is submitted in partial fulfilment of the requirements for the degree of PhD
(NB. This section can be removed if the award of the degree is based solely on examination of the thesis)

ABSTRACT

Autonomous vehicles free drivers from driving and allow them to engage in some non-driving related activities. However, the engagement in such activities could reduce their awareness of the driving environment, which could bring a potential risk for the takeover process in the current automation level of the intelligent vehicle. Therefore, it is of great importance to monitor the driver's behaviour when the vehicle is in automated driving mode.

This research aims to develop a computer vision-based driver monitoring system for autonomous vehicles, which characterises driver behaviour inside the vehicle cabin by their visual attention and hand movement and proves the feasibility of using such features to identify the driver's non-driving related activities. This research further proposes a system, which employs both information to identify driving related activities and non-driving related activities. A novel deep learning-based model has been developed for the classification of such activities. A lightweight model has also been developed for the edge computing device, which compromises the recognition accuracy but is more suitable for further in-vehicle applications. The developed models outperform the state-of-the-art methods in terms of classification accuracy. This research also investigates the impact of the engagement in non-driving related activities on the takeover process and proposes a category method to group the activities to improve the extendibility of the driving monitoring system for unevaluated activities. The finding of this research is important for the design of the takeover strategy to improve driving safety during the control transition in Level 3 automated vehicles.

Keywords:

Action recognition, Non-driving related task, Deep learning, Level 3 automation

ACKNOWLEDGEMENTS

I would like to express my deep sense of gratitude to my primary supervisor, Dr. Yifan Zhao, for his tireless and continuous support, patience and motivation during my whole research. My thanks also go to my associate supervisor, Prof. James Brighton, for his guidance and help.

I would also like thanks to all the members of our research group, Dr. Weixiang Du, Dr. HaoChen Liu, Dr. Xiaocai Shan, Youdao, Jun, Kailun, Shuozhi, Aoxiang, and my friend Feiyang. It is their kind support that makes my life in UK wonderful, especially, during the COVID-19 pandemic.

My gratitude also goes to Arjun Thirunavukarasu, Gavin Allen, Euan Williams, Joe Magrath and Daniel Mateos for their support from the industrial perspective. I also thank my review teams in Cranfield University, Dr. Jane Hodgkinson and Dr. Gilbert Tang, for their guidance and constructive feedback during my research.

Last but not the least, I would like to thank all my family and my friends who have helped and encouraged me to go through my research journey. I am so grateful to have you all in my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

AAP	Adaptive average pooling
ADAS	Advanced driver assistance system
AI	Artificial Intelligence
AV	Autonomous vehicle
CLNF	Conditional Local Neural Fields
CNN	Convolutional neural network
COCO	Common objects in context
CPU	Central processing unit
DDT	Dynamic driving task
DMV	Department of Motor Vehicles
DRA	Driving related activity
EEG	Electroencephalogram
ERR	Error Reduction Ratio
FC	Fully connected
FFNN	Feedforward neural network
FLOPS	Floating point operations per second
FPN	Feature pyramid network
FPS	Frames per second
GPU	Graphics processing unit
GRP	Gaze related parameter
HAD	Highly automated driving
HCI	Human-computer interaction
HMI	Human-Machine Interaction
HRP	Head pose related parameter
LSTM	Long short-term memory
NDRA	Non-driving related activity
NFIR	Nonlinear finite impulse response
NHTSA	National Highway Traffic Safety Administration
OEDR	Object and event detection and response
OLS	Orthogonal least squares
PDM	Point Distribution Model
RCNN	Regional convolutional neural network

ResNet	Residual Network
RMSE	Root Mean Square Error
RNN	Recurrent neural network
ROI	Region of interest
SAE	Society of Automotive Engineers
SGD	Stochastic gradient descent
SSD	Single-shot detector
TOR	Take-over request
VNRX	Volterra Non-linear Regressive with eXogenous
YOLO	You only look once

1 Introduction

Autonomous vehicles (AVs) as known as self-driving vehicles refer to vehicles that could sense the environment, make decisions, and control themselves to drive safely with limited or no human driver assistance [1]. In recent years, considerable progress in the AVs has been made with the significant advances in sensing devices, artificial intelligence algorithms and the internet of things in autonomous driving. Despite lots of efforts have been made and resources have been invested by the automotive industry, fully autonomous driving is not realistic yet. The commercialised technology on the current market focuses on the advanced driver assistance systems (ADASs) that cover the features such as braking control and lane assist. Based on the capability of the AVs, the automation level has been defined by the Society of Automotive Engineers (SAE) [2].

1.1 Automation level

The driving automation has been classified into 6 levels from fully manual driving to fully automated driving defined by SAE, as presented in Figure 1-1.

- Level 0 No driving automation: the vehicle is purely controlled by the human driver. Most of the current vehicles on the road are at this level.

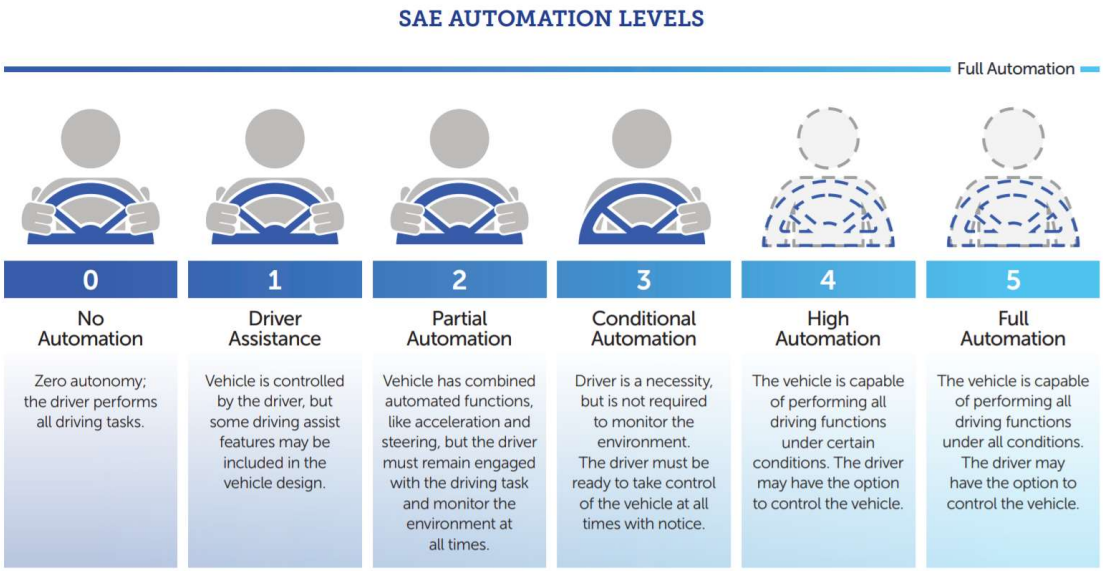


Figure 1-1 Automation levels defined by SAE [3]

- Level 1 Driver assistance: for certain driving scenarios, either the lateral or the longitudinal motion can be controlled by the vehicle.
- Level 2 Partial driving automation: for certain driving scenarios, both the lateral and the longitudinal motion of the dynamic driving task (DDT) can be controlled by the vehicle. The driver is expected to keep monitoring the driving environment and execute the appropriate response.
- Level 3 Conditional driving automation: for certain driving scenarios, the vehicle could achieve the entire DDT and monitor the driving environment with the expectation that the driver executes the appropriate response for the requests to intervene triggered by system failure.
- Level 4 High driving automation: for certain driving scenarios, the vehicle performs the entire DDT without the expectation of the driver intervention in most circumstances.
- Level 5 Full driving automation: The AV performs the entire DDT unconditionally. The vehicle requires no human attention and intervention.

Based on this definition, most of the current ADASs implemented vehicles are qualified as Level 2. In 2021, Both Daimler's Mercedes-Benz and BMW claimed that Level 3 will be deployed in the coming S-class and 7 series. Meanwhile, the development and the test of the Level 4 automation mainly focus on some specific use cases such as Waymo Level 4 autonomous truck which is designed for freight hauling on the highway, and Toyota e-Palette autonomous shuttle which aims to operate in the athletes' villages of the Tokyo 2020 Olympic and Paralympic Games.

1.2 Traffic safety

One of the most important aims of developing AVs is to reduce traffic accidents on the road. From the report produced by the National Highway Traffic Safety Administration (NHTSA), 94% of fatal traffic accidents are caused by human error [3]. Fully automated driving is considered to have the capability of eliminating human error in traffic. However, robust automated driving has not been achieved yet. The immature design of the current road-testing AVs still causes the accident. Recorded by the California Department of Motor Vehicles (DMV), on California

public roads, from 2014 to January 2020, there are 168 reported AVs collision cases, in which the vehicles were in automatic driving mode [4]. In Japan, the operation of the Level 4 autonomous shuttles has been suspended due to an accident that injured a visually impaired athlete. Moreover, the accident caused by commercial AVs even cost lives. In 2016, a Tesla Model S with the Autopilot (an ADASs) engaged collided with and travelled under a trailer at 74 mph and then stopped after colliding with two chain-link fences and a utility pole [5]. The driver was killed, even there was sufficient sight distance to afford time for the driver to act to prevent the crash. In 2018, a Tesla Model X with the Autopilot engaged crashed the attenuator of the road at a speed of 70.8 mph, then struck by 2 following vehicles and caught the fire [6]. The investigation presents that the Autopilot system has not detected the vehicle is in an improper position on the road and did not provide any type of warning. The vehicle data showed that before the crash the driver's hands have not been detected on the steering wheel for 6 seconds. The phone usage data also suggests that the driver was playing a game before the crash. It should be noted that in both fatalities, the driver's distraction is considered as the main factor that caused the accidents. The drivers were engaging in some non-driving related activities (NDRAs) such as *watching movies* and *playing games*, which made them has limited awareness of the hazard ahead and they have not taken any emergency braking or evasive steering to avoid the collision.

1.3 Motivation and research gaps

In conditional driving automation, the driver does not need to take the object and event detection and response (OEDR) task [2]. The system is required to monitor the driving environment, detect and recognise the objects and events and take the proper action against such objects and events. However, the driver is expected to respond to the request to intervene issued by the vehicle. It means, in some driving scenarios, the driver could take his/her eyes off the road and engage in some NDRAs such as *playing games, reading and sending email* but they need to prepare to takeover if the vehicle fails to complete the DDT. The engagement of the NDRAs could reduce the driver's monitoring capability of the

surrounding environment and the attention to the driving task, which could carry high risks to other road traffic participants if the driver is required to control the vehicle for an emergency event. In the Tesla fatalities, the overtrust and overreliance of the drivers on the vehicle automation system make them prolonged disengage in the DDT and immerse in some NDRAs. When the crash happened, they were neglectful of the forward environment and did not take any action. In the current and following automation level (level 2 and level 3), the driver is responsible for monitoring the driving environment and the automation system, and takeovering the vehicle if the intervention is requested. Therefore, the driver's behaviour, especially NDRAs, need to be monitored to avoid the prolonged disengagement of DDT. The impact of the NDRAs engagement on the driver's takeover also needs to be evaluated for the proper design of the takeover modality to secure a smooth and safe control transition.

The research gaps are listed as below:

- The existing driver monitoring systems mainly focus on the NDRAs recognition and most of the methods extract the features from the driver's body gesture, which lacks the estimation of the driver's visual attention. There is limited literature on the recognition of driving-related activities (DRAs), which is important to evaluate the driver's situation awareness before the takeover.
- The activities investigated in the existing NDRAs recognition research and the public driver distraction dataset are normally easy to be differentiated in the spatial domain, such as *eating, calling, watching movies*, etc. There is a lack of research on the classification of the high-similarity activities, *typing on the phone* and *playing games,* which could have different levels of mental demands that affect the driver's takeover quality. The employed methods are mainly image-based. Very limited research focus on the driver's behaviour characterisation in the spatio-temporal domain.
- Most of the existing deep learning-based methods are developed on the high-performance workstation. There is very limited research on the development of lightweight systems in this field, which is important for in-

vehicle applications. There is a lack of inference latency evaluation on the edge computing device.

- The research on the impact evaluation of the NDRAs engagement on the takeover process is limited and specific on individual activity, which limits the extendibility of the driver monitoring system for the unevaluated NDRAs.

1.4 Aims and Objectives

1.4.1 Aim

This thesis aims to develop an Artificial Intelligence (AI) enabled solution to characterise the driver's behaviour of NDRAs engagement and understand its impact on the driver's take-over performance in the level 3 automated driving vehicle. Specifically, the Computer Vision, Machine Learning and Deep Learning approaches will be used to characterise the driver-object interaction behaviour.

1.4.2 Objectives

The aim will be achieved by meeting the following 6 objectives.

1.4.2.1 Literature review

This thesis investigated the NDRAs that the driver could engage in the automated driving vehicle and the state-of-the-art computer vision-based methods that are used for activity identification or recognition. It also reviewed the experiment setup for evaluation of the NDRAs' implication in the take-over process. The style of this thesis is paper-format, and the literature review is delivered in each individual paper chapter.

1.4.2.2 Visual attention related NDRAs recognition

The human eye focus reflects their visual attention, localisation of which in the vehicle cabin could help to identify whether the driver is checking the driving environment or engaging with some visual related NDRAs, such as phone using, tablet using, centre console interacting. This objective links to Chapter 1.

1.4.2.3 Hand gesture based NDRAs recognition

The located visual attention is not sufficient to classify the specific NDRAs with the same object, for instance, *playing games* or *watching movies with a phone*. Analysing the pattern of the interaction between the driver and the object/device, specifically, the hand gesture could improve the recognition performance. This objective links to Chapter 3.

1.4.2.4 Fusion of visual attention and hand movement for NDRAs/DRAs recognition

The driver's gaze focus and hand behaviour could be combined for better detection of the NDRAs engagement and refined activity classification. The fusion of these two feeds could achieve the driver's behaviour recognition inside the vehicle cabin. This objective links to Chapter 4.

1.4.2.5 Development of the efficient NDRAs recognition system for edge computing

Fast and continuous recognition of the driver's NDRAs engagement could help evaluate the driver's situation awareness and mental state before the take-over process, which requests the system is efficient and able to be executed in the edge computing device. This objective links to Chapter 5.

1.4.2.6 Investigation of the NDRAs' implication on the take-over quality

The investigation of the impact of specific NDRAs on the take-over process could support determining the modality of the take-over request and design of the Human-Machine Interaction (HMI) for a safe and smooth control transition. This objective links to Chapter 6.

1.5 Experiment summary

In this research, a series of experiments have been done for evaluating the recognition of different types of NDRAs and the takeover performance. A brief introduction has been given in this section from the perspectives of experiment design, experiment platform, and produced NDRAs dataset. The detail will be illustrated in the following chapters.

1.5.1 Experiment design

Several experiments have been done for the objectives of visual-related NDRAs recognition, hand gesture-based NDRAs recognition, driver behaviour monitoring (fusion of visual attention and hand movement), and takeover performance evaluation.

1.5.1.1 Experiments for visual-related NDRAs recognition

Two experiments were conducted separately:

1. Visual attention estimation experiment: this experiment is a feasibility study for mapping the driver's visual attention. Driver's facial information and the view that mimics the driver's view has been captured by a dual-camera system.
2. NDRAs engagement experiment: this experiment records the driver's behaviour mainly the head movement during the visual-related NDRAs engagement.

1.5.1.2 Experiment for hand gesture-based NDRAs recognition

3. Hand gesture based NDRAs experiment: this experiment aims to explore the participant's hand movement pattern during the NDRAs engagement.

1.5.1.3 Experiment for driver behaviour monitoring during automated driving

4. Driver behaviour monitoring experiment: this experiment employed 2 cameras to capture the participants head and hand movement while the vehicle is in automated driving. The participants were required to engage in some NDRAs when the vehicle is driving automatically, and they were allowed to check the road during the engagement.

1.5.1.4 Experiment for takeover performance evaluation

5. Takeover performance evaluation experiment: this experiment aims to evaluate the impact of the NDRAs engagement on the takeover process during automated driving.

1.5.2 Experiment platform

1.5.2.1 Data collection platform

1. The Land Rover Discovery 4: this vehicle stayed stationary during the experiment.
2. The Land Rover Discovery 5: this vehicle was modified to accommodate both autonomous and human driving. To ensure safety, a steering wheel and a set of pedals were added in the back seat, which allows a safety driver to intervene and override the autonomous system.

1.5.2.2 Data process platform

1. Workstation: A PC with an Intel i7 9700k CPU, 32GB memory and an NVIDIA GeForce RTX 2080 GPU.
2. Workstation: A PC with an Intel i9 10900k CPU, 64GB memory and an NVIDIA Quadro RTX 8000 GPU.
3. Edge computing devices: Three Jetson modules, including Jetson Nano, Jetson TX2 and Jetson AGX Xavier, were evaluated.

1.5.3 NDRAs dataset

There are 3 NDRAs datasets produced in this research. These evaluated activities were selected by considering the outcomes from surveys [7], [8].

1. Visual related NDRAs dataset

   This dataset focuses on the activities that require visual attention engagement. There are 5 different NDRAs in this dataset which are *reading books, watching movies* with a cell phone, *sending an E-mail* by using a laptop, *playing games* with a tablet and *interacting with the centre console* to select a radio channel. Each NDRAs required the participant's eye gaze interaction with different objects like books, phones, etc. There are 6 participants in this dataset. The video data were captured by two cameras, which cover both the participant's facial information and the view inside the vehicle cabin.

2. Hand gesture related NDRAs dataset

This dataset focuses on the activities that require the participant's hand engagement with the object. There are 10 NDRAs in this dataset, which includes 5 activities *browsing websites, sending emails, playing games, reading,* and *watching videos* with 2 objects phone and tablet. A total of 10 participants were recruited for this experiment. The video data was captured by a camera, which was mounted on the roof of the vehicle between two front seats and face to the driver.

3. Driver behaviour dataset

This dataset includes not only NDRAs but also the DRAs. It captures the driver's head and hand movement with 2 cameras. There are 6 activities involved in this dataset, which are 4 types of NDRAs and 2 types of DRAs. The evaluated NDRAs are *reading news, watching videos, playing games* and *answering questionnaires* using a tablet. The DRAs are *road checking* and *driving*. 14 participants were recruited for this experiment. The videos were recorded in different weather and lighting conditions including sunny, cloudy, rainy and snowy.

1.6 Thesis structure

In this thesis, computer vision-based methods are researched and employed to monitor the driver's behaviour due to their property of low cost, non-intrusive and well-accepted generalisation, which is promising for further application in AVs. The activities that the driver could engage in inside the vehicle cabin can be categorised as DRAs and NDRAs. DRAs refer to the driver's driving activity and environment sensing activity. NDRAs studied in existing researches and surveys normally include *calling*, *reading*, *playing games*, *watching movies*, *eating* and *sleeping*, etc [8], [9]. When the vehicle is driving automatically, the driver is constrained in the seat, the pattern of the DRAs/NDRAs engagement can be characterised as the interaction between the driver and the semantic objects,

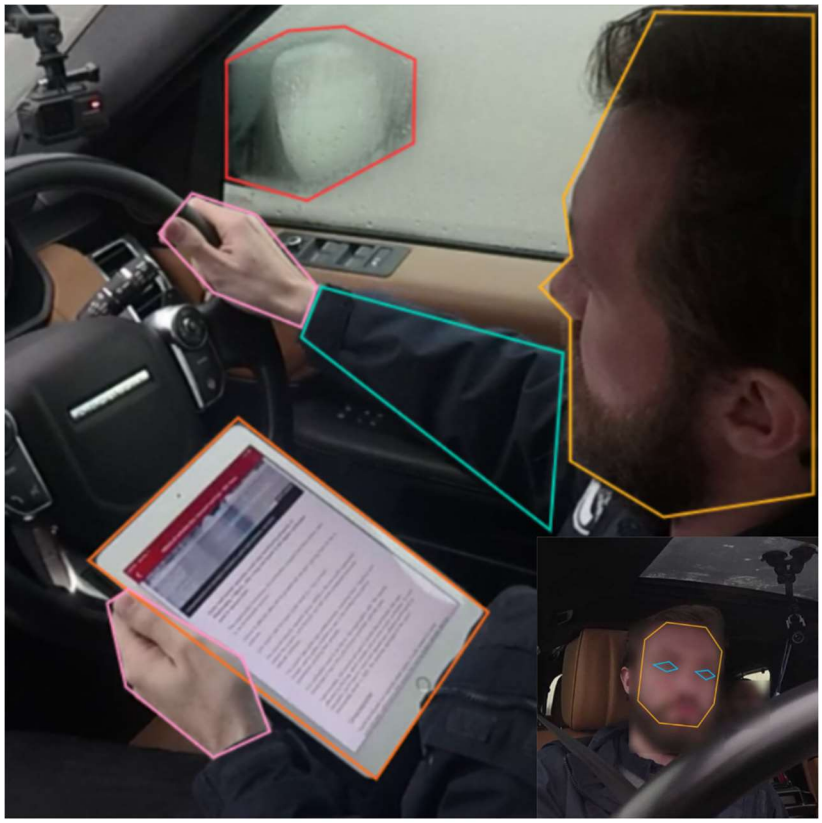


Figure 1-2 Valuable features for driver behaviour monitoring

such as the tablet, phone, and wing mirrors. Figure 1-2 presented the features that could be used for activity recognition. For the DRAs, the driver always checks the driving environment. The driver's head and gaze movement contain the attention information, which indicates the engagement of the DRAs and reflects its awareness of the driving environment [10]. Moreover, the recognition of the driving activity requests the interaction between the driver's hand/arm movement and the steering wheel. The NDRAs recognition could be more complicated and challenging since the driver's behaviour could be similar during the activity engagement. The driver's hand movement pattern when he/she is interacting with some objects needs to be studied. Therefore, the investigation of the driver's attention, hand movement and semantic objects in the vehicle is necessary for monitoring the driver's behaviour.

The structure of the thesis is presented in Figure 1-3. The main part of the thesis is contributed by 4 published journal papers, 1 journal paper under review and 1 published conference paper. Driver's behaviour inside the vehicle cabin is

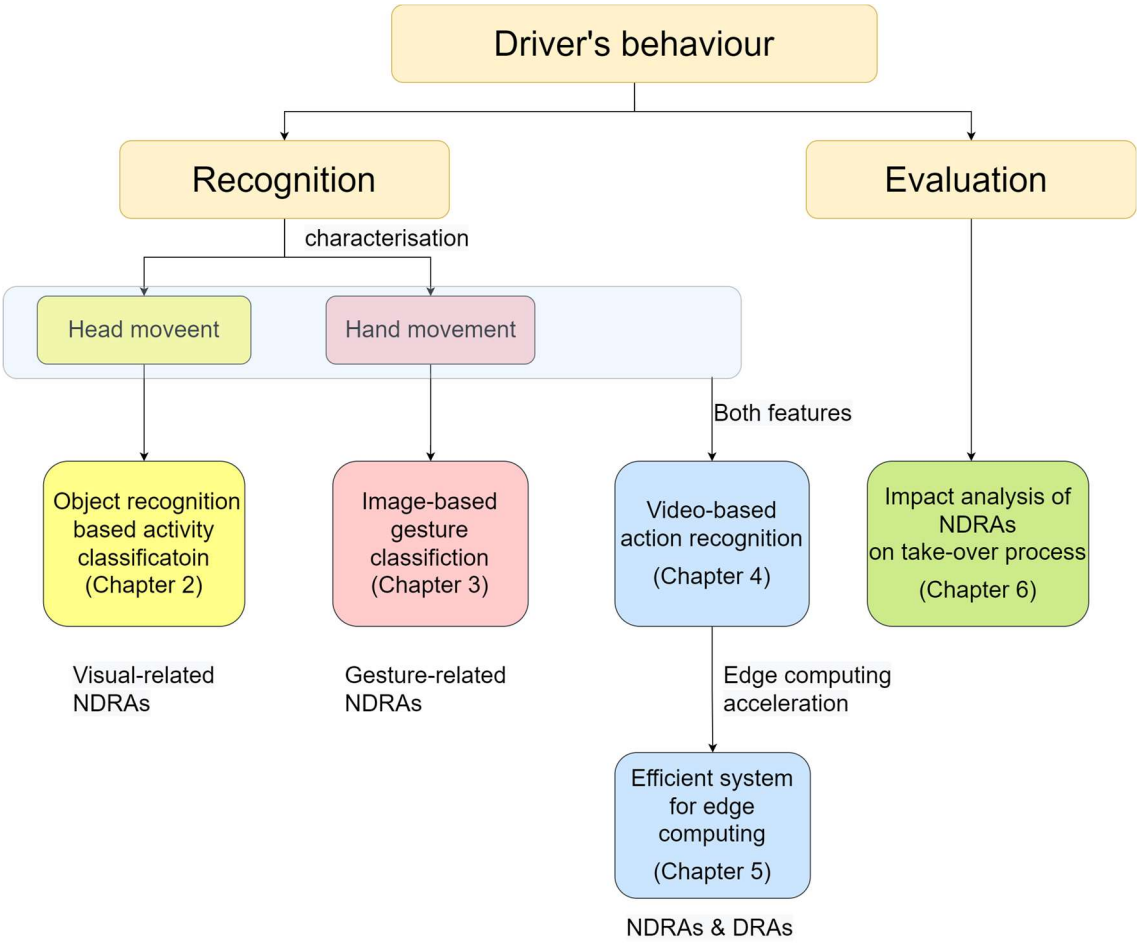


Figure 1-3 Thesis structure

characterised as head and hand movement. Based on these features, Chapter 1 (2 papers included) and 3 explore the ways to recognise some types of NDRAs. Chapter 1 proposed a gaze estimation and mapping-based method to identify the visual-related NDRAs. Chapter 3 consider the NDRAs recognition a pure classification problem based on the driver's hand movement. These two chapters prove the feasibility of using driver's visual attention and hand movement to recognise some certain NDRAs, respectively. Furthermore, Chapters 4 and 5 combine both driver's head and hand movement to achieve driver behaviour recognition, which is not only limited to the NDRAs recognition. It can also determine whether the driver engages in some DRAs like road-checking. Specifically, Chapter 4 proposes a novel action recognition method through extracting the spatial-temporal features from videos. Chapter 5 further presents an efficient model to achieve behaviour monitoring with edge computing devices, which aims to be used in real vehicle applications. Chapter 6 proposes a method

Table 1-1 Experiment and NDRAs dataset for each chapter

Term	Chapter 1	Chapter 3	Chapter 4	Chapter 5	Chapter 6
Collection platform	1,2	1	2	2	2
Processing platform	1	1	2	3	1
Experiment design	1,2	3	4	4	5
NDRAs dataset	1	2	3	3	-

to category the NDRAs into different groups and investigated the impact of the NDRAs engagement on the takeover process at both group and individual levels. Such a study could guide the further design of the takeover strategy and modality. The experiment setting and employed NDRAs datasets for each chapter are presented in Table 1-1. In the end, Chapter 7 discusses the outcomes of the research in terms of its impact on the real world and gives a general conclusion and future work.

1.7 Reference

[1] R. Hussain and S. Zeadally, “Autonomous Cars: Research Results, Issues, and Future Challenges,” *IEEE Commun. Surv. Tutorials*, vol. 21, no. 2, pp. 1275–1313, 2019, doi: 10.1109/COMST.2018.2869360.

[2] B. W. Smith, “SAE levels of driving automation,” *Cent. Internet Soc. Stanford Law Sch.*, p. 1, 2014.

[3] “Automated Driving Systems: A Vision for Safety 2.0,” Feb. 2017. [Online]. Available: https://www.nhtsa.gov/sites/nhtsa.gov/files/documents/13069a-ads2.0_090617_v9a_tag.pdf.

[4] Y. Song, M. V. Chitturi, and D. A. Noyce, “Automated vehicle crash sequences: Patterns and potential uses in safety testing,” *Accid. Anal. Prev.*, vol. 153, p. 106017, Apr. 2021, doi: 10.1016/j.aap.2021.106017.

[5] National Transportation Safety Board, “Collision Between a Car Operating With Automated Vehicle Control Systems and a Tractor-Semitrailer Truck,”

Natl. Transp. Saf. Board, p. 63, 2017, [Online]. Available: https://www.ntsb.gov/investigations/AccidentReports/Reports/HAR1702.pdf.

[6] National Transportation Safety Board, “Collision Between a Sport Utility Vehicle Operating With Partial Driving Automation and a Crash Attenuator, Mountain View, California, March 23, 2018,” *Natl. Transp. Saf. Board*, 2018.

[7] M. Sivak and B. Schoettle, “Motion Sickness in Self-Driving Vehicles,” no. April, 2015, [Online]. Available: https://deepblue.lib.umich.edu/handle/2027.42/111747.

[8] F. Naujoks, D. Befelein, K. Wiedemann, and A. Neukum, “A Review of Non-driving-related Tasks Used in Studies on Automated Driving,” in *Advances in Intelligent Systems and Computing*, vol. 597, 2018, pp. 525–537.

[9] E. Shi and A. T. Frey, “Non-driving-related tasks during level 3 automated driving phases–measuring what users will be likely to do.,” *Technol. Mind, Behav.*, vol. 2, no. 2, Jul. 2021, doi: 10.1037/tmb0000006.

[10] L. Petersen, L. Robert, X. J. Yang, and D. Tilbury, “Situational Awareness, Driver’s Trust in Automated Driving Systems and Secondary Task Performance,” *SAE Int. J. Connect. Autom. Veh.*, vol. 2, no. 2, pp. 12-02-02–0009, May 2019, doi: 10.4271/12-02-02-0009.

2 Visual attention-related NDRAs recognition

This chapter introduces a method to recognise the visual attention-related NDRAs based on the estimation of the driver's gaze, which is based on two published papers:

1. *L. Yang, K. Dong, A. J. Dmitruk, J. Brighton, and Y. Zhao, "A Dual-Cameras-Based Driver Gaze Mapping System With an Application on Non-Driving Activities Monitoring," IEEE Trans. Intell. Transp. Syst., vol. 21, no. 10, pp. 4318–4327, Oct. 2020, doi: 10.1109/TITS.2019.2939676.*

2. *L. Yang, K. Dong, Y. Ding, J. Brighton, Z. Zhan, and Y. Zhao, "Recognition of visual-related non-driving activities using a dual-camera monitoring system," Pattern Recognit., vol. 116, p. 107955, Aug. 2021, doi: 10.1016/j.patcog.2021.107955.*

2.1 Object-based NDRAs recognition with the gaze mapping system

2.1.1 Introduction

In a Level 3 automated vehicle, according to the SAE (J3016) Automation Levels definition, the driver could engage in some non-driving related activities (NDRAs) when the vehicle is under the automated driving mode [1]. However, since the level of full automation has not been reached, the driver is still expected to respond appropriately to a takeover request from the vehicle [2]. NDRAs could affect the driver's hazard awareness and a high attention level on the NDRA could result in a negative effect on driving quality [3] or even accidents during the transition of control between the vehicle and driver [4], [5]. Therefore, the effect of various NDRAs on takeover quality needs to be investigated and evaluated. In recent years, some studies have been reported to evaluate the take-over performance (e.g. reaction time and driving quality) after switching from NDRAs [6], [7] and the effects of Human-Machine Interaction (HMI) design supporting this activity switch and monitoring the driving environment [8]. There is very limited literature investigating the identification and tracking of NDRAs automatically. Sivak and Schoettle [9] reported that the main NDRAs that drivers engage in the

UK are *reading* (9.9%), *sleeping* (9.4%), *texting* or *talking with friends* (7.1%), *working* (6.4%) and *watching movies* (5.4%). Since NDRAs in a level 3 and above automated vehicle are diverse and the type and engaged duration of NDRAs will lead to different take-over performance [4], it is necessary to classify/identify and track them automatically for designing an intelligent HMI for takeover.

As objects and human poses are feature-rich, human-object interaction has been widely investigated in the early stages of human action recognition [10], [11], through the integration of object recognition, pose estimation and action identification [12]. For successful NDRA identification, the driver is constrained on the seat, space limitation and body occultation pose a challenge for driver action estimation. Le *et al*. [13] proposed a convolutional neural network (CNN)-based approach to achieve the driver behaviour parsing. It localises some body parts of the driver like head, hand, etc. by semantic segmentation in still images to achieve the detection of some actions, such as hands on steering wheel and hands on phone. Several deep learning-based approaches have been proposed for video-based human action recognition, with the development of artificial intelligence in multi-object detection [10]. Such approaches extend object detection to action detection through the multi-stream CNN, which combines the spatial and temporal information [14], [15]. It recognises the action by using the moving parts of the human body instead of pose estimation. Some CNN-based approaches have been proposed for the NDRA or secondary task recognition in recent years. Xing *et al*. [16] used the image of the driver's body by removing the background, as the input of the CNN model to recognise NDRAs. Eraqi *et al*. [17] extended inputs by including raw images, skin-segmented images, face images, hands images, and "face+hands" images. Then trained CNN model for each stream is further used to obtain the final prediction using a genetic algorithm based on their outputs. Yang *et al*. [18] proposed a 2-stream CNN based system, which extracts the spatial features from raw images and the movement features from the corresponding optical flow images to achieve NDRA recognition. However, such CNN-based NDRA recognition approaches mainly focus on encoding the specific movement of the driver. It lacks the capability of tracking

the driver's visual attention. Studying the driver's visual attention can directly determine whether the driver is engaging in NDRAs, which is important to develop an intelligent HMI design for a safe take-over. As most NDRAs (e.g. *reading, texting, working* and *watching movies*) require interaction between objects (e.g. book, tablet, or dashboard) and the human eye (gaze), this chapter proposes a novel framework for gaze-related NDRA identification and tracking that consists of three parts: object recognition, gaze estimation and an activity classifier. Several object detection frameworks have been proposed such as YOLO [19] and Faster R-CNN [20]. These frameworks can recognise a few general objects in real-time, but they lack semantic segmentation and accurate outliner detection. Since object segmentation is needed in this proposed framework, the Mask R-CNN [21] is used as part of the proposed framework. The eye gaze features have been applied in some applications of advanced driver-assistance systems (ADAS) for the purpose of distraction and fatigue detection [22], [23] or gaze attention estimation [24]. The developed gaze estimation systems mainly focus on the modelling of the eye-gaze based on the image captured by the camera, which is in front of the human face [25], [26]. Since the image used in these systems have no further information about the activity that the driver could engage in. It can not be used directly for driver behaviour recognition. The applications of gaze estimation for ADAS are normally driver gaze zone estimation [27]. Fridman *et al*. [28] allocated the driver's gaze into different regions by extracting their facial features with a single camera. Xiao and Feng [29] proposed a driver's visual attention system by using a smartphone, in this method, the rear camera is used to capture the moving object and the front camera is used to estimate the driver's gaze. The view of the rear camera is divided into 9 zones, and the system aims to check if the driver is aware there is a moving object inside these zones. Both studies made a fixed assumption between the eye gaze direction and the driver's behaviour, which is not applicable for characterisation of NDRAs due to its high complexity and uncertainty. Therefore, to further implement the gaze estimation method into NDRAs recognition, the estimated gaze needs to be mapped into a view, which contains the driver's behaviour in the vehicle cabin.

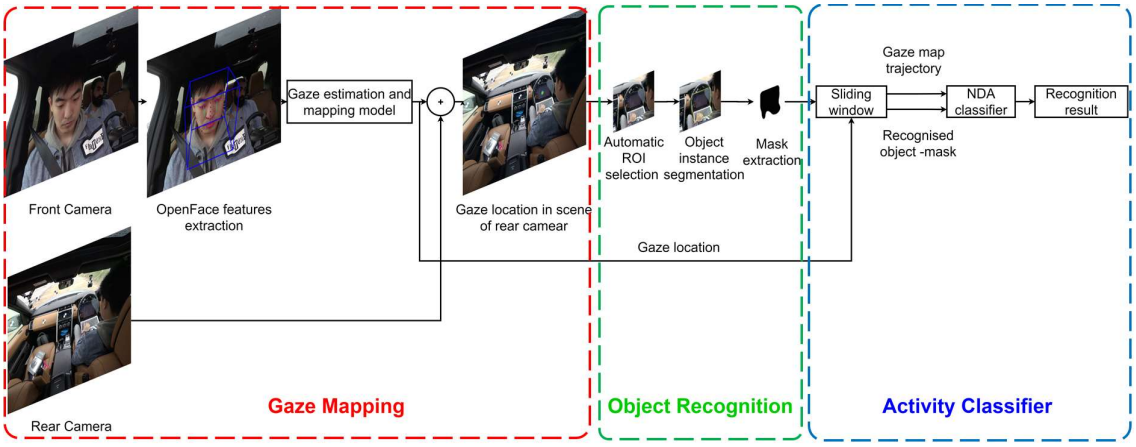


Figure 2-1 The proposed framework for NDRA identification that consists of three parts: gaze mapping, object recognition and activity classifier

This chapter presents a non-intrusive and cost-effective dual-camera based NDRA identification and tracking framework. It maps the driver's eye gaze, achieved by the first/front camera facing the driver, and the object scene is captured by the second/rear camera, using a complex system modelling technique, called Volterra Non-linear Regressive with eXogenous inputs (VNRX) model. The object is automatically recognised and located through the Mask R-CNN algorithm. Based on the mapped gaze and the location of the segmented object, an activity classier using the sliding time window technique is proposed to identify and track the type of NDRA.

2.1.2 Methodology

2.1.2.1 Framework architecture

The proposed framework has 3 components: gaze mapping, object recognition and an NDRA classifier. As shown in Figure 2-1, the driver's gaze is estimated by a dual-camera system. The front camera is used to capture and extract the driver's facial and gaze features which are used to estimate the gaze which is then mapped into the scene of the rear camera and visualised by a heat map. The estimated gaze location in the scene of the rear camera helps define a region of interest (ROI) for object recognition, which will significantly reduce the recognition time and increase the accuracy and success rate. The recognition result shows the object label, confidence score and location of each object

represented by an object-mask list. The sliding time window technique is used to construct a novel NDRA classifier for decision-making through considering the historic information of eye gaze location and recognised object masks. The details of each part are introduced below.

2.1.2.2 Gaze estimation

2.1.2.2.1 System framework

The framework of the gaze estimation system is divided into four steps including video acquisition, feature extraction, gaze mapping and heat map visualisation. As shown in the flowchart illustrated by Figure 2-2, the first feed of video is captured through a camera placed in front of the driver, as indicated by Camera 1, to capture the facial features including eye gaze and head movement. The second feed of video is captured through a camera placed above the driver, referred to as Camera 2 in Figure 2-2, to mimic the driver's view. The driver's gaze directions along with other parameters including face location and orientation are extracted based on videos from Camera 1. These parameters are considered as the inputs of the model for gaze mapping. The proposed method tends to include the driver's facial features as more as possible and let the later modelling/mapping process determine which features should be included for

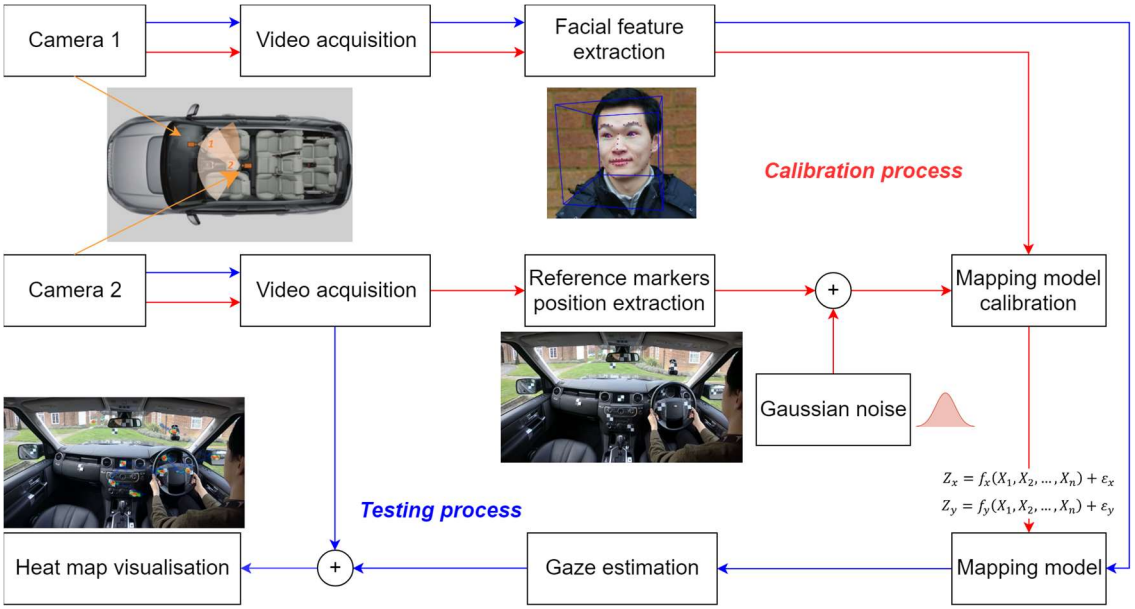
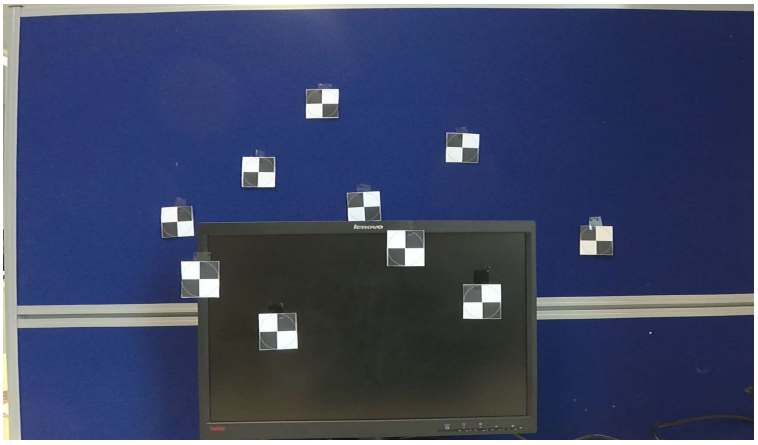


Figure 2-2 The flowchart of the gaze mapping system. There are two processes in this system including calibration in red and testing in blue

estimating the output, the mapped location of eye gaze in images of Camera 2. The mapping model calibration is to establish a model to represent the relationship between the face features in images of Camera 1 and eye-gaze locations on images of Camera 2. In the calibration (or training) process, the eye-gaze location in images of Camera 2 is known using markers placed on the vehicle. This chapter assumes that the gaze is a region with an approximately Gaussian distribution which represents the driver's observation intensity [30]. Gaussian noise with a pre-set sigma is therefore applied on the marker locations on images from Camera 2, as the known outputs of training data. From the system identification point of view, adding noise to the desired output can reduce overfitting and improve model generalisation. A large value of sigma will reduce the accuracy of fitting but improve the model generalisation. In this research, the

(a)



(b)

Figure 2-3 (a) The spatial distribution of the markers in Land Rover Discovery 4 for the in-vehicle experiment. (b) The spatial distribution of the markers in laboratory for the indoor experiment

sigma value was chosen as 10 pixels to achieve the optimal balance. Once this relationship is established, this model can be deployed on face features extracted from a testing video of Camera 1 and produce a mapping on the scene captured by Camera 2.

Considering the NDRA as a dynamical process, this study focuses on the eye gaze on a certain time window and a form of the heat map is proposed for visualisation. The details of each step are presented below.

2.1.2.2.2 Video acquisition

The Land Rover Discovery 4 was used as the test vehicle. Camera 1 is located in the windshield in front of the driver. The location of Camera 2 is set on the top of the driver towards the windscreen. The markers shown in Figure 2-3 (a) were placed on the strategic locations inside the vehicle including the dashboard, side mirrors, rear-view mirror, windscreen, multimedia display and steering wheel etc. These locations are fixed and friendly for the driver to look at. In this study, a total number of 12 markers were used.

Before implementing the system in the vehicle, a feasibility study has been conducted in the laboratory to better evaluate the performance of the proposed system. The layout of the cameras is the same as mentioned above. Ten markers are located randomly and shown in Figure 2-3 (b). It should be noted that there are 4 markers on the monitor which has a shorter distance to Camera 2.

For both experiments, the employed cameras were Garmin Virb Action Camera. Camera 1 provides the video with a resolution of 1024×768 pixels and 24 frames per second. Since a wider field of view is requested for Camera 2, the resolution is set as 1440×1080 pixels and the temporal resolution remains the same value.

2.1.2.2.3 Feature Extraction

In recent years, several gaze and head tracking methods have been proposed [31], [32]. As one of the most popular open-source facial analysis tools, OpenFace is utilized for the purpose of extracting the features of the driver's gaze and head due to its fine performance and robustness. It is capable of facial landmark detection and action unit recognition, head pose and eye-gaze

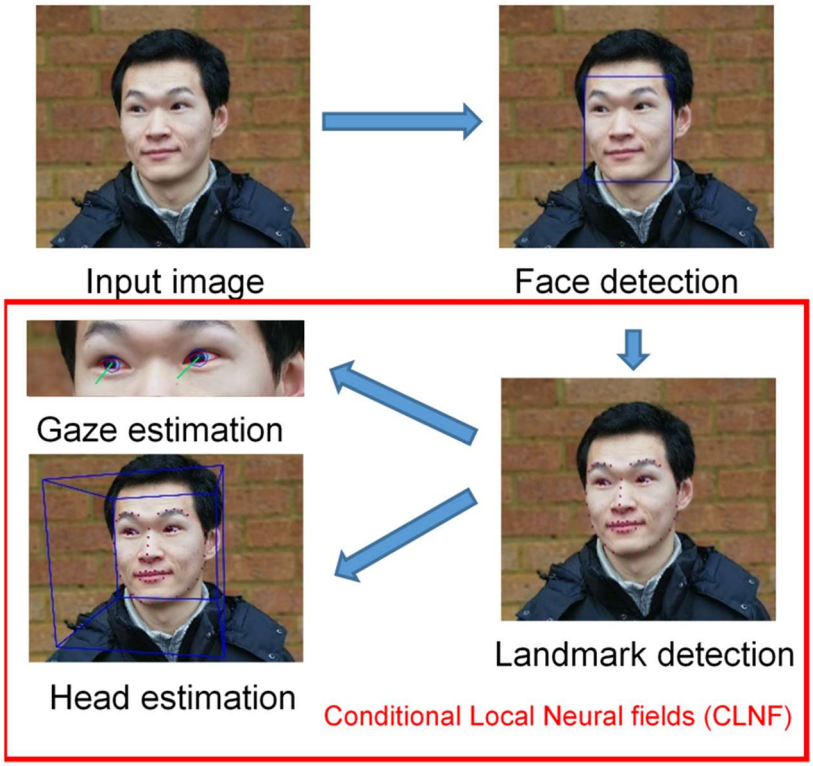


Figure 2-4 OpenFace facial behaviour analysis process

estimation [25], [33]. The algorithm starts with face detection then is followed by the 68 facial landmarks detection. These landmarks are used to estimate the head pose and track the eye gaze. The process is illustrated in Figure 2-4. Conditional Local Neural Fields (CLNF) framework is utilized as a shape registration approach for detecting the facial landmarks [34]. There are two components for CLNF which are Point Distribution Model (PDM) and patch experts. PDM captures variations of the landmark shape and the local appearance variations of each landmark are captured by patch experts. For head pose estimation, the orthographic camera projection is used to project the 3D representation of facial landmarks. The SynthesEyes training dataset [35] is used to train the PDM and CLNF patch experts for the eye-region registration task. Once the eye and the pupil are located, the data are used to calculate the gaze vector for each eye. The gaze estimation ability of this model is validated by the MPIIGaze dataset [36]. The performance of this approach on driver monitoring has been evaluated in the research of Zhao *et al*. [37].

Considering the complexity and uncertainty of the driver's behaviours during NDRAs, this chapter proposes to use both head information and gaze information

to build up the gaze heat map. The selected parameters are divided into two categories: the head pose related parameters (HRPs) and the gaze related parameters (GRPs). HRPs include the position of the detected head with respect to Camera 1, denoted by *pose_Tx, pose_Ty* and *pose_Tz*, and head orientation in 3D, denoted by *pose_Rx, pose_Ry* and *pose_Rz*. GRPs include the information of the gaze direction in radians, denoted by *gaze_angle_x* and *gaze_angle_y*. The direction vector is an average value for both eyes in world coordinates.

2.1.2.2.4 Feature Mapping

This research proposes to use the orthogonal least squares (OLS) algorithm to establish the correspondence between the face features based on the coordinate of Camera 1 and the eye gaze mapping based on the coordinate of Camera 2. This is an approach that has been used in nonlinear system identification where OLS searches through all possible candidate model terms to select the most effective ones to build the model. The significance of each selected model term is measured by the ERR index which indicates how much of the change in the system response, in percentage, can be accounted for by including the relevant model terms. This capability is important for this study because the facial features have been extracted as more as possible to ensure the proposed system can accommodate the diversity of driver's behaviour, meanwhile, we need to avoid producing an over-complex model that over-fits the training data and produces relatively poor testing performance. This algorithm allows us to only select the important face features for modelling to reach the balance between model complexity and gaze estimation performance. Furthermore, the capability to accommodate nonlinear modelling is important to cope with the distortion of images of Camera 2, which is the by-product where a wide field of view is required.

The Volterra Non-linear Regressive with eXogenous inputs (VNRX) model, also known as nonlinear finite impulse response (NFIR) model, is used in this research to represent a multi-inputs and single-output system, where the inputs are the facial features and the output is the eye gaze location on images of Camera 2. It

should be noted that the eye gaze location includes two values: x and y, which will be modelled independently. The models can be expressed as:

$$Z_x = f_x(X_1, X_2, \dots, X_n) + \varepsilon_x \quad \textbf{(2-1)}$$

$$Z_y = f_y(X_1, X_2, \dots, X_n) + \varepsilon_y \quad \textbf{(2-2)}$$

where $X_1, X_2, \dots, X_n$ are the face features; n is the number of collected face features; Z_x and Z_y are the eye gaze location in x and y direction respectively; f_x and f_y are some unknown linear or nonlinear mappings link the inputs and output; ε_x and ε_y are module residual.

Consider a function in a linear form:

$$Y(k) = \sum_{i=0}^{N} \theta_i p_i(k), k = 1,2, \dots, M \quad \textbf{(2-3)}$$

where $Y(k)$ is the system output (eye gaze location in x or y direction), $p_i(k)$ are regressors constructed by input variables, θ_i is the vector of unknown coefficients of regressions to be estimated, M denotes the number of data points in the training data set, and N denotes the number of terms in the model that is yet to be determined. If the model order is set as q, the candidate term set where $p_i(k)$ select from, denoted by C, can be expressed

$$C = C_1 \cup C_2 \cup \dots \cup C_l \cup \dots \cup C_q \quad \textbf{(2-4)}$$

where C_1 is the linear term set, expressed as

$$C_1 = \bigcup_{a=1}^{n} X_a \quad \textbf{(2-5)}$$

and C_2 is the 2nd order nonlinear term set, expressed as

$$C_2 = \bigcup_{a_1=1}^{n} \bigcup_{a_2=a_1}^{n} X_{a_1} X_{a_2} \quad \textbf{(2-6)}$$

and C_l is the l^{th} order nonlinear term set, expressed as

$$C_l = \bigcup_{a_1=1}^{n} \bigcup_{a_2=a_1}^{n} \cdots \bigcup_{a_l=a_{l-1}}^{n} \prod_{i=1}^{l} X_{a_i} \quad \textbf{(2-7)}$$

Equation (2-3) is re-written as

$$Y = P\Theta \quad \textbf{(2-8)}$$

where

$$Y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(M) \end{bmatrix}, P = \begin{bmatrix} p^T(1) \\ p^T(2) \\ \vdots \\ p^T(M) \end{bmatrix}, \Theta = \begin{bmatrix} \theta(1) \\ \theta(2) \\ \vdots \\ \theta(M) \end{bmatrix} \quad \textbf{(2-9)}$$

and $P^T(k) = \big(p_1(k), p_2(k), \dots, p_N(k)\big)$. Matrix P can be decomposed as $P = W \times A$ where

$$W = \begin{bmatrix} w_1(1) & w_2(1) & \dots & w_N(1) \\ w_1(2) & w_2(2) & \dots & w_N(2) \\ \vdots & \ddots & \ddots & \vdots \\ w_1(M) & w_2(M) & \dots & w_N(M) \end{bmatrix} \quad \textbf{(2-10)}$$

and $A = \{a_{ij}\}$ is an upper triangular matrix with unity diagonal elements. Equation (2-4) is then rewritten as

$$Y = WG \quad \textbf{(2-11)}$$

where $G = A\Theta = [g_1 \quad g_2 \quad \cdots \quad g_N]^T$. Equation (2-11) is now ready to represent the relation between Y and G.

We then estimate the importance of each model term to the variation of the system output. Initially, set values $a_{ij} = 0$ for $i \neq j$ (A then becomes an identity matrix), so $w_1(k) = p_1(k)$, and calculate g_1 as

$$g_1 = \frac{\sum_{k=1}^{M} w_1(k) y(k)}{\sum_{k=1}^{M} {w_1}^2(k)} \quad \textbf{(2-12)}$$

For $j = 2,3, \dots, M$, set $a_{jj} = 1$ and then calculate

$$a_{ij} = \frac{\sum_{k=1}^{M} w_i(k) p_j(k)}{\sum_{k=1}^{M} {w_i}^2(k)} \quad \textbf{(2-13)}$$

where $i = 1,2, \ldots, j-1$. Next, the algorithm calculates

$$w_j(k) = p_j(k) - \sum_{i=1}^{j-1} a_{ij} w_i(k) \quad \textbf{(2-14)}$$

and

$$g_j = \frac{\sum_{k=1}^{M} w_j(k) y(k)}{\sum_{k=1}^{M} {w_j}^2(k)} \quad \textbf{(2-15)}$$

The ERR value for each term p_i is finally defined as

$$ERR_i = \frac{g_i^2 \sum_{k=1}^{M} {w_i}^2(k)}{\sum_{k=1}^{M} y^2(k)} \quad \textbf{(2-16)}$$

Values of ERR range always from 0% to 100%. The larger the ERR the higher dependence between the $\{p_i\}$ terms and the output. Therefore, it is an indicator to represent the importance of each term (constructed by the face features as inputs) to the output.

The estimation of the coefficient of each selected term can be computed from

$$\left.\begin{aligned} \hat{\theta}_N &= \hat{g}_N \\ \hat{\theta}_i &= \hat{g}_i - \sum_{k=i+1}^{N} a_{ik} \theta_k, i = N-1, \ldots, 1 \end{aligned}\right\} \quad \textbf{(2-17)}$$

Through the above algorithm, a polynomial model based on Equation (2-3) can be established for each direction of the eye gaze location. The models can then be used for estimation of eye gaze location by given the face features.

2.1.2.2.5 Heat Map Visualisation

The heat map is a common visualisation approach to represent the spatial distribution of the data [38]. This research assumes that the eye gaze at a certain time or frame (t) can be represented by a circle which is defined by three parameters: $x_0(t)$ and $y_0(t)$, the location of the centre, and d, the diameter of the

circle. The spatial distribution inside the circle follows the Gaussian distribution. The value of d is affected by the image resolution of Camera 2. It was set as 40 pixels for gaze visualisation.

Considering at the frame t, the eye gaze centred at $(x_0(t), y_0(t))$, the intensity of the pixel(x, y) in the heat map, where $\left(x - x_0(t)\right)^2 + \left(y - y_0(t)\right)^2 \leq d^2$, can be defined as

$$S(x, y, t) = \mathrm{e}^{\left(-\frac{(x-x_0(t))^2+(y-y_0(t))^2}{2\sigma^2}\right)} * 100\% \quad \textbf{(2-18)}$$

The intensity of the pixels unsatisfied with the constraints is set as 0.

To represent the trajectory of gaze, this study integrates the gaze spatial distribution within a certain time window $[t - h,\ t]$, where t is the number of the frame and h is the window length. The accumulated eye gaze can be written as

$$S_a(x, y, t) = \frac{1}{h}\sum_{i=0}^{h-1} S(x, y, t - i) \quad \textbf{(2-19)}$$

To better visualise the gaze trajectory in real-time, this research proposes a weighted accumulation of eye gaze to construct the trajectory, written as

$$S_c(x, y, t) = \frac{1}{h}\sum_{i=0}^{h-1} S(x, y, t - i) * (1 - \frac{i}{h}) \quad \textbf{(2-20)}$$

The value of $S_a(x, y, t)$ and $S_c(x, y, t)$ is between 0 and 1. The window length h can be adjusted in terms of various applications.

2.1.2.3 Object recognition

CNN-based algorithms have achieved critical advances for the object recognition problem. The object recognition models based on CNNs can be categorised into two different types: one-stage and two-stage. Two-stage models such as Faster R-CNN [20] and Mask R-CNN [21] usually produce higher accuracy than one-stage models such as YOLO [19] and SSD [39] but they perform under lower detection speed. To match the requirement of this framework, this chapter selects the Mask R-CNN model, which is an extension of the Faster R-CNN model for

pixel-to-pixel instance segmentation task. There are two reasons for this selection: 1) compared with the recent frameworks training the COCO dataset, Mask R-CNN outperforms Faster R-CNN, YOLO and SSD in terms of accuracy and the speed is acceptable; 2) Mask R-CNN extends previous frameworks and locates exact pixels of each object instead of only bounding boxes, which is important for this study because the region of the object must be accurate to determine if the eye gaze is located in this region.

Instance segmentation is a challenging task that combines two independent processes: object detection and semantic segmentation. The multi-task scheme could create spurious edges and produce systematic errors in overlapping instances [40]. To solve this problem, Mask R-CNN extends Faster R-CNN by adding a branch for predicting segmentation masks in a pixel-to-pixel manner, in parallel with the existing branch for classification and bounding box regression. The core operation in Faster R-CNN for attending to instances, RoIPool, performs coarse spatial quantisation for feature extraction [41]. To fix the misalignment, Mask R-CNN replaces the RoIPool layer with a simple and quantisation-free layer which is called RoIAlign and faithfully preserves exact spatial locations. The RoIAlign layer uses bilinear interpolation to compute the exact values of the input features at four regularly sampled locations in each ROI bin and then performs max or average pooling on features. In spite of being a seemingly minor change, RoIAlign improves mask accuracy significantly. The proposed Mask R-CNN architecture is illustrated in Figure 2-5. It should be noted that the input image is

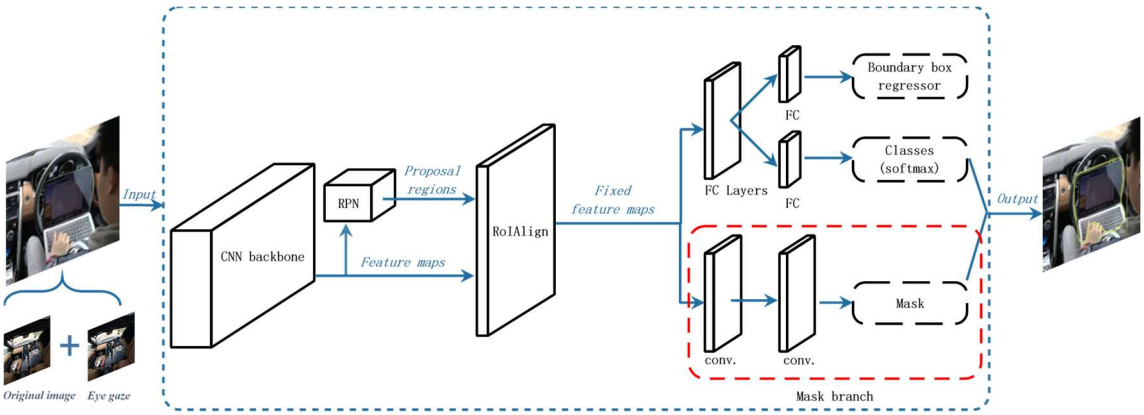


Figure 2-5 The Mask R-CNN architecture for the object recognition

a cropped image considering the eye gaze. The size of this ROI is a parameter to set considering the size of the targeted objects.

There are several implementations of Mask R-CNN so far. This research selected the *maskrcnn_benchmark* for the proposed system due to its best performance in training and inference. *Maskrcnn_benchmark* is up to twice as fast as a *Detectron* while matching and exceeding Detectron accuracy [42]. There are 5 NDRAs (involving 5 types of objects) considered in this chapter where the phone, the laptop, and the book can be detected with the COCO-pre-trained model, but a tablet and car interior cannot be detected with this pre-trained model. A dedicated database was then created for the latter cases. The dataset consists of 200 images acquired from the rear camera for each object including tablet, control console, wing mirror, windscreen, rear-view mirror, and dashboard. We selected the ResNet-101-FPN as the backbone network and trained these data starting with a learning rate of 0.001, which was divided by 10 after every 30 epochs. The total number of epochs was set as 100. All training works were implemented on an NVIDIA Quadro P6000 graphics card machine which has 24 GB DDR5X memory.

2.1.2.4 Activity classifier

The hypothesis of this study is that activities that required visual attention can be identified by estimating the driver's gaze and recognising the object that is gazed on. Therefore, the inputs of the classifier are the representation of the driver's gaze map and the recognised object-masks (there could be multiple objects in the ROI). Considering that the driver's behaviour during the engagement of NDRA is continuous, the historical temporal information is crucial for activity recognition. The proposed classifier employs the sliding time window technique to enhance the resilience to noise.

The driver's gaze trajectory with a certain time window h is presented in Equation (2-20). For further decision making, the trajectory of gaze is binarized as:

$$\bar{S_c}(x,y,t) = \begin{cases} 1 & if\ S_c(x,y,t) \geq T_g \\ 0 & otherwise \end{cases} \quad \textbf{(2-21)}$$

where T_g is the threshold.

The object could be easily occulted by the driver's hand during the NDRA engagement, which could lead to the poor performance of object recognition. To increase the robustness of overall performance, the recognised object-masks are estimated based on the historical information within the time window. The list of the segmented mask of objects at the time t, expressed as $N(x,y,t) = \{N_1(x,y,t), N_2(x,y,t), \dots, N_k(x,y,t)\}$, is a set of binary images, where k is the total number of recognised objects. Since the ROI to produce $N(x,y,t)$ is selected based on the gaze location $Z(t)$, to create an object-mask within a time window h, an offset needs to be considered as the selected ROI could be different for each time step. The revised mask for the i^{th} object by removing the offset can be expressed as:

$$N_{\text{offset},i}(x,y,t) = N_i\big(x - Z_x(t), y - Z_y(t), t\big) \tag{2-22}$$

The final recognised object-mask for the i^{th} object is achieved by calculating the union of all masks for this object within this window. It can be expressed as

$$N_i(x,y,t) = \bigcup_{j=0}^{h-1} N_{\text{offset},i}(x,y,t-j) \tag{2-23}$$

Finally, the intersection of the binarized trajectory of gaze $\overline{S_c}(x,y,t)$ and each recognised object-mask $ON_i(x,y,t)$ is calculated. The one that has the maximal area of intersection is selected as the recognised class l, which can be written as

$$l(t) = \underset{1\le i\le k}{\operatorname{argmax}} \|\overline{S_c}(x,y,t) \cap ON_i(x,y,t)\| \tag{2-24}$$

where $\|\cdot\|$ indicates the operation to calculate the area.

2.1.3 Results

In this section, the performance of the gaze estimation, object recognition and the NDRAs identification is given. Specifically, the gaze estimation system is evaluated in both laboratory and vehicle.

2.1.3.1 Gaze estimation

2.1.3.1.1 Indoor experiment

Two tests were conducted in this experiment based on the level of freedom of head movement. In the first test, the participant was asked to gaze at the 10 markers one by one avoiding moving head forwards or backwards, so the shift of eye gaze was primarily achieved by head rotation. In the second test, the participant was given more freedom of head movement and both rotation and translation were allowed, aiming to simulate the increased complexity and uncertainty of head movement during NDRAs. The participant was required to gaze at each marker for at least 5 seconds. The data of the transition period when moving from one marker to another was removed. A total number of 1000 frames (100 frames per marker) were selected for training and testing. For each marker, 70% of data were randomly selected for training and the remaining 30% of data were for testing.

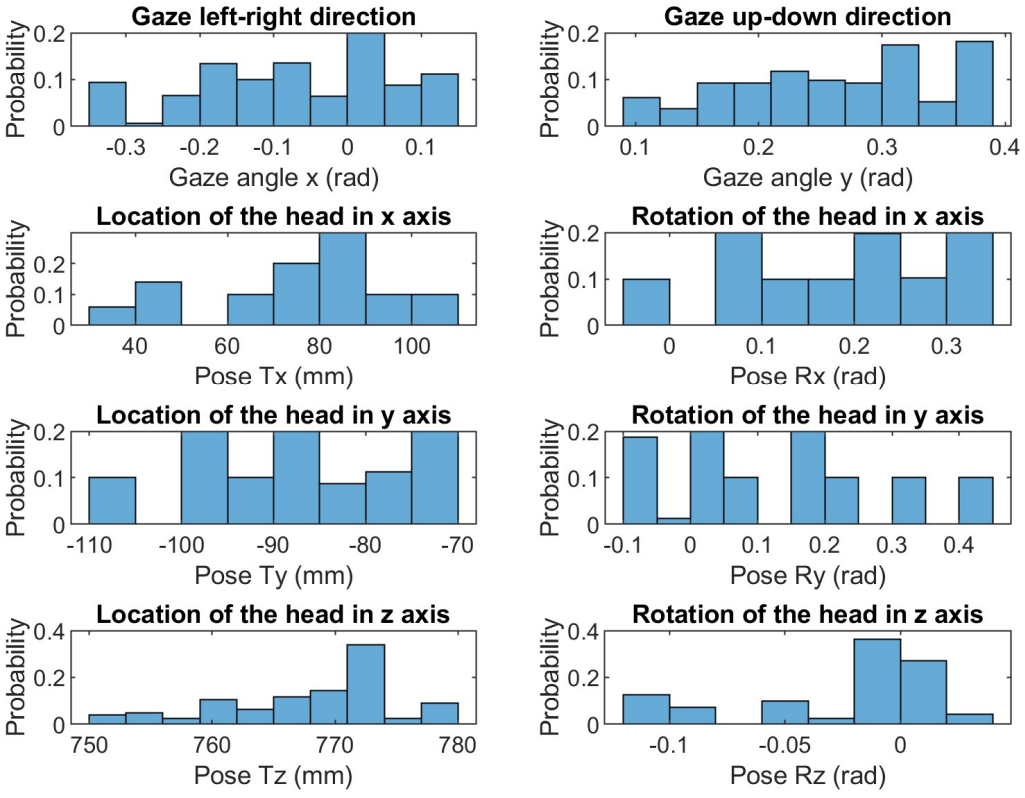


Figure 2-6 Histograms of the facial features of the training data for the model calibration of the first test of the indoor experiment

Figure 2-6 presents the histograms of eight facial features of the training data in the first test. It can be observed that the head rotation movement is within 0.5 rad in pitch (*pose_Rx*) and yaw (*pose_Ry*). The roll movement of the head (*pose_Rz*) is relatively small, within 0.15 rad, which is expected because there is not much rolling movement required to scan all markers. The variation of head position in the z-axis (*pose_Tz*), indicating the distance from the head to Camera 1, is within 30 mm. Although the translation of head was limited in this test, the head rotation caused a small variation of head depth.

Table 2-1 shows an example of the estimated 2nd-order nonlinear models of gaze in X and Y directions. The number of the model term is limited to 10. The model term is ranked based on the ERR value which represents the importance of each model term to the variation of gaze. It can be observed from Table 2-1 that the most important term is 'constant' for both X and Y which refers to the baseline of the head movement and relates to the initial state of the participant. As expected, the second important term is the gaze angle for the considered direction. It is interesting to observe that HRPs also make a significant contribution to the model, which suggests that both HRPs and GRPs must be considered due to the complexity of human behaviour and the distortion of cameras. To quantify the performance of the proposed system in the first test, the produced models were

Table 2-1 An example of the estimated 2nd order nonlinear model for the first test of the indoor experiment

Model	X		Y	
Priority	Model term	Coefficient	Model term	Coefficient
1	constant	811.25	constant	528.27
2	gaze_angle_x	-1582.93	gaze_angle_y	474.31
3	pose_Tz* pose_Rx	53.43	pose_Tx	-2.03
4	gaze_angle_y	-917.28	pose_Tx* pose_Rx	-35.57
5	pose_Tx	-0.58	gaze_angle_x	465.13
6	pose_Tx* pose_Rx	-46.94	gaze_angle_y* gaze_angle_y	821.15
7	pose_Rx	853.26	pose_Ty* pose_Ty	0.21
8	pose_Rx* pose_Ry	-4419.04	gaze_angle_y* pose_Rz	16140.28
9	pose_Ry	450.32	gaze_angle_x* gaze_angle_y	-2771.10
10	gaze_angle_y* pose_Tz	-62.47	pose_Rx	1248.56

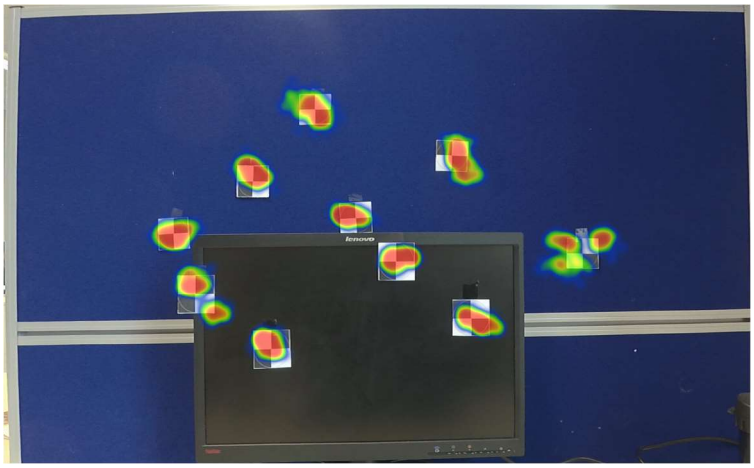
Table 2-2 Model performance comparison of the indoor experiment

Term	Root Mean Square Error	
	Pixel	Millimetre
Test1_X	11.89 ± 9.00	9.25 ± 7.00
Test1_Y	9.22 ± 6.55	7.17 ± 5.10
Test2_X	27.33±14.29	21.26 ± 11.12
Test2_Y	20.71±11.51	16.11 ± 8.95

applied in the testing data and the Root Mean Square Error (RMSE) of the estimated gaze location and the centre of marker (without adding Gaussian noise) for all markers was computed to represent the model accuracy. Since the testing data were randomly selected, this process was repeated 1000 times to ensure statistical significance. Table 2-2 provides the mean (overall accuracy of the model) and the standard deviation (precision of model) of the 1000 calculated accuracies for two tests. The size of the markers in the mapping frame is 37 pixels (28.8 mm). From the results of the first test in Table 2-2, it can be observed that the error of gaze estimation in the X direction is 11.89±9.00 pixel (9.25±7.00 mm), while for the Y direction the error is smaller with a value of 9.22±6.55 pixel (7.17±5.10 mm). The errors of both directions are well smaller than the marker size, which indicates a fine performance of the proposed system when the head translation is limited. The performance of the Y direction is better than the X direction. This observation is reasonable because the makers cover a larger range of the X direction which leads to a higher level of distortion. The accumulated eye gaze map, calculated by Equation (2-19), is presented in Figure 2-7 (a), where all estimated gaze points well fall into the markers, although there are some slight shifts between the centre of the gaze circle and the centre of the markers.

In the second test, the head movement was more complex by introducing both translation and rotation of the head. It has been observed from Table 2-3 that the head position in the z-axis (*pose_Tz*) has a variation of 210 mm, which is 7 times

(a)



(b)

Figure 2-7 (a) The accumulated eye gaze mapping for the first test of the indoor experiment. (b) The accumulated eye gaze mapping for the second test of the indoor experiment

higher than the first test. The ranges of other features are similar to the ones of the first test. The second test aims to test the flexibility of the proposed mapping algorithm against the diverse head movement of NDRAs. Table 2-4 presents an example of the estimated 2nd-order nonlinear models of gaze in X and Y directions. The number of the model term is limited to 10. It can be observed that the top 2 terms are the same as the ones of the first test, however, HRPs make more contribution to the model evident by more appearance in the selected model terms, particularly *pose_Tz*. As shown in **Table A-1** and **Table A-2** in Appendix, the proportion of ERR of HRPs in the X direction is increased from 0.55% in the first test to 5.88% in the second test. The proposed algorithm successfully

Table 2-3 The data range comparison of the extracted facial features of the training data of the indoor experiment

Features	The first test	The second test
Gaze_angle_x	[-0.35, 0.15] rad	[-0.35, 0.25] rad
Gaze_angle_y	[0.08, 0.38] rad	[0.1, 0.4] rad
Pose_Tx	[30, 110] mm	[25, 115] mm
Pose_Ty	[-110, -70] mm	[-108, -70] mm
Pose_Tz	[750, 780] mm	[660, 870] mm
Pose_Rx	[-0.05, 0.35] rad	[-0.05, 0.38] rad
Pose_Ry	[-0.1, 0.45] rad	[-0.25, 0.45] rad
Pose_Rz	[-0.14, 0.04] rad	[-0.14, 0.07] rad

demonstrated flexibility by selecting terms including *pose_Tz* to reflect the increased variation of head translation.

Table 2-2 also shows the quantified performance of the second test, using the same approach as the first test. It is shown that the RMSE in the X direction is 27.33±14.29 pixel (21.26 ± 11.12 mm) and 20.71±11.51 pixel (16.11 ± 8.95 mm) for the Y direction. As expected, the overall performance is not as good as the first test due to the increased complexity of head behaviour, but the error is still smaller than the marker size (28.8 mm). It is interesting to observe that the performance in X and Y directions are similar for this case which suggests that the interference caused by camera distortion is overtaken by the interference caused by severe head movement. Figure 2-7 (b) illustrates the accumulated eye gaze map for the second test. In comparison with Figure 2-7 (a), the regions of the gaze estimation are larger and more irregular but still well cover the majority markers. It can be observed from Figure 2-7 that the visualised results of the 4 markers on the monitor which have a shorter distance to Camera 2 than other markers also show a similar performance, which demonstrates the robustness of the proposed system in terms of the depths of the object.

Table 2-4 An example of the estimated 2nd order nonlinear model for the second test of the indoor experiment

Model	X		Y	
Priority	Model term	Coefficient	Model term	Coefficient
1	constant	776.76	constant	545.12
2	gaze_angle_x	-2253.33	gaze_angle_y	1612.76
3	pose_Tz* pose_Ry	1.92	gaze_angle_x	-207.97
4	pose_Tx* pose_Tx	0.04	pose_Tz* pose_Tz	-0.01
5	pose_Ry	-863.78	pose_Tx* pose_Rx	-27.94
6	pose_Tx	-5.23	gaze_angle_y* gaze_angle_y	3018.13
7	gaze_angle_x* pose_Tz	-2.32	pose_Tx* pose_Ty	0.23
8	pose_Rz	-62.47	pose_Ry* pose_Rz	-2826.91
9	gaze_angle_x* pose_Rz	987.56	pose_Ry	-432.25
10	pose_Tz	0.46	pose_Ty	2.89

2.1.3.1.2 In-vehicle experiment

In this experiment, a wider field of view of Camera 2 in comparison to the in-door experiment was used due to limited space in the vehicle, which inevitably introduced more distortion on images. Furthermore, 12 markers were laid out on the regions of interest, which have more diverse distances to the plane of Camera 2 in comparison with the indoor tests. Due to these factors, a more sophisticated model is required to cope with the increased complexity. Therefore, a 3rd-order

Table 2-5 Model performance of the in-vehicle experiment

Term	Root Mean Square Error	
	Pixel	Millimetre
X	7.80 ± 5.99	12.00 ± 9.22
Y	4.64 ± 3.47	7.14 ± 5.34

Figure 2-8 The accumulated eye gaze mapping for the vehicle experiment

nonlinear model was estimated with the number of the model term of 25. It should be noted that in this experiment the participant was asked to scan the markers with the limited translation of head, as the first indoor test. The approach to select the training and testing data was the same as the indoor experiment.

As shown in Table 2-5, the RMSE in X and Y direction is 12.00± 9.22 mm and 7.14±5.34 mm respectively, which is well smaller than the marker size (28.8 mm). The performance is better than the first indoor test with a cost of increased model complexity. The error in the Y direction is almost half of that of the X direction which is due to the head movement range in the X direction being much larger than the range in the Y direction. The interference of distortion is therefore more significant in the X direction. The accumulated eye gaze mapping is visualised in Figure 2-8, which clearly demonstrates the fine performance of the proposed system.

2.1.3.2 Object recognition

A pre-trained Mask R-CNN model provides 81 categories based on the COCO dataset, which was used to recognise the book, the cell phone and the laptop.

The automatic ROI (with a size of 640×480 pixels) selection module based on the mapped gaze location was applied in this framework. In Figure 2-9, the dotted lines represent the object recognition performance with the automatic selection of ROI against different values of the confidence threshold while the solid lines represent those without ROI. It can be clearly seen that the recognition success rate with ROI is consistently higher than that without ROI for all ranges of the confidence threshold across three types of objects. This is probably due to the reduced interference of other objects in the raw image. It is expected that following the increase of the confidence threshold, the recognition success rate decreases. It is shown in Figure 2-9 that when the threshold is 0.6, the recognition success rates with ROI for all 3 kinds of objects are above 80%. A low confidence threshold leads to a high risk of misrecognition of the object, which could result in the accuracy decrease of NDRA identification, and it will also increase the system computational cost. A high confidence threshold leads to a high risk of missing the targeted object, which results in the failure of NDRA identification. To

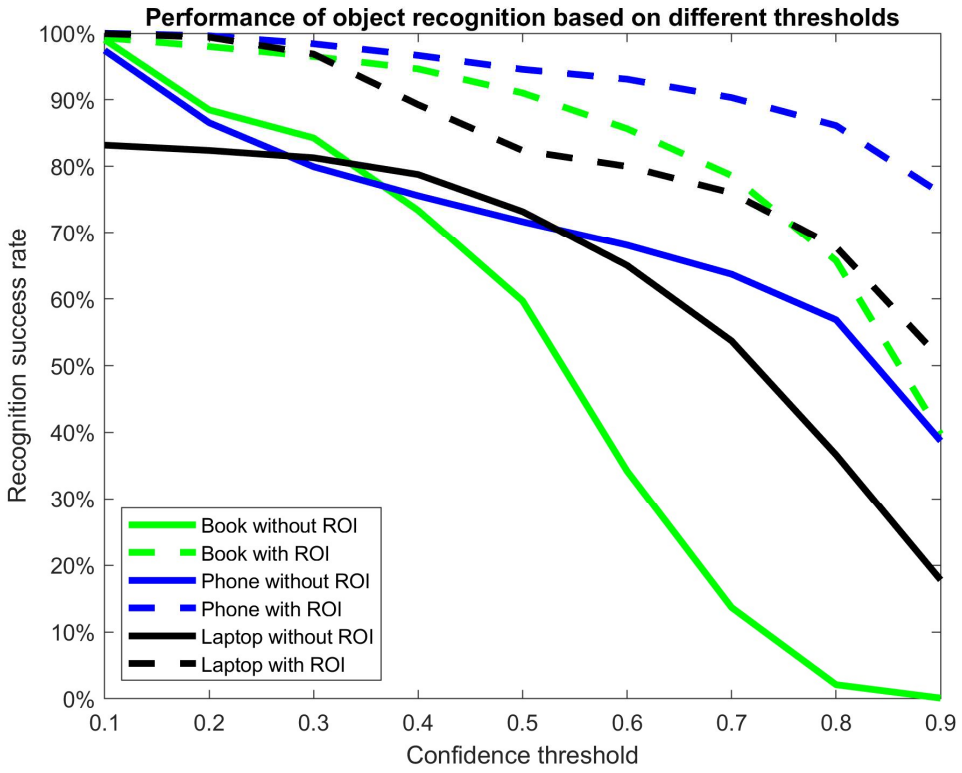


Figure 2-9 Comparison of object recognition performance based on the different confidence threshold for book, phone and laptop

balance this trade-off, the confidence threshold for the below results was set as 0.6.

Figure 2-10 plots the performance comparison of object recognition for images with and without ROI for each participant. An increase in the success rate is shown and is significant for all participants and objects. It should be noticed that the success rate of participant 4 with ROI for book recognition is around 44%, the reason is addressed below. The success rates of participant 5 for laptop recognition are almost 0% due to a heavy occlusion caused by cloth.

Table 2-6 shows the overall performance of both accuracy and processing time by averaging all participants. Phone recognition shows the highest success rate (93.10%), probably due to the smallest size of the object. From phone, book, to laptop, the object size becomes bigger, and it is observed that the success rate becomes lower. This is because a large object has a high possibility of being covered by the human body, which leads to high dissimilarity with the training

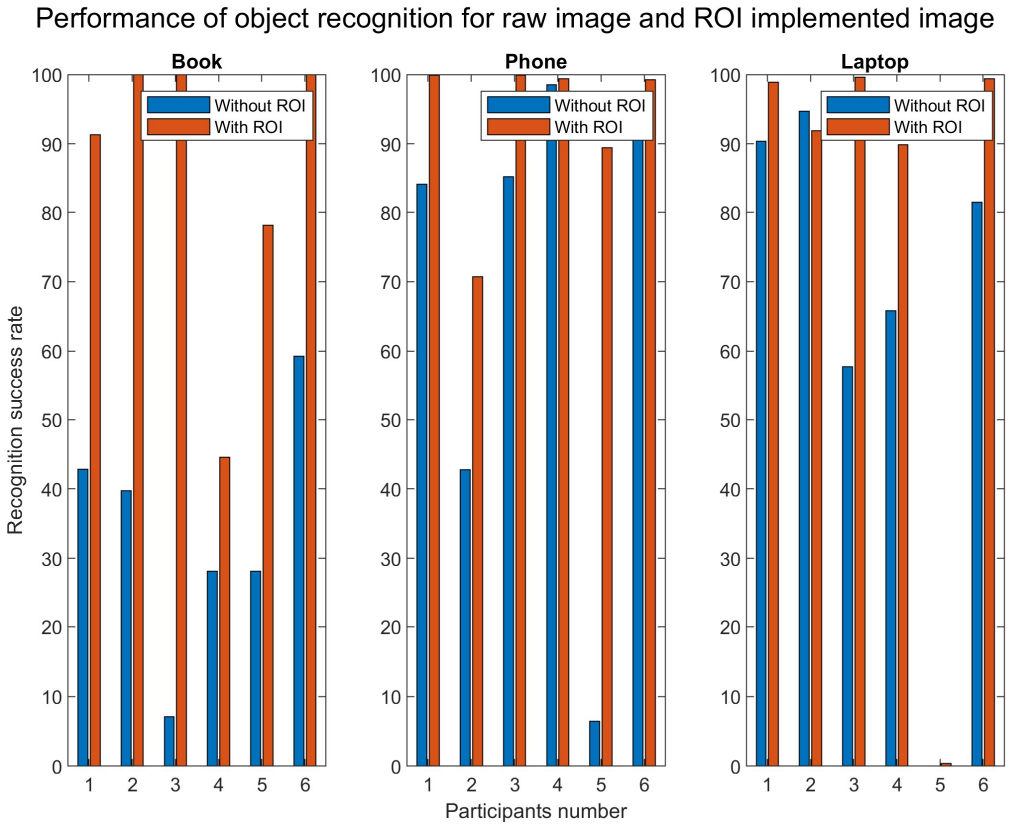


Figure 2-10 Object recognition performance comparison for raw image and ROI implemented image based on all participants

Table 2-6 Comparison of performance of object recognition for data with and without ROI

Object	Success rate		Processing time per image (s)	
	Without ROI	With ROI	Without ROI	With ROI
Book	34.20%	85.66%	0.462	0.191
Phone	68.07%	93.10%	0.443	0.181
Laptop	65.01%	80.00%	0.476	0.183
Average	55.76%	86.25%	0.460	0.185

data. An increment of more than 30% has been observed in terms of the average success rate with the gaze-based ROI detection implemented. The average processing time for all objects decreases from 0.460s to 0.185s with a time reduction percentage of 60%. There is no significant difference in processing time in terms of the type of object.

For a tablet and vehicle interior, since a dedicated training database was developed for a specific tablet and vehicle only, the success rates are almost 100%. To extend its application on other types of tablets and vehicles, much more training data are required, which is not the focus of this research.

2.1.3.3 NDRAs identification and analysis

The recognised object mask and the gaze trajectory map were used to identify the type of NDRA that the driver is engaging in. Table 2-7 presents the identification accuracy of 5 tested NDRAs for all participants. For the NDRA of *reading a book*, the average accuracy is 85.04% with a standard deviation of 19.56%. The high value of standard deviation is caused by the result of participant 4 which is only 43.07%. High identification accuracy has been achieved for the NDRAs of playing phone and playing tablet, with an averaged value of 90.11% and 99.64% respectively. The standard deviations across participants are less than 10%. The performance of the NDRA of working on a laptop is more than 85% except for participant 5 caused by the failure of object recognition due to cloth occlusion. The average accuracy of the NDRA of interacting with the centre

Table 2-7 NDRAs identification accuracy for all participants

NDRAs	Participants						Average
	1	2	3	4	5	6	
Reading a book	99.80%	100.00%	88.74%	43.07%	84.67%	93.94%	85.04% ± 19.56%
Playing a phone	100.00%	72.78%	92.26%	91.14%	87.40%	97.08%	90.11% ± 8.75%
Working on a laptop	96.35%	77.98%	81.51%	79.96%	6.26%	96.56%	73.10% ± 30.82%
Playing a tablet	100.00%	100.00%	98.68%	100.00%	99.17%	100.00%	99.64% ± 0.53%
Interacting with centre console	96.66%	86.17%	80.09%	70.84%	76.54%	87.71%	83.00% ± 8.34%

console is 83.00%, which is relatively lower than others and the reason that will be explained below.

Table 2-8 presents the contribution of the error caused by object recognition and gaze mismatch, which provides a deeper insight into how the object recognition and gaze trajectory estimation affect the prediction results. For the NDRA of reading a book, ER is larger than EM for all the participants. Specifically, the low NDRAs recognition accuracy of participant 4 is mainly caused by object recognition failure (53.49%). For the NDRA of playing a phone, ER for participant 2 shows a high value, which is 22.53%. Apart from that, the EM is larger than ER,

Table 2-8 Error contribution of the NDRAs. ER and EM refers to the error caused by recognition failure and mismatch, respectively

NDRAs	Participants											
	1		2		3		4		5		6	
	ER	EM	ER	EM	ER	EM	ER	EM	ER	EM	ER	EM
Reading a book	0%	0.20%	0%	0%	8.44%	2.82%	53.49%	3.44%	10.12%	5.21%	3.45%	2.61%
Playing a phone	0%	0%	22.53%	4.69%	1.15%	6.59%	1.98%	6.88%	5.11%	8.49%	0.94%	1.98%
Working on a laptop	0.66%	2.99%	10.20%	11.82%	8.15%	10.34%	9.62%	10.42%	88.89%	4.85%	1.19%	2.25%
Playing a tablet	0%	0%	0%	0%	0%	1.32%	0%	0%	0%	0.83%	0%	0%
Interacting with centre console	0%	3.34%	0%	13.83%	0%	19.92%	3.54%	25.62%	0%	23.46%	0%	12.29%

which suggests that the inaccurate gaze estimation is the main contribution of the NDRAs recognition failure. Since the size of the phone is normally small, enlarging the diameter of the estimated gaze region could increase the NDRAs identification accuracy. The main contribution of the low playing a laptop recognition accuracy is the body occultation, which affects the recognition of the object recognition (especially for participant 5) and also the NDRAs identification (the occulted object cannot match with the right gaze map). For the NDRA of interacting with the centre console, the recognition error is mainly caused by unmatching. It is because of a lower gaze estimation accuracy.

Figure 2-11 presents some snapshots of eye gaze mapping and object recognition, which suggests that in most of the cases the eye gaze is well located

Figure 2-11 NDRAs identification visualisation examples. These images are cropped from raw images for appropriate visualisation

inside the recognised object. For participant 5 who is working on a laptop, it can be seen that the estimated gaze is inside the laptop while the laptop is failed to be recognised, which could be solved by adjusting the position of the rear camera. For the NDRA of reading a book, the book cannot be consistently recognised, for example with participant 4, which is suffering from the shadow and strong illumination caused by the sun. This interference reduces the accuracy of the book recognition and further affects the performance of NDRA identification. This is the main reason that the identification result of reading a book for participant 4 (43.07%) is significantly lower than the results of others. Compared with other NDRAs, the behaviour during interacting with the centre console shows a different pattern. The eye gaze is not always well located in the

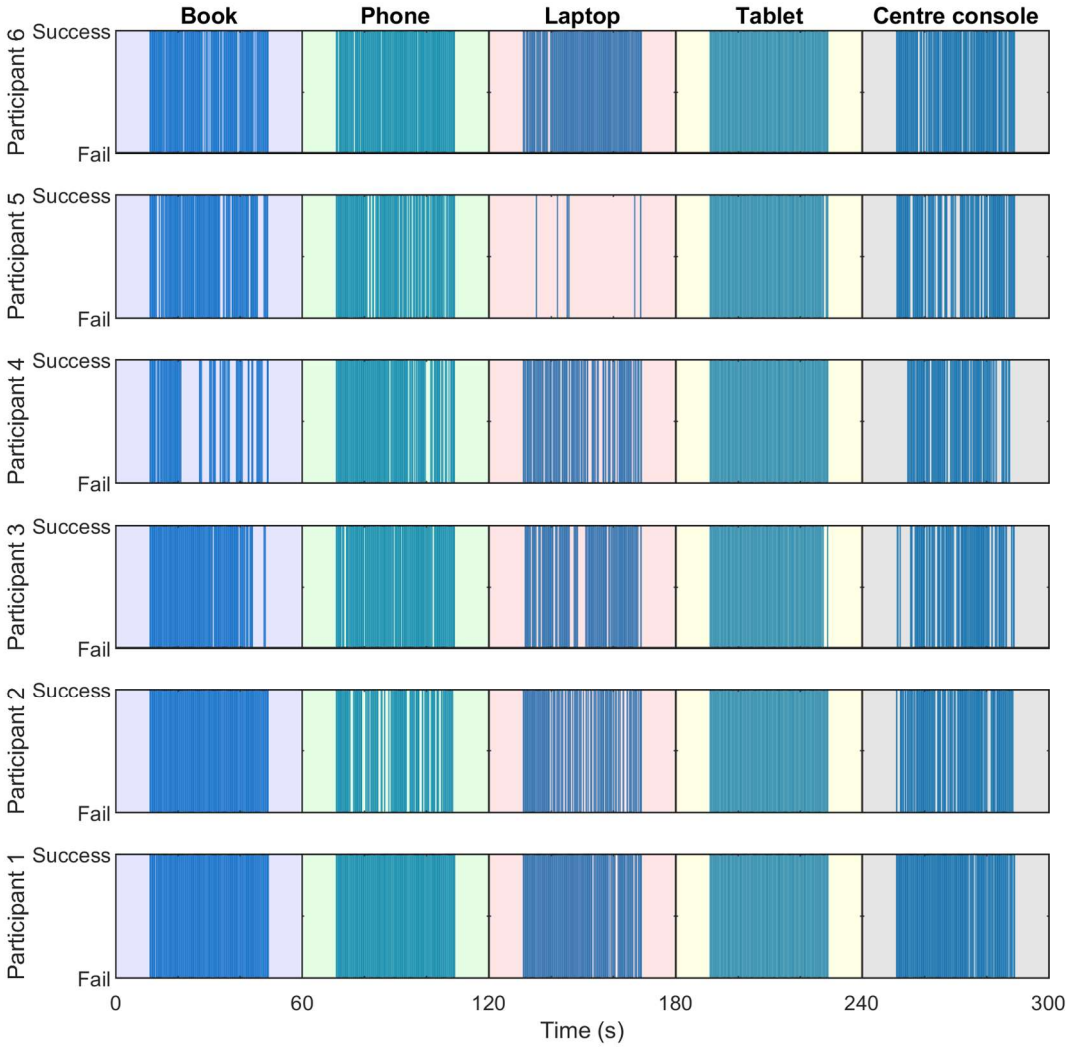


Figure 2-12 NDRAs identification and tracking for all participants. Five activities are distinguished by different colours of the background

centre of the centre console although the object is always well recognised, which leads to a relatively low NDRA identification accuracy. There is a relatively large head rotation and body movement towards the left side which lead to a relatively large error of gaze mapping.

To show the performance of NDRA tracking, the blue bar in Figure 2-12 represents the successful identification of NDRAs. It should be noted that only the middle 40 s for each NDRA was analysed which justifies the large blank areas at the beginning and ending stages of each NDRA. It can be observed that there are some discrete failures of identification due to failed object recognition or inaccurate gaze estimation. An additional reason could be that the participants are looking away from the object, which is highly possible in real applications. To improve the accuracy of tracking, a large time window size in the classifier is suggested. However, it will sacrifice the performance of tracking the rapid change of NDRAs.

2.1.3.4 Comparison with the state-of-the-art

The proposed approach has been compared with some state-of-the-art methods from the perspective of both action recognition and specific NDRAs recognition, which are

(1) ResNet-50 [43]. It has a 50 layer 2D CNN architecture and achieves the action classification in the spatial domain. A pre-trained model on the ImageNet dataset is employed in this study.

(2) Two-stream CNN-based approach (2-stream) [18] for NDRAs recognition. This method uses the information of both the RGB image stream and its associated current and historical optical flow frames to achieve the classification of the NDRAs.

(3) 3D ResNets-18 (R3D18) [44]. It is based on ResNes-18 architecture that mainly utilises the 3D residual block in the whole network to encode the spatial-temporal information for action recognition.

Table 2-9 Comparison of the proposed method with 4 state-of-the-art methods on the NDRA dataset

Method	**ResNet-50**[43]	**2-stream**[18]	**R3D18**[44]	**R2+1D**[45]	**Ours**
Accuracy	81.8%	86.3%	86.5%	85.3%	86.2%

(4) (2+1)D ResNets (R(2+1)D) [45]. It factorises the 3D residual block in R3D18 into a 2D spatial residual block and a 1D temporal residual block. Compared to the 3D convolution with the same number of parameters, such a structure doubles the number of nonlinearities, which improves the model's capability of representing complex functions.

All methods were tested on the collected NDRA data. As mentioned before, 40s video data for each activity and each participant was used for NDRA recognition, which was split into 40 instances. All training and testing data were extracted from the rear camera by cropping a region that covers the human-object interaction. There are a total of 1200 instances in the dataset. k-fold cross-validation is employed to evaluate the models' performance based on the participants, where k is set as 3 in this study. For each k, data of 4 participants were used for training and the remaining 2 participants for testing.

The results are presented in Table 2-9. It can be observed that the proposed method achieves similar performance with other state-of-the-art methods, where ResNet-50 has relatively low accuracy. It should be noted that, firstly, most of the existing NDRAs recognition methods, including the selected 4 methods, focus on the hand interaction between the driver and the object. Such methods lack the investigation of the driver's visual attention, which could lead to a misdetection of NDRA engagement since the driver could check the road with their hand holding the object. The awareness of the driving environment is also important for the take-over strategy. Secondly, although the proposed method uses deep learning methods, it is fundamentally different from other compared methods. This method tends to be transparent and the type of activity is determined by considering the location of eye gaze and the type of object with the gaze. There is no further

Table 2-10 Model performance based on different model order for the in-vehicle experiment

Term	Root Mean Square Error (pixel)	
	X	Y
Linear	56.49 ± 11.09	20.21 ± 7.91
2nd order	10.39 ± 7.50	5.94 ± 4.49
3rd order	9.04 ± 6.77	4.87 ± 3.66

training process required to include a new type of activity. However, other compared methods will have to be trained again to include more activities. Thirdly, compared with the proposed method, the deep learning-based methods normally take a longer time to provide a prediction as information within a certain time window is used, which could slow down the system response.

2.1.4 Discussion

The performance of the proposed framework is largely affected by two factors: the accuracy of the driver's gaze estimation and the success rate of object recognition.

For the gaze estimation system, the order of the model and the number of the model term determine the model complexity which affects its performance. Table 2-10 presents the model performance based on different model orders, where the number of the model term was set as 20. It can be observed that the RMSE in the X direction has been reduced from 56.49 pixels to 10.39 pixels, equal to an 81.6% improvement of accuracy when the 2nd order model is used instead of the linear model. When the model complexity is increased from 2nd order to 3rd order, the increment of mode performance is much less significant (13% improvement). A similar pattern has been observed in the Y direction. On the one hand, the model is preferred to be as simple as possible to a) ensure low computational time for real-time applications, and b) avoid the over-fitting problem. On the other hand, the model should be sophisticated enough to cope with the interference of camera distortion and head movement. For all tests conducted in this study, the 2nd order nonlinear is appropriate. However, the

optimal model order can be different if a different camera is used. Generally speaking, a camera with high distortion requires a high order of model and more number of model terms. All these observations can be applied to the number of the selected model term. It is suggested to select the model as simple as possible as long as the error of estimation is smaller than the markers. If a high resolution of eye gaze mapping is required, smaller markers should be used.

The selection of the markers' location affects the system performance. In the vehicle experiment, some strategic locations were chosen such as the windscreen, wing mirrors, steering wheel and dashboard aiming to cover popular areas which the driver is often gazing on. Figure 2-13 plots the RMSE of estimated gaze on markers against the percentage of the distance from markers to the centre of the image to the image size. It can be seen that there is an average error of around 6 pixels for the markers around the centre, and the error increases following the increment of distance with an approximately quadratic

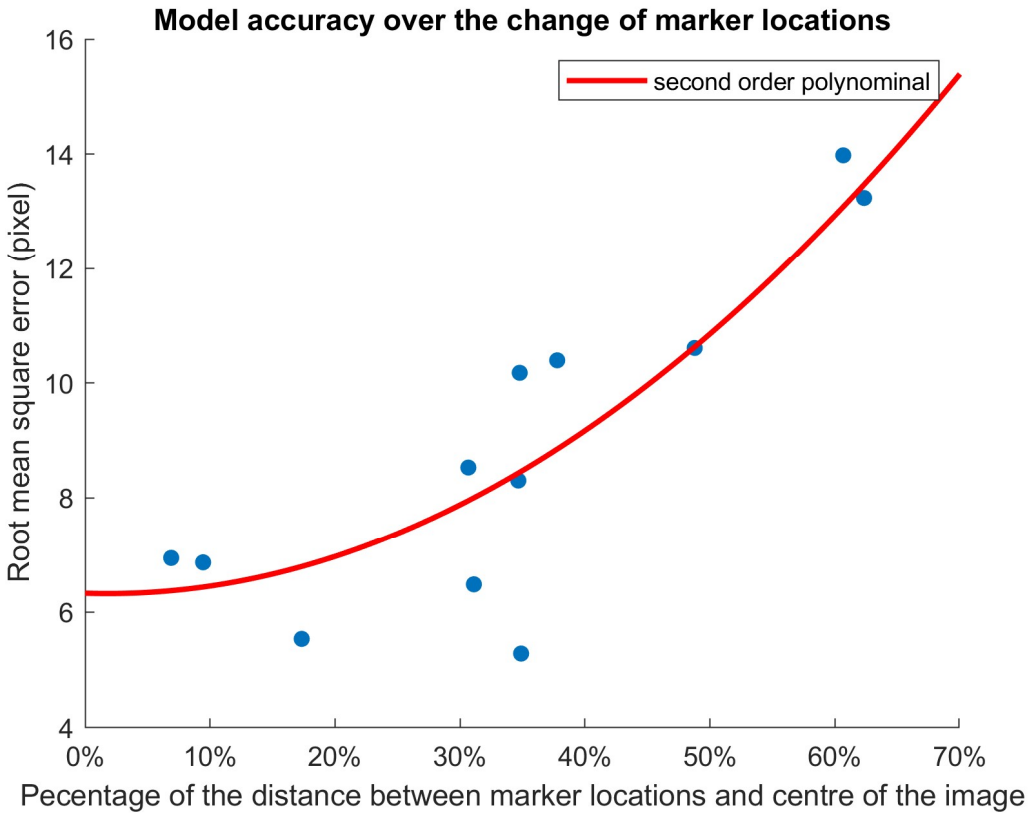


Figure 2-13 The model accuracy against the change of the marker locations, which suggests the influence of t distortion

relationship. This observation is clear evidence that the model performance is affected by distortion of the lens. Apart from the distortion, another reason for relatively poor performance on the edge of the image is caused by the OpenFace algorithm. When the driver gazes at the area around the edge of the image from Camera 2, the head rotation is usually large. The accuracy of facial features extracted by Camera 1 is compromised because some landmarks are hidden or partly visible. Using multiple cameras to capture the driver's facial features can address the problem but will increase the complexity and cost of the system.

For object recognition, there are a few challenges:

- To extend the universality of various models of a certain type of object, a large dataset such as COCO should be used. The dataset also should consider the diverse range of NDRAs.
- The main difference in object recognition between this study and other studies is that the driver is usually holding the object which inevitably leads to occlusion by hands or body if the camera position is not appropriate. The confidence level of recognition will be reduced. The confidence threshold, therefore, must be selected carefully. This problem is especially significant for small objects.
- The location of the rear camera must consider two factors: avoiding the occlusion of the human body on the object and reducing the noise caused by illumination. This problem is especially significant for large objects.
- Sunlight will cause the reflection of glass-surface objects such as the phone, the laptop and the tablet. It could decrease the recognisability or confidence score of object recognition.
- Other developed action recognition algorithms, which has the potential application for NDRA recognition, usually focus on the driver's hand location or the interaction between hand and object/device. However, in the real driving scenario, the driver is engaging in NDRA while observing surrounding situations. The proposed approach directly measures the driver's visual attention, which is crucial for further evaluation of the driver's awareness of the driving environment for a safe take-over transition.

2.1.5 Conclusions

This chapter proposes a dual-camera based NDRAs identification framework that benefits from computer vision, nonlinear system modelling and deep learning. It has been successfully demonstrated that this framework can identify the NDRAs which require visual attention. The main strengths of this technique are:

- The error of the gaze estimation system in the in-vehicle experiment for the X and Y direction is 7.80±5.99 pixels and 4.64±3.47 pixels respectively with an image resolution of 1440×1080 pixels.
- NDRAs required visual attention can be identified by inferring the object that the driver is looking at. The average success rate of this proposed framework is 86.18%. The performance is affected by both object recognition and gaze estimation, which could be further improved through creating the specific dataset for training and better locating the rear camera.
- The proposed gazed-based ROI module embedded in this framework contributes about a 30% improvement of average success rate and about a 60% decrease of processing time. The size of this ROI can be customised according to the resolution of the rear camera.
- The proposed active classifier improves the resilience to noise, such as the object can not be recognised suddenly due to occultation, by using a sliding time window.
- The research of driver distraction in a human-driving vehicle can benefit from this study. The proposed system can be used to detect the driver's distraction behaviour by extending the types of objects to recognise, such as side mirror checking behaviour, dashboard checking behaviour, etc.

The main limitations are:

- The proposed framework is not applicable to the NDRAs without visual attention, such as listening to music.
- The gaze estimation model could be subjective. A calibration process therefore is suggested for each driver before testing. Achieving a generic model to remove the calibration process requires further studies.

- As a camera-based approach, the performance of object recognition suffers from noise caused by harsh illumination, surface reflection and object occultation.
- It should be noted that the proposed solution is only based on the object that the driver is engaging in. It cannot show whether the driver is watching a video or texting a message when a phone is used. To refine these NDRAs, further studies are required.
- A potential problem to apply it in a driving vehicle is the facial feature extraction will be compromised due to the potential heavy movement of the driver body and camera movement caused by poor road condition

For future work, with the increasing computational capability of portable devices like mobile, some existing lightweight models for object recognition can be used for a portable device-based real-time NDRA recognition system. Furthermore, the tracking of NDRAs could determine the duration of the engagement. The impact of different durations for various NDRAs on driver's state and take-over performance needs to be evaluated, which is crucial for further design of take-over strategy to achieve the smooth and safe take-over transition.

2.1.6 References

[1] B. Wandtner, N. Schömig, and G. Schmidt, "Effects of Non-Driving Related Task Modalities on Takeover Performance in Highly Automated Driving," *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 60, no. 6, pp. 870–881, Sep. 2018, doi: 10.1177/0018720818768199.

[2] B. W. Smith, "SAE levels of driving automation," *Cent. Internet Soc. Stanford Law Sch.*, p. 1, 2014.

[3] S. H. Yoon and Y. G. Ji, "Non-driving-related tasks, workload, and takeover performance in highly automated driving contexts," *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 60, pp. 620–631, Jan. 2019, doi: 10.1016/j.trf.2018.11.015.

[4] S. H. Yoon, Y. W. Kim, and Y. G. Ji, "The effects of takeover request modalities on highly automated car control transitions," *Accid. Anal. Prev.*,

vol. 123, no. November 2018, pp. 150–158, Feb. 2019, doi: 10.1016/j.aap.2018.11.018.

[5] I. JEGHAM, A. BEN KHALIFA, I. ALOUANI, and M. A. MAHJOUB, “Safe Driving : Driver Action Recognition using SURF Keypoints,” in *2018 30th International Conference on Microelectronics (ICM)*, Dec. 2018, vol. 2018-Decem, no. Icm, pp. 60–63, doi: 10.1109/ICM.2018.8704009.

[6] B. Wandtner, N. Schömig, and G. Schmidt, “Secondary task engagement and disengagement in the context of highly automated driving,” *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 58, pp. 253–263, Oct. 2018, doi: 10.1016/j.trf.2018.06.001.

[7] F. Naujoks, S. Höfling, C. Purucker, and K. Zeeb, “From partial and high automation to manual driving: Relationship between non-driving related tasks, drowsiness and take-over performance,” *Accid. Anal. Prev.*, vol. 121, pp. 28–42, Dec. 2018, doi: 10.1016/j.aap.2018.08.018.

[8] A. Eriksson, S. M. Petermeijer, M. Zimmermann, J. C. F. de Winter, K. J. Bengler, and N. A. Stanton, “Rolling Out the Red (and Green) Carpet: Supporting Driver Decision Making in Automation-to-Manual Transitions,” *IEEE Trans. Human-Machine Syst.*, vol. 49, no. 1, pp. 20–31, Feb. 2019, doi: 10.1109/THMS.2018.2883862.

[9] M. Sivak and B. Schoettle, “Motion Sickness in Self-Driving Vehicles,” no. April, 2015, [Online]. Available: https://deepblue.lib.umich.edu/handle/2027.42/111747.

[10] H.-B. Zhang *et al.*, “A Comprehensive Survey of Vision-Based Human Action Recognition Methods,” *Sensors*, vol. 19, no. 5, p. 1005, Feb. 2019, doi: 10.3390/s19051005.

[11] M. Ziaeefard and R. Bergevin, “Semantic human activity recognition: A literature review,” *Pattern Recognit.*, vol. 48, no. 8, pp. 2329–2345, Aug. 2015, doi: 10.1016/j.patcog.2015.03.006.

[12] Bangpeng Yao and Li Fei-Fei, “Recognizing Human-Object Interactions in

Still Images by Modeling the Mutual Context of Objects and Human Poses," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1691–1703, Sep. 2012, doi: 10.1109/TPAMI.2012.67.

[13] T. H. N. Le, C. Zhu, Y. Zheng, K. Luu, and M. Savvides, "DeepSafeDrive: A grammar-aware driver parsing approach to Driver Behavioral Situational Awareness (DB-SAW)," *Pattern Recognit.*, vol. 66, no. December 2016, pp. 229–238, Jun. 2017, doi: 10.1016/j.patcog.2016.11.028.

[14] A. Ullah, K. Muhammad, I. U. Haq, and S. W. Baik, "Action recognition using optimized deep autoencoder and CNN for surveillance data streams of non-stationary environments," *Futur. Gener. Comput. Syst.*, vol. 96, pp. 386–397, Jul. 2019, doi: 10.1016/j.future.2019.01.029.

[15] J. Zhang and H. Hu, "Domain learning joint with semantic adaptation for human action recognition," *Pattern Recognit.*, vol. 90, pp. 196–209, Jun. 2019, doi: 10.1016/j.patcog.2019.01.027.

[16] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, and F.-Y. Wang, "Driver Activity Recognition for Intelligent Vehicles: A Deep Learning Approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5379–5390, Jun. 2019, doi: 10.1109/TVT.2019.2908425.

[17] H. M. Eraqi, Y. Abouelnaga, M. H. Saad, and M. N. Moustafa, "Driver Distraction Identification with an Ensemble of Convolutional Neural Networks," *J. Adv. Transp.*, vol. 2019, pp. 1–12, Feb. 2019, doi: 10.1155/2019/4125865.

[18] L. Yang *et al.*, "A refined non-driving activity classification using a two-stream convolutional neural network," *IEEE Sens. J.*, vol. XX, no. XX, pp. 1–1, 2020, doi: 10.1109/JSEN.2020.3005810.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, vol. 2016-Decem, pp. 779–788, doi: 10.1109/CVPR.2016.91.

[20] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[21] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.322.

[22] T. D’Orazio, M. Leo, C. Guaragnella, and A. Distante, “A visual approach for driver inattention detection,” *Pattern Recognit.*, vol. 40, no. 8, pp. 2341–2355, Aug. 2007, doi: 10.1016/j.patcog.2007.01.018.

[23] T. D’Orazio, M. Leo, and A. Distante, “Eye detection in face images for a driver vigilance system,” in *IEEE Intelligent Vehicles Symposium, 2004*, 2004, pp. 95–98, doi: 10.1109/IVS.2004.1336362.

[24] F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi, “Driver Gaze Tracking and Eyes Off the Road Detection System,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2014–2027, Aug. 2015, doi: 10.1109/TITS.2015.2396031.

[25] T. Baltrusaitis, P. Robinson, and L.-P. Morency, “OpenFace: An open source facial behavior analysis toolkit,” in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2016, pp. 1–10, doi: 10.1109/WACV.2016.7477553.

[26] A. Kar and P. Corcoran, “A Review and Analysis of Eye-Gaze Estimation Systems, Algorithms and Performance Evaluation Methods in Consumer Platforms,” *IEEE Access*, vol. 5, pp. 16495–16519, 2017, doi: 10.1109/ACCESS.2017.2735633.

[27] S. Vora, A. Rangesh, and M. M. Trivedi, “Driver Gaze Zone Estimation Using Convolutional Neural Networks: A General Framework and Ablative Analysis,” *IEEE Trans. Intell. Veh.*, vol. 3, no. 3, pp. 254–265, Sep. 2018, doi: 10.1109/TIV.2018.2843120.

[28] L. Fridman, P. Langhans, J. Lee, and B. Reimer, “Driver Gaze Region Estimation without Use of Eye Movement,” *IEEE Intell. Syst.*, vol. 31, no. 3, pp. 49–56, May 2016, doi: 10.1109/MIS.2016.47.

[29] D. Xiao and C. Feng, “Detection of drivers visual attention using smartphone,” in *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, Aug. 2016, pp. 630–635, doi: 10.1109/FSKD.2016.7603247.

[30] A. T. Duchowski, “A breadth-first survey of eye-tracking applications,” *Behav. Res. Methods, Instruments, Comput.*, vol. 34, no. 4, pp. 455–470, Nov. 2002, doi: 10.3758/BF03195475.

[31] M. La Cascia, S. Sclaroff, and V. Athitsos, “Fast, reliable head tracking under varying illumination: an approach based on registration of texture-mapped 3D models,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 4, pp. 322–336, Apr. 2000, doi: 10.1109/34.845375.

[32] E. Skodras and N. Fakotakis, “Precise localization of eye centers in low resolution color images,” *Image Vis. Comput.*, vol. 36, pp. 51–60, Apr. 2015, doi: 10.1016/j.imavis.2015.01.006.

[33] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L.-P. Morency, “OpenFace 2.0: Facial Behavior Analysis Toolkit,” in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, May 2018, pp. 59–66, doi: 10.1109/FG.2018.00019.

[34] T. Baltrusaitis, P. Robinson, and L.-P. Morency, “Constrained Local Neural Fields for Robust Facial Landmark Detection in the Wild,” in *2013 IEEE International Conference on Computer Vision Workshops*, Dec. 2013, pp. 354–361, doi: 10.1109/ICCVW.2013.54.

[35] E. Wood, T. Baltruaitis, X. Zhang, Y. Sugano, P. Robinson, and A. Bulling, “Rendering of Eyes for Eye-Shape Registration and Gaze Estimation,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, vol. 2015 Inter, pp. 3756–3764, doi: 10.1109/ICCV.2015.428.

[36] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, “Appearance-based gaze estimation in the wild,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, vol. 07-12-June, pp. 4511–4520, doi: 10.1109/CVPR.2015.7299081.

[37] Y. Zhao *et al.*, “An Orientation Sensor-Based Head Tracking System for Driver Behaviour Monitoring,” *Sensors*, vol. 17, no. 11, p. 2692, Nov. 2017, doi: 10.3390/s17112692.

[38] R. Netzel and D. Weiskopf, “Hilbert attention maps for visualizing spatiotemporal gaze data,” in *2016 IEEE Second Workshop on Eye Tracking and Visualization (ETVIS)*, Oct. 2016, pp. 21–25, doi: 10.1109/ETVIS.2016.7851160.

[39] W. Liu *et al.*, “SSD: Single Shot MultiBox Detector,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9905 LNCS, 2016, pp. 21–37.

[40] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, “Fully Convolutional Instance-Aware Semantic Segmentation,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, vol. 2017-Janua, pp. 4438–4446, doi: 10.1109/CVPR.2017.472.

[41] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, “Object Detection With Deep Learning: A Review,” *IEEE Trans. Neural Networks Learn. Syst.*, pp. 1–21, 2019, doi: 10.1109/TNNLS.2018.2876865.

[42] “Faster R-CNN and Mask R-CNN in PyTorch 1.0.” https://github.com/facebookresearch/maskrcnn-benchmark (accessed Apr. 20, 2019).

[43] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[44] K. Hara, H. Kataoka, and Y. Satoh, “Learning Spatio-Temporal Features with 3D Residual Networks for Action Recognition,” in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, Oct. 2017, vol. 2018-Janua, pp. 3154–3160, doi: 10.1109/ICCVW.2017.373.

[45] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, “A Closer Look at Spatiotemporal Convolutions for Action Recognition,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6450–6459, doi: 10.1109/CVPR.2018.00675.

3 Hand gesture based NDRAs recognition

This chapter is based on the published paper*: L. Yang et al., "A refined non-driving activity classification using a two-stream convolutional neural network," IEEE Sens. J., vol. 21, no. 14, pp. 1–1, 2020, doi: 10.1109/JSEN.2020.3005810.*

3.1 Introduction

Freely engaging in non-driving related activities (NDRAs) may be allowed in the future when the driver is driving a level 3 automated driving vehicle [1]. According to the definition of the SAE (J3016) Automation Levels [2], the driver should respond appropriately to the request to intervene. However, the engagement of NDRAs could reduce the driver's perceptual and cognitive capability on driving and situation awareness, which could result in a negative impact on the take-over response [3]. From the perspective of driving safety, Kim *et al*. [4] suggested when the take-over request is given by the vehicle, the driving performance after the take-over could be affected by the driver's age, gender and experience, but the status before the take-over might be more relevant. Although some approaches [4], [5] have been proposed in recent years to directly evaluate the driver's mental workload, the evaluated accuracy is not satisfactory due to the lack of convincible ground truth. The evaluation of the workload could be subjective and it is hard to be quantified. The further research results show that different types of NDRA and driving scenarios could cause different cognitive loads of the driver which affect the performance of the take-over quality and take-over time [6], [7]. For instance, visual related activities tended to take a longer reaction time than auditory related activities [8]. To achieve high-quality take-over and safety enhancement [9], it is therefore crucial to precisely identify, distinguish and track the type of NDRA that the driver is engaging in, then to evaluate the status and attention level or workload for the improvement of vehicle safety and operational efficiency. However, there is very limited literature focusing on that.

Analogous to NDRAs, secondary tasks as non-driving related tasks have been widely researched in human-driving in recent years. Li and Busso [10] claimed that secondary tasks can be recognised by evaluating the driver's mirror-

checking action. However, when the driver is doing NDRAs in an automated driving vehicle, the frequency of the mirror-checking will significantly decline. Therefore, this action is not considered an appropriate indicator for NDRA recognition. Jin *et al*. [11] proposed to recognise 6 secondary tasks (Bluetooth calls, cell phone calls, sending text messages, operating car-mounted players, chatting and singing) by combining both extracted eye movement and vehicle state characteristics. Martin *et al*. [12] presented a 3-stream recurrent neural network (RNN) system based on the driver's upper body pose. This system evaluates the transient skeleton movement, the spatial relationship of body parts and the knowledge about the vehicle interior to recognise 6 secondary tasks (drinking from a bottle, eating, using a phone for texting, making a call and reading a book). Xing *et al*. [13] collected both the colour and depth images of the driver's behaviour inside the vehicle cabin. Besides, the Kinect recorded the 3-D head rotation angles and the upper body joint position. A feedforward neural network (FFNN) was established to analyse the collected data and identify the secondary tasks. All these studies can recognise some kinds of secondary tasks like using a phone, operating the car-mounted player and chatting while driving manually. They presume that the primary task is driving which limits the diversity and continuity of the secondary tasks. These methods, therefore, cannot be directly applied for recognising NDRAs with high complexity and uncertainty. Yang *et al*. [1] proposed a dual-cameras based drive gaze mapping system that could be used to recognise some NDRAs with visual attention by mapping the gaze on the object that the driver is engaging in. However, such an object-based recognition approach can only identify that the driver is interacting with a phone but cannot recognise whether the driver is watching a movie (passive interaction) or playing a game (active interaction). The level of the driver's engagement in these activities in terms of perception and cognition is different according to the interaction mode, which leads to different performance after the take-over. The activities like reading or watching videos are considered passive-interaction activities since the driver intakes the information passively. But some like texting and playing games request a strong active interaction between the device and the driver. Consequently, the interaction mode could result in a different workload

of the driver [4], [8]. A further refinement of NDRAs classification in terms of object/device and task is therefore highly essential to design a more intelligent and efficient take-over process. This chapter proposes a novel region of interest (ROI) based 2-stream (visual scene and optical flow) convolutional neural network system to achieve this target through identifying both the device that the driver is engaging in and the task (e.g. *reading, playing a game, watch a movie, emailing* etc.) simultaneously.

3.2 Methodology

3.2.1 System Architecture

In the early stage of human action recognition, the human-object interaction has been widely researched, through the integration of object recognition, pose estimation and action identification [14], [15]. For the NDRAs recognition, the movement restriction and the body occultation enhance the difficulty of human pose estimation since the driver is sitting on the seat. Object detection methods can also be used to recognise some actions inside a vehicle such as hands-on-steering-wheel or using a phone [16]. Such methods recognise the human body parts and the object by sematic instance segmentation. With the development of multi-object detection, several CNN-based approaches have been proposed for action recognition in video. The achievements have been made from the perspective of the CNN framework or network design [17]–[19]. The evaluation of such existing researches is based on representative video datasets, such as HMDB-51 [20], UCF-101 [21], Kinetics [22]. These researches focus on the classification of actions with distinctive features like cutting in the kitchen, swing, archery etc. [21]. However, in this chapter, we focus on the classification of those phone-using and tablet-using NDRAs with high similarities. Such NDRAs happen inside of a vehicle and the driver is constrained on the seat. The spatial moving scale and intensity of activities are quite lower and harder to distinguish than the distinctive ones abovementioned. In this chapter, we propose that the classification process can be divided into 3 steps. In the first step, by extracting the ROI of the raw image captured by the camera, the interaction between the driver and the object can be limited to a region, which is helpful to reduce the

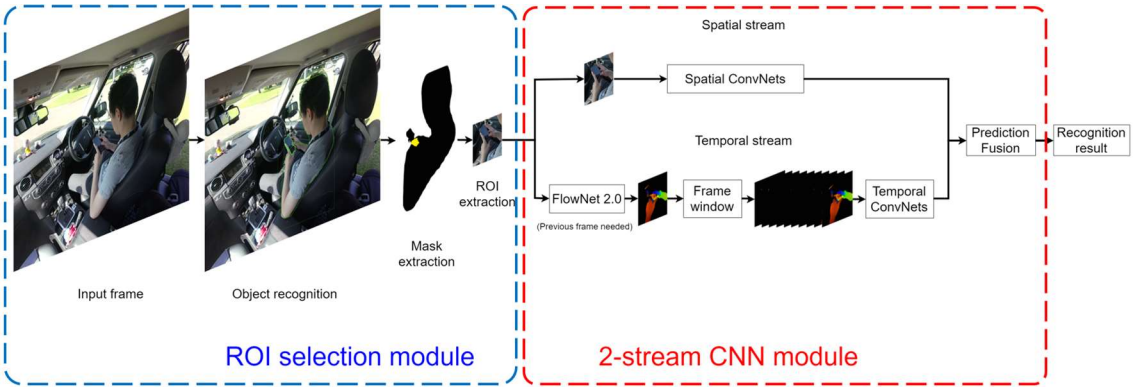


Figure 3-1 The proposed framework for NDRAs recognition that consists two parts: ROI selection module and 2-stream CNN module

noise and the processing time. The second step is to classify the object or device the driver is operating on. It relies on the analysis of the object's spatial information. The last step is to indicate how the driver interacts with the object based on pattern recognition. It is achieved by motion estimation. The last 2 steps can be run in parallel. The final result is given by fusing the 2 steps.

The flowchart of the system is illustrated in Figure 3-1, where the proposed system contains two modules: the ROI selection module and the 2-stream CNN module. The input frames are collected by a camera which is mounted on the roof of the vehicle to ensure that the object and hands are captured. The ROI module provides a region of human-object interaction (highlighted in Figure 3-1), which aims to significantly reduce the processing time and background noise for the 2-stream CNN module, and furtherly improve the classification accuracy. Then the detected ROI is fed into the 2-stream CNN module. The input of the spatial stream is from the RGB images and the input of the temporal stream is from a stack of optical flow frames which represent the motion between two adjacent frames within a certain time window. Then the prediction scores of the spatial and temporal streams will be fused to promote the final NDRA classification result.

3.2.2 ROI Selection

The raw RGB frames captured by the camera carry abundant information from both inside and outside of the vehicle. When we attempt to characterise and identify NDRAs, the most important parts are the object operated by the driver and the pattern of the driver's behaviour, especially the figures and hands. This

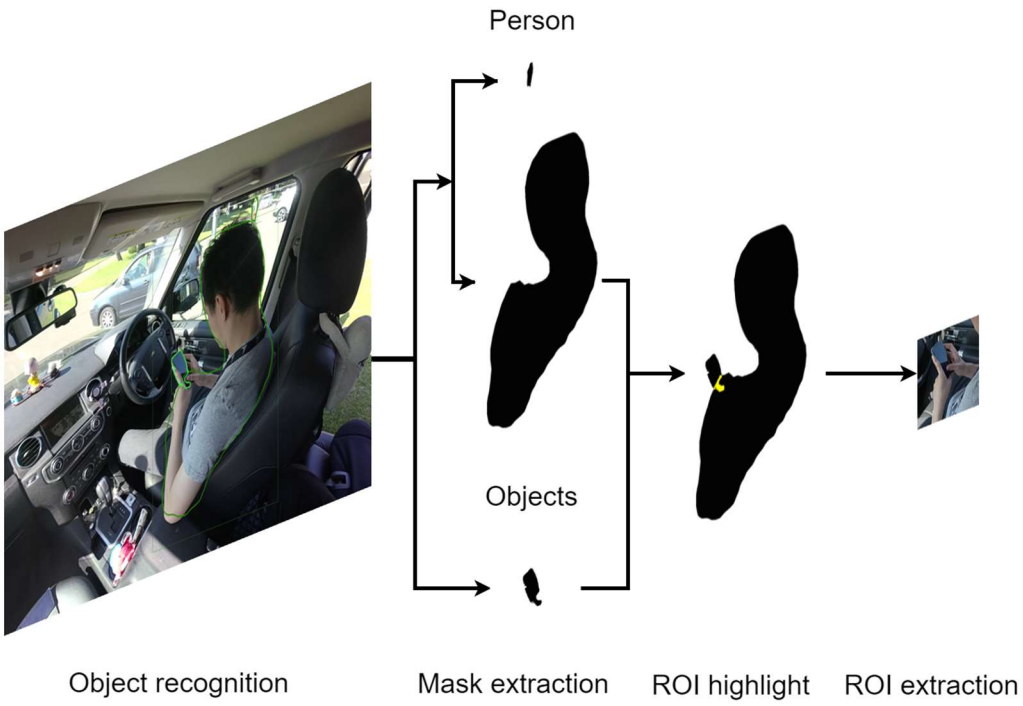


Figure 3-2 The flowchart of the ROI selection module

module aims to extract a region covering these parts from the raw frame due to two reasons. The first benefit is to help achieve real-time or near real-time performance. The size of the images fed into CNN should be small and informative. To keep the details of useful information, cropping the useless background is better than downsizing. The second benefit is to eliminate background noise. The scene change on the window during driving could introduce interference to pattern recognition. To achieve these aims, the raw frame is initially analysed by an object recognition algorithm, Mask R-CNN. It is a state-of-the-art object instance segmentation algorithm that could classify objects and localise them in pixels [23]. Comparing with the methods which can only provide a bounding box to localise the object, this algorithm offers a more accurate boundary as a mask on the recognised object, which is crucial to determine whether the driver is engaging in the object. The details are presented in Figure 3-2. In this module, Mask R-CNN is applied to recognise the driver and potential objects which could be involved in NDRAs, along with the masks. Then the ROI is selected based on the centre of the overlapping or connected area between human and object. The cropped frame will then be used as an input of

the 2-stream CNN module. If there is no ROI detected, the following module will not be activated, which suggests there is no related object or person in the scene, or the person and the object are recognised but the person is not interacting with it. For the estimation of optical flow, it is assumed that the location of the ROI within the time window does not change over time. If the object or driver is not detected or the ROI location difference between the last frame and current frame is smaller than a pre-set threshold, the current ROI will be the same as the ROI in the last frame. The ROI will only be updated if the location change exceeds the threshold. The threshold was set as 40 pixels in this study. The size of the ROI is customisable. In this case, the size was set as 320 × 320 pixels, where the raw image size is 1920 × 1440 pixels.

3.2.3 Optical Flow Estimation

Optical flow information has wide applications in studying vision-related tasks such as human pose estimation [24], video classification [25] and action recognition [26]. The rich motion information can be used to characterise the driver's behaviour between two adjacent frames. Compared to other optical flow estimation tools like DeepFlow [27] and Flow Fields [28], FlowNet 2.0 achieves the finest estimation performance. It provides the end-to-end optical flow estimation with convolutional networks [29]. The motion vector of each pixel is visualised by colour coding. The detail can be found in [30].

The processed optical flow frames for both raw and ROI frames are presented in Figure 3-3. The optical flow frame extracted from two adjacent raw frames includes the pixel motion from various moving sources, e.g., human, device, outside scene. We assume that the driver's behaviour associated with the device trajectory is the most important factor, particularly, the hand movement, to determine the task as detailed as possible. The obtained information from the optical flow frame can be categorised into 4 parts: scene change outside the window, body movement, device movement, and system noise, as marked in Figure 3-3. From the optical flow frame, a moving vehicle and a pedestrian outside the window can be observed and regarded as outside noise. There is also some system noise on the right side of the frame. All this information has no

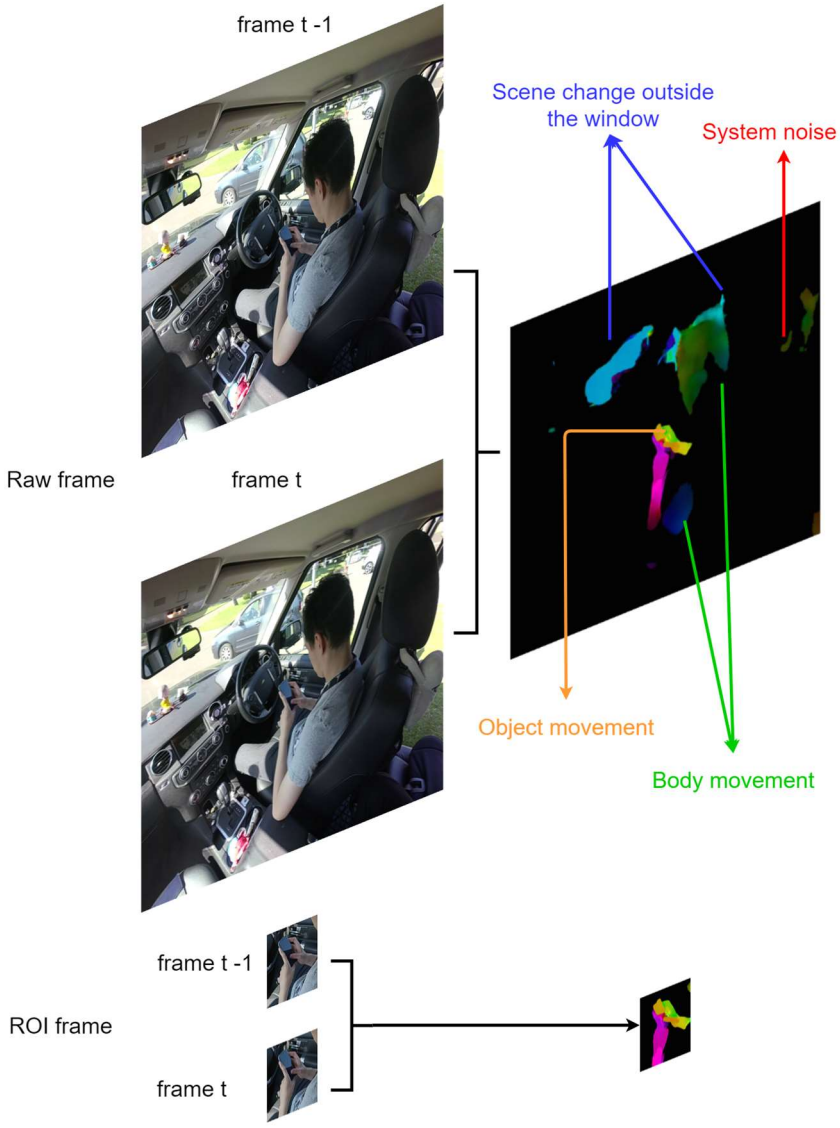


Figure 3-3 The comparison of the optical flow frame performance between raw frames and ROI frames

strong relevance to the pattern of the driver's behaviour. It can be considered as noises that could result in a negative effect on the performance of the temporal stream. It should be noted that although the driver's head and arm movement could be related to NDRAs it is relatively subjective and ignored in this study. In contrast, the optical flow of the ROI frames provides clear features related to the driver's hand and object movement. It is therefore used as one of the inputs for the 2-stream CNN module.

3.2.4 2-stream CNN

The challenge of action recognition in a still RGB image is that it cannot provide spatiotemporal features [18]. Particularly for NDRA recognition, common methods like pose estimation and scene recognition are not applicable. The driver is constrained on the seat and the only moving parts of the driver are the hands or head. The features extracted from the still image are not enough to differentiate most of NDRAs. In recent years, several CNN-based action recognition architectures have been proposed to improve the ability to capture the spatiotemporal features and increase the accuracy of the action recognition in videos, such as CNN with long short-term memory (LSTM) [31], 3D CNN [17], 2-stream CNN [19] and 2-stream 3D CNN [22]. The temporal stream of the 2-stream architecture offers the features of movement in the time domain and helps to identify the driver's behaviour. However, the state-of-the-art algorithm provided by the 3D CNN model in the 2-stream architecture requests large-scale datasets due to the complexity of the network [32]. Unlike the representative datasets mentioned above, the dataset used in this study is relatively small. One of the differences in data is that the features of the driver's behaviour are constrained in a small region. A complex network could increase the training burden and easily lead to an overfitting problem. Hence, a 2-stream architecture with 2D CNN model is proposed in this chapter. To achieve a better recognition performance, the CNN model in the 2-stream architecture is built based on the Residual Network (ResNet) due to its strong capability of training deeper networks [33].

The architecture of the CNN module is presented in Figure 3-4. The input of the spatial stream is a single ROI RGB frame at the current time and the input of the temporal stream is a stack of 10 optical flow frames (equals to 0.42s with a sample rate of 24 fps) on ROI calculated from 11 adjacent frames including the current frame. Traditionally, the input of the temporal stream is a stack of two-channel frames (two vectors). For an arbitrary pixel (u, v) in a single frame at the time t, the motion vector of this pixel is denoted as $(\overrightarrow{p_t^x(u,v)}, \overrightarrow{p_t^y(u,v)})$. The input for the temporal stream is denoted as $S_t(u, v, c)$, where c indicates the channel index. The corresponding input stack can be expressed as follow:

$$\begin{cases} S_t(u,v,2k-1) = \overrightarrow{p^x_{t-k+1}}(u,v) \\ S_t(u,v,2k) = \overrightarrow{p^y_{t-k+1}}(u,v) \end{cases} \quad \textbf{(3-1)}$$

where $u = [1,w], v = [1,h], k = [1,N]$, w and h are the width and height of the frame respectively, N denotes the number of the frame inside the stack.

In this study, we visualise the optical flow with colour coding. The vector field is then converted from two channels into three RGB channels. The input stack for the current frame t can then be expressed as follow:

$$\begin{cases} S_t(u,v,3k-2) = \overrightarrow{p^R_{t-k+1}}(u,v) \\ S_t(u,v,3k-1) = \overrightarrow{p^G_{t-k+1}}(u,v) \\ S_t(u,v,3k) = \overrightarrow{p^B_{t-k+1}}(u,v) \end{cases} \quad \textbf{(3-2)}$$

The number of optical flow frames in the stack, N, is configurable. It depends on how much historical information is required. Its performance will be addressed and discussed below.

ResNet-50 models are then built for both streams independently. There are 5 groups of convolution layers shown in Figure 3-4. In the convolutional layer 1,

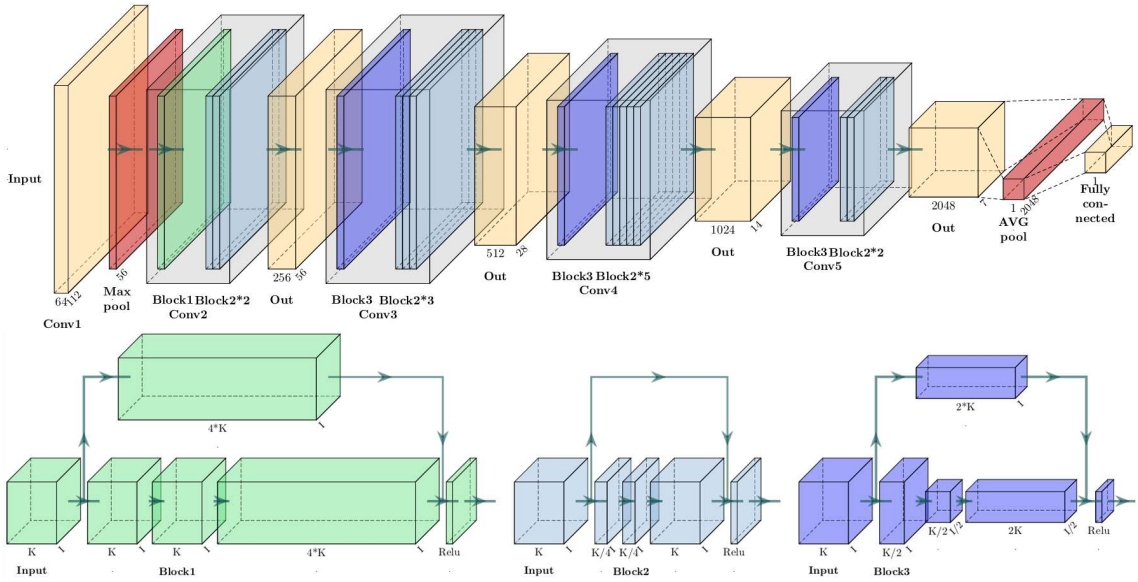


Figure 3-4 The architecture of ResNet 50 CNN. There are three types of convolutional blocks in this network, which are detailed in the bottom graph and indicated as different colours

both models extract 64 feature maps from the input. The difference between these 2 streams is the input, which is a 3-channel RGB image for the spatial stream or a 30-channel optical flow stack for the temporal stream. The last 4 convolution layer groups are made up of 3 types of residual block, which are shown at the bottom of Figure 3-4. The design of the shortcut structure in the block can be expressed as:

$$x_{l+1} = \mathrm{F}(x_l, \{W_i\}) + x_l \quad \textbf{(3-3)}$$

where x_l is the input of the layer l. $\mathrm{F}(x_l, \{W_i\})$ represents the function where the residual mapping is learned. Such residual structure alleviates the problem of exploding and vanishing gradient and usually achieves good performance in a deeper network [33].

The training process started with a pre-trained ResNet-50 model. The loss function used in training can be described as:

$$Loss(x, label) = -x[label] + \log\left(\sum_j e^{(x[j])}\right) \quad \textbf{(3-4)}$$

where x is the output that has been one hot encoded. $label$ is the true class. j is the index of the classes. The stochastic gradient descent (SGD) algorithm is used as an optimizer [34], which can be expressed as:

$$w_{n+1} = w_n - \gamma \nabla_w \mathrm{L}(z_n, w_n) \quad \textbf{(3-5)}$$

where n is the number of iteration. The gradient descent method focuses on the randomly picked mini-batch z_n. The loss L is minimised bases on the gradient of the weight vector w and the chosen gain γ. Furthermore, the learning rate is controlled in the training process. It starts with a high learning rate to accelerate the process and then reduces when the loss of the validation dataset stops improving.

After the training process, the trained model assesses the prediction scores of both streams. Finally, both scores are fused through a model expressed as:

Table 3-1 The NDRAs that drivers want to do in automated driver vehicle [35]

NDRAs	U.S.	China	India	Japan	U.K.	Australia
Read	14%	10.8%	11.1%	8.4%	9.9%	8.3%
Text or talk	12.7%	21.5%	16.3%	11.0%	7.1%	10.1%
Sleep	8.8%	11.2%	5.1%	18.9%	9.4%	9.0%
Watch movies	7.8%	1.7%	13.4%	9.2%	5.4%	7.3%
Work	6.2%	5.6%	17.7%	1.0%	6.4%	6.5%
Play games	2.6%	1.4%	2.3%	1.8%	2.5%	2.5%
Other	1.8%	0.7%	0.8%	0.3%	2.2%	1.3%

$$S_i = \frac{R_i}{\sum_{i=0}^{n-1}|R_i|} + \frac{O_i}{\sum_{i=0}^{n-1}|O_i|} \quad \textbf{(3-6)}$$

where S is the fusion score, R is the prediction score from the spatial CNN module, O is the prediction score from the temporal stream, i is the class index, and n is the number of NDRAs class.

3.2.5 Experiment Setup and Performance Validation

A Land Rover Discovery 4 was used as the test vehicle. The employed camera was the Garmin Virb Action Camera which was mounted on the roof of the vehicle between two front seats. The resolution of the camera was set as 1920 × 1440 pixels and images were sampled at 24 frames per second (fps). A PC with an Intel i7 9700k CPU, 32GB memory and an NVIDIA GeForce RTX 2080 GPU was employed for all deep learning related work.

During the experiment, the vehicle stayed stationary. A total of 10 participants (6 male and 4 female) were recruited for this experiment. The participants' age is in

Table 3-2 Categories for NDRAs recognition

Term	Browsing websites	Sending emails	Playing games	Reading	Watching videos
Phone	PB	PE	PG	PR	PV
Tablet	TB	TE	TG	TR	TV

a range from 22 to 26. They were requested to sit on the driving seat with the fastened seat belt and used the phone and tablet to conduct the selected activities one by one. Each activity lasted 1 minute. As shown in Table 3-1, Sivak and Schoettle [35] suggested that the common NDRAs are reading, texting, working, watching movies and playing games. From this survey, a total of 10 types of NDRA were identified and evaluated in this experiment, as presented in Table 3-2. The class of each activity is presented in 2 capital letters for the convenience of the result presentation. The first letter refers to the object (P and T stand for phone and tablet respectively), and the second letter refers to the task. For instance, PE refers to sending emails using a phone. Auditory guidance using Google Cloud Text-to-Speech was provided in this experiment to ensure consistency across all participants.

In this experiment, the participants need some time to follow the auditory guide for the NDRAs transition. Therefore, only the middle 40 seconds video was used for training, validation and testing. Each video has been split into 20 segments with a length of 2 seconds for each segment. There are 2000 segments in total for all participants and all NDRAs. From these segments, 64% of them was randomly selected for the training process, 16% of them was used for the validation process and 20% of them was used for the testing process. In the training process, 1 instance was randomly picked from each segment for both streams. The validation process was activated after each training epoch to adjust some hyperparameters like learning rate. 3 instances were randomly picked from each segment for both streams in this process. The testing process happened after the training process to evaluate the performance of the system. The following analysis is based on the results of the testing process

3.3 Results

3.3.1 Two Streams

An example of input frames for the 2-stream CNN module for each NDRA is presented in Figure 3-5, where the first column is the raw image with a full resolution, the second column is RGB images of the selected ROI as the spatial stream, and the remaining columns are the optical flow frames as the temporal

stream. From the RGB images of ROI, the difference can be observed between the phone-related activities and the tablet-related activities. The difference includes (a) the size of the object, (b) the distance between the object and the driver's body, and (c) the hand gesture. Therefore, the spatial stream should be able to differentiate the first 5 NDRAs and the last 5 NDRAs. However, this difference between the first 4 phone-related activities is dramatically dropped. It can be predicted that the classification accuracy for these 4 activities will be relatively low if only the spatial stream is applied. Furthermore, it can be seen that

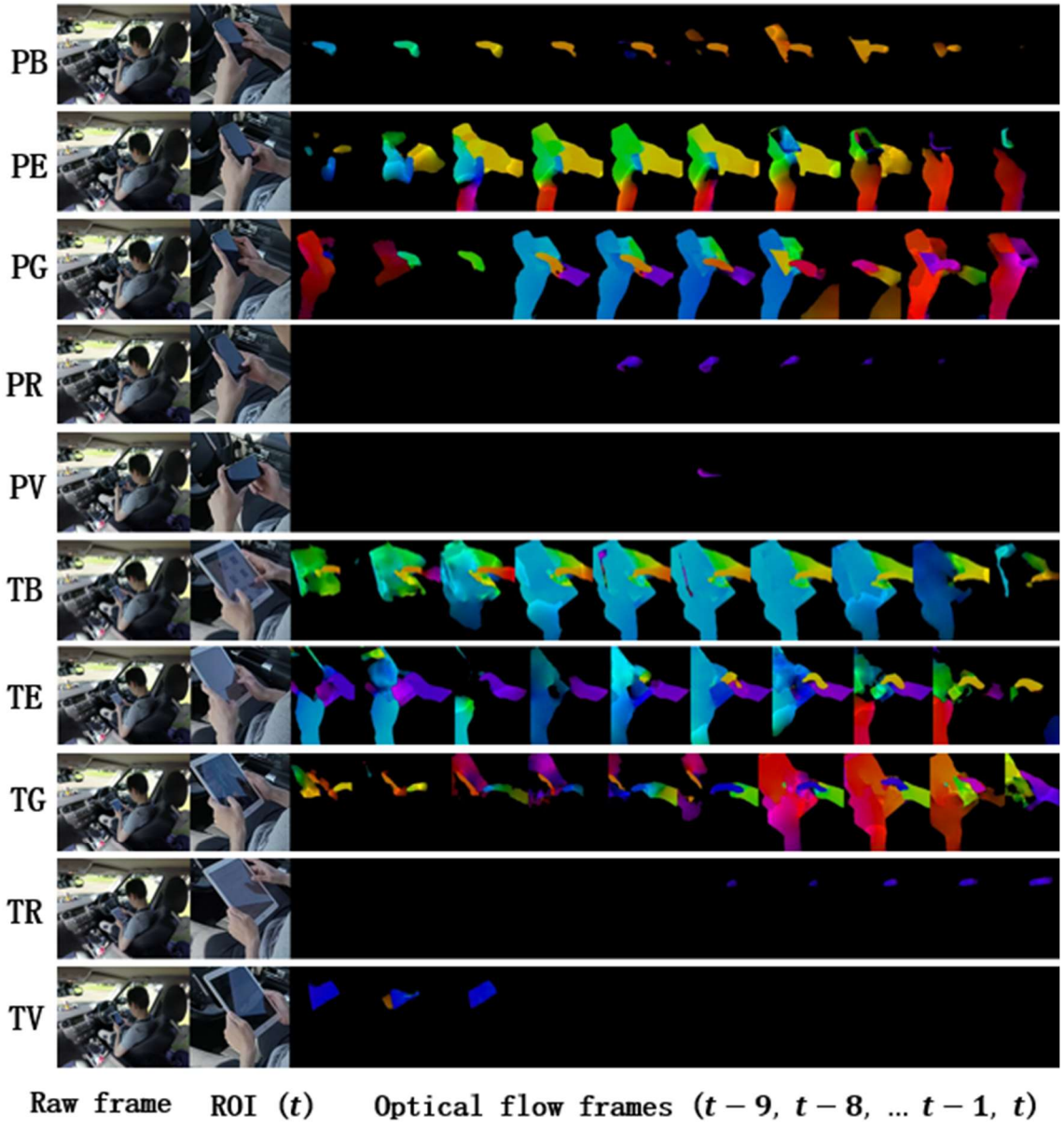


Figure 3-5 Examples of raw frame and input frames of 2-stream CNN module. There is some overlap between optical flow frames to fit the figure size

there is some reflection on the screen of the phone and the tablet. The change of illumination could affect the spatial information of the object while the driver is doing the same NDRA, which could furtherly bring a negative impact on the classification performance.

The optical flow frames contain more information on the driver's motion behaviour. It can be seen that activities like PB, PR, and TR involve one hand most of the time. Meanwhile, some activities like PE, PG, TB, TE, and TG need two hands for interaction. Another dimension of the difference between the two-hand related activities is the hands and fingers movement. For example, the different colour pattern between PE and PG suggests a different interaction mode with the device. The driver's behaviour on these NDRAs can be differentiated by the movement vectors of the hands and fingers which are represented by colours and their accumulation in the time domain. It also should be noticed that the optical flow stream is sensitive to the relatively high-frequency interaction for NDRAs like playing games, sending emails. For some other NDRAs like watching videos or reading, particularly with the tablet, the driver may stay with the same pose for a long time without any movement, as shown in TR and TV.

3.3.2 Classification Performance

The classification performance of the spatial stream only is presented in Figure 3-6. It can be found that phone-related activities can be easily distinguished from tablet-related activities, evidenced by zero error. However, for the classification among the phone-related activities or the tablet-related activities, the performance is not satisfactory. For PB, PE, PG and PR, the recall is lower than 50%, more than half of the true instance has not been recognised. TB and TR are difficult to be differentiated as well. This indicates that the spatial stream is not able to offer a persuasive NDRA classification for the same object. Besides, it can be observed that the value of both recall and precision for watching videos by phone (PV) and tablet (TV) are high, which suggests a reliable NDRA classification. The reason is that the way how participants interact with objects is quite special. When participants are conducting some activities like browsing websites or sending emails, they usually hold the phone or tablet vertically.

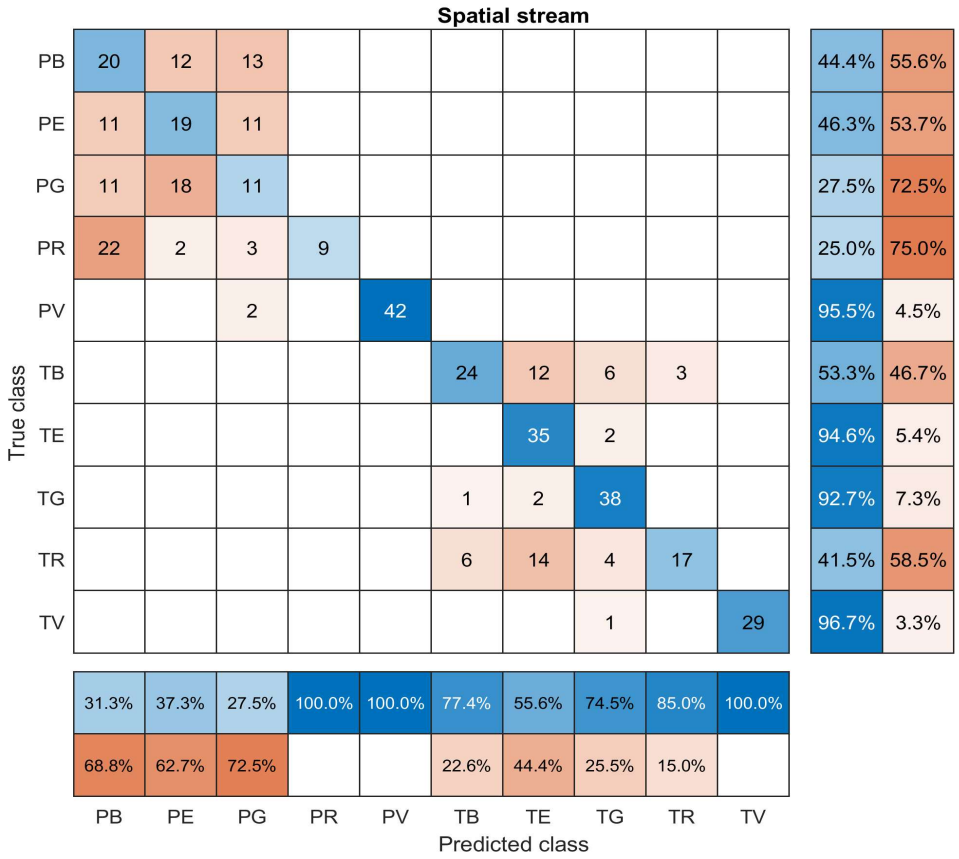


Figure 3-6 Confusion matrix of NDRAs recognition for the spatial stream. The precision and recall for each class are presented in the bottom and right of the figure, respectively, where the blue colour indicates the true value and the orange colour indicates the false value

However, for watching videos, most participants hold the phone or tablet horizontally. Comparing with the phone-related NDRAs, the tablet-related NDRAs classification shows a better performance in the spatial stream for both recall and precision. The content on the tablet's screen may have a contribution to the classification while that is not available for the phone, as shown in Figure 3-5.

Figure 3-7 presents the confusion matrix of the classification using the temporal stream only. The recall of most NDRAs is around 75%, except TR. Almost half of the true instance has been predicted as PR, which is because both NDRAs lack movement. The precision of most NDRAs is above 80%, while the precision of PR is only 38.6% Both recall and precision of sending emails are the highest (above 90%) no matter using a phone (PE) and tablet (TE). This is contributed by the special interaction mode in comparison with others.

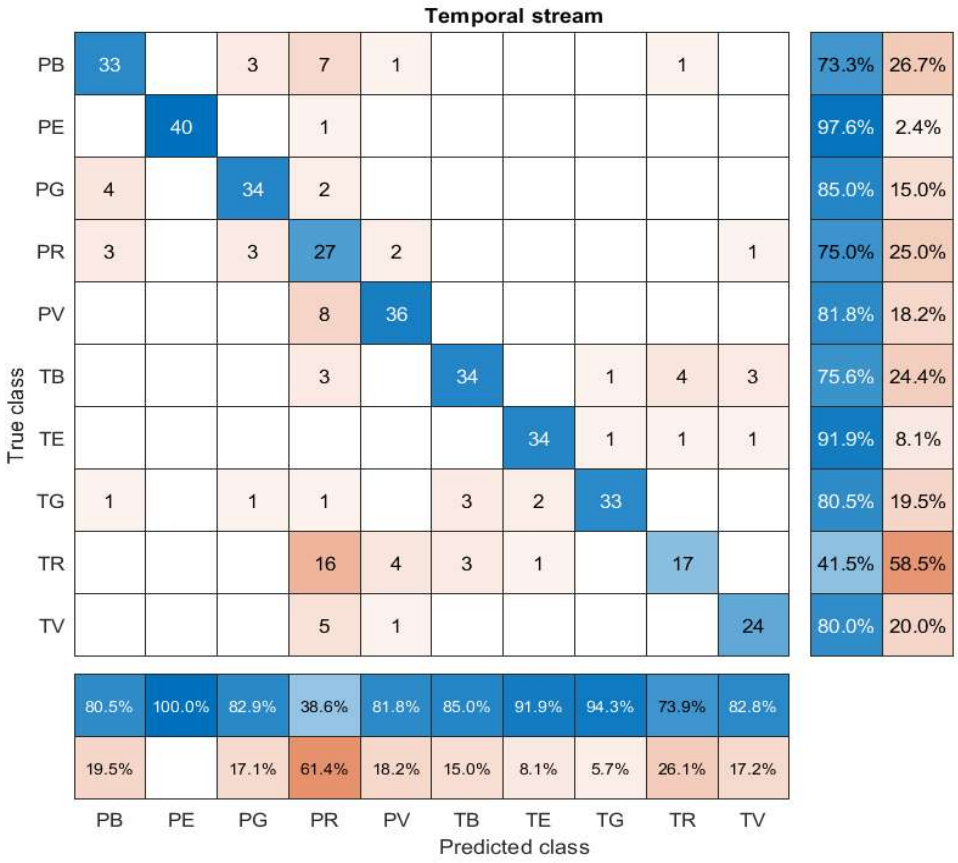


Figure 3-7 Confusion matrix of NDRAs recognition for the temporal stream

The fusion result of the proposed 2-stream approach is shown in Figure 3-8, which demonstrates a significant improvement for all NDRAs in contrast to the results of any single stream. The classification error among the NDRAs with the same object has been dramatically reduced. The overall accuracy is presented in Table 3-3. The overall accuracy has been improved from 61.0% (the spatial stream only) to 90.5%. Specifically, for the phone-related activities, the accuracy has been improved from 49.0% to 88.3%. For the tablet-related activities, the accuracy has been improved from 73.7% to 92.8%. In terms of the performance of a single stream, the temporal stream performs much better for phone-related activities. While for tablet-related activities, the performance is similar. The weighted F1 scores for all 3 terms are similar to the accuracy results. The top-3 error of the proposed method is only 0.5%. Specifically, for the spatial stream, the top-3 error is 10.5% while the weighted F1 value is only 60.6%. It suggests that the spatial stream could achieve a good performance on classifying the activities

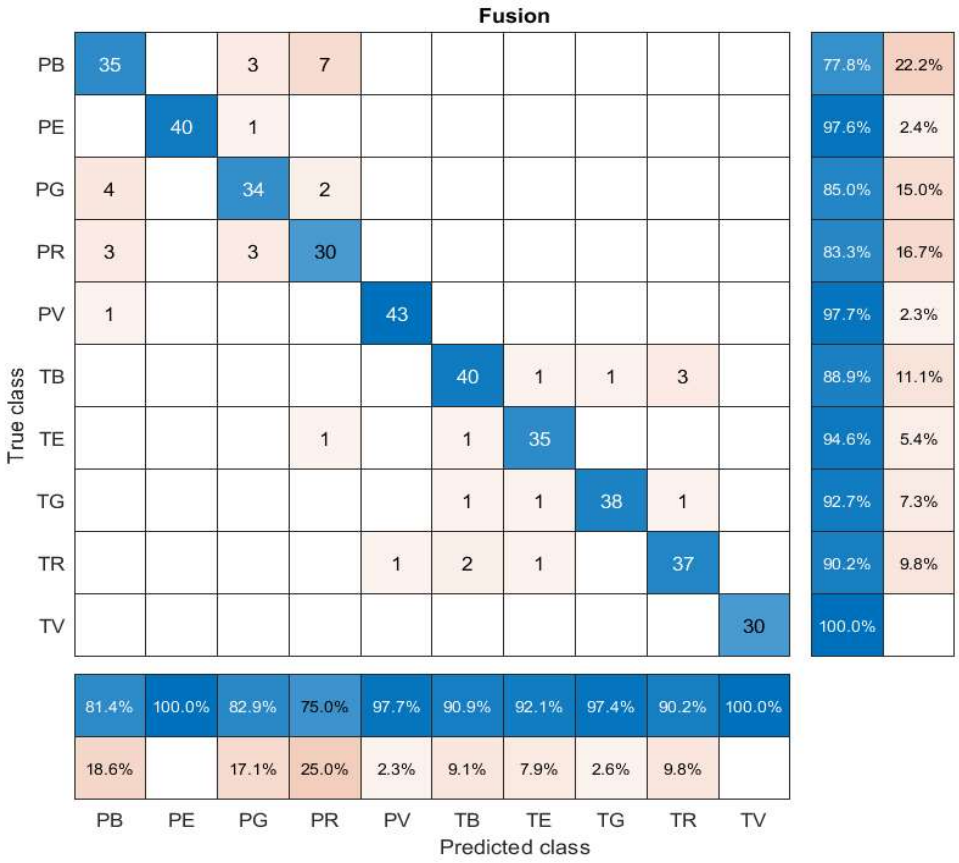


Figure 3-8 Confusion matrix of NDRAs recognition for the fusion of 2 streams

into some object-related groups, however, it cannot further classify the specific class from groups with the spatial information only.

Table 3-4 shows the overall performance when the ROI automatic selection is removed from the approach, which is similar to the work of [20]. It is suggested that the ROI automatic selection contributes almost 20% of accuracy. Furthermore, the performance of the spatial stream is especially sensitive to the

Table 3-3 Overall accuracy of NDRAs recognition

Term	Spatial stream	Temporal stream	Fusion
P accuracy	49.0%	82.5%	88.3%
T accuracy	73.7%	73.2%	92.8%
Accuracy	61.0%	78.0%	90.5%
Weighted F1	60.6%	78.7%	90.6%
Top-3 error	10.5%	4.3%	0.5%

Table 3-4 Overall accuracy of NDRA recognition without ROI selection

Term	Spatial stream	Temporal stream	Fusion
Accuracy	19.0%	66.2%	72.5%
Weighted F1	15.1%	66.4%	71.5%
Top-3 error	32.5%	10.2%	5.5%

ROI, where the accuracy drops from 61% to 19% in comparison to the temporal stream where the accuracy drops from 78% to 66%). This is probably because the spatial stream is easier to be interfered with by the complex driving environment.

3.3.3 Conflicted Cases Analysis

In this section, the details of conflicted cases are presented to further explain the reason why the fusion of two streams can help increase the accuracy of NDRA recognition. Figure 3-9 presents 3 cases where the fusion result is correct but the result from a single stream is not always right. It includes the “false-true-true case”, “true-false-true case” and “false-false-true case” for the spatial stream only, the temporal stream only and 2-steam respectively. The ground truth class is highlighted by a red block.

From the false-true-true case (the ground truth is PE), for the result of the spatial stream only, the scores of the first four classes are quite close. PB has the highest score that leads to a false result. However, both the temporal stream and 2-stream make the right decision. This is because the interaction mode of writing email is relatively unique from the others. For the true-false-true case (the ground truth is TR), with the help of the content extracted from the screen, the spatial stream achieves a true prediction although the scores of TB, TE and TR are similar. The prediction result of the optical flow is false due to the interference of PR and PV. This is because hand movement information in these activities is limited. After fusing these 2 streams, the prediction result is true. The bottom subfigure of Figure 3-9 presents the false-false-true case. Similar to the last case, the temporal stream cannot provide a true prediction due to the similarities

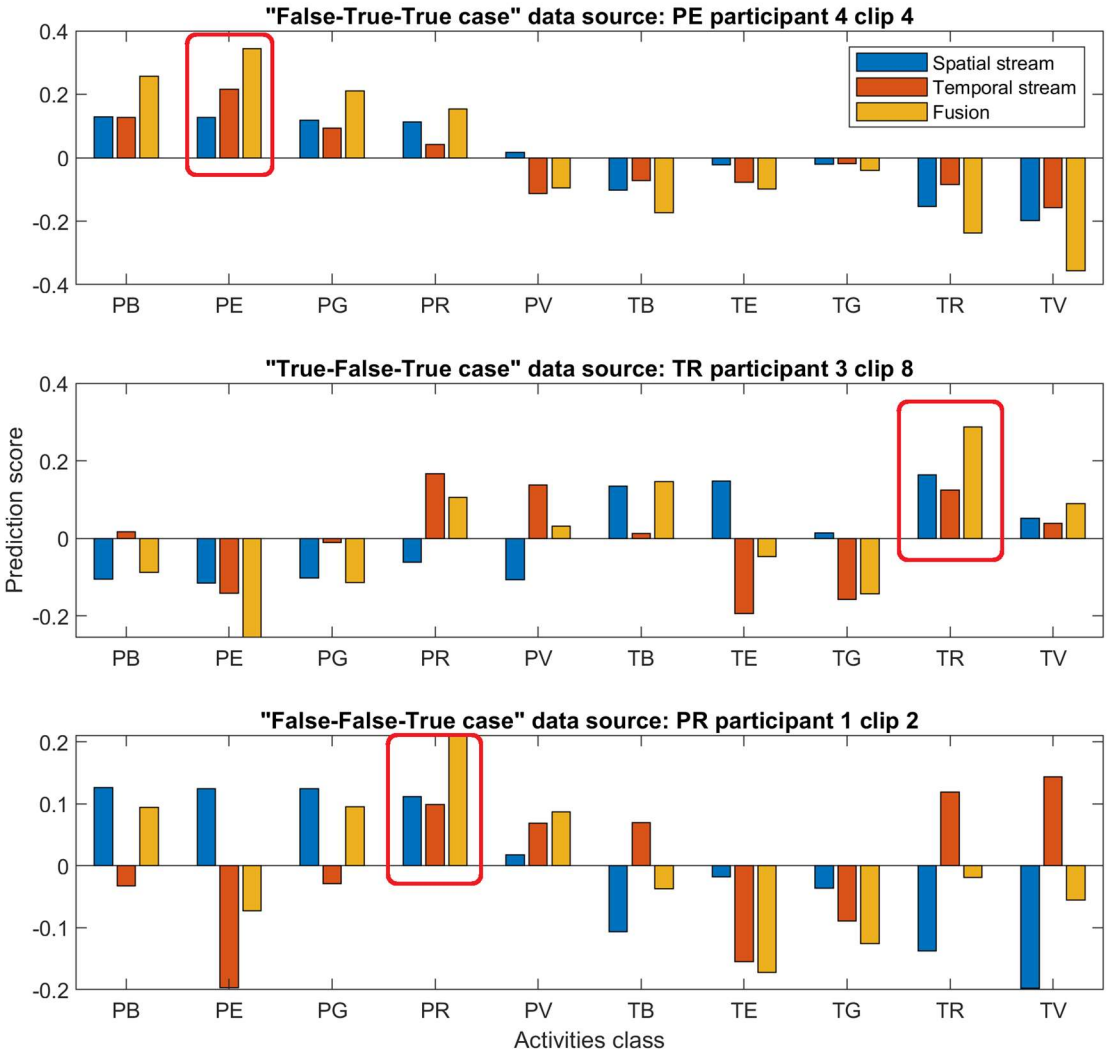


Figure 3-9 Prediction results for inference cases. The true class is highlighted by a red block

between TR and TV. It means that it is hard to differentiate reading and watching videos purely from the optical flow for the same reason above. Meanwhile, the spatial stream also suffers from the interference of PB, PE and PG. However, after combining the two streams, the score of PR is significantly higher than the others, which demonstrates the superiority of the proposed solution.

3.4 Discussion

For the proposed NDRA classification system, the performance could be affected by a few factors including the camera position and the number of frames for the temporal stream (N). A few other camera positions have been tested in the

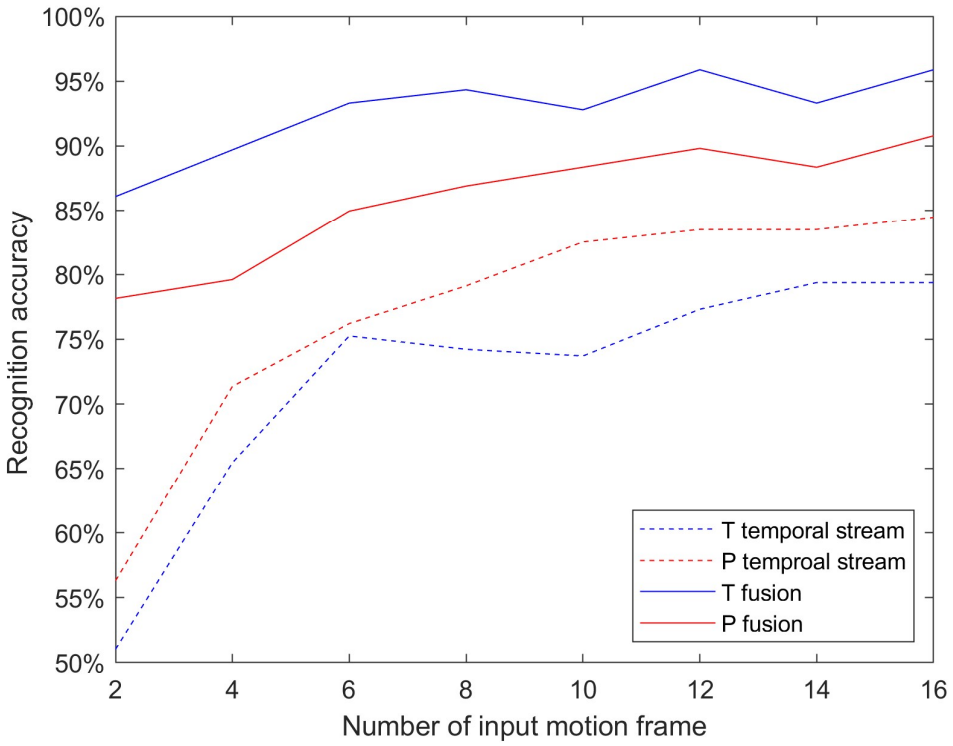


Figure 3-10 Impact of number of input motion frame on performance of temporal stream and fusion

experiment including the windscreen in front of the driver, the side window near the front passenger seat. On those positions, a clear view of the object and hands could not be obtained due to occultation caused by the human body or steering wheel. It is essential to recognise the driver and the object from the captured images. The selected camera position achieved the best performance of the tested positions. Although the side window is included, the ROI module can successfully remove this type of noise.

A stack of optical flow frames is regarded as the input of the temporal stream. The performance of the single temporal stream and 2-stream against the number of frames in the stack is presented in Figure 3-10, where P indicates the phone-related activities and T indicates the tablet-related activities. It can be observed that, in general, with the increment of N, the recognition accuracy increases due to the consideration of increasing temporal information. However, a larger number of frames also indicates that the system takes more time to determine

the type of NDRA, which is not helpful for real-time system deployment in the future. In this experiment, the number was set as 10 for the balance.

It should be noted that the analysis of this study is offline based and the real-time performance is not evaluated. From our point of view, it is not necessary and unlikely to output a decision for every frame because an activity usually is defined as a period of interaction. Using the mentioned PC, the average processing rate is 3.07, 16.38 and 126.17 fps for ROI selection, optical flow estimation and two-stream CNN activity recognition, respectively. It is our notion that the system can update the outcome every 1 second. Furthermore, the experiments were conducted on a stationary vehicle. There will be some challenges to deploy it to a driving vehicle. For example, camera vibration could introduce noise to the optical flow estimation. As a computer-vision approach, the rapid variation of illumination will also introduce extra noise for object recognition.

3.5 Conclusion

This chapter proposed a single-camera-based NDRA classification method using a 2-stream CNN benefiting from both spatial and temporal information of an automatically selected ROI. The spatial stream extracts the spatial features of the driver and the engaged object, and the temporal stream characterises the pattern of the interaction behaviour. With this method, different tasks with the same object can be differentiated. The key findings of this study are listed below.

1. The spatial stream achieves good performance in the action recognition dataset like UCF-101, HMDB-51, since the scenario of each action category is quite different. However, for the fine recognition of NDRA in this study, this stream is not sufficient.

2. The content of the tablet screen can help increase the classification accuracy in the spatial stream. However, this is not applicable for small-size objects like phones due to reflection.

3. The temporal stream shows good performance on NDRAs involving high-frequency interaction like sending emails or playing games, but low performance on NDRAs with very limited interaction such as watching videos or reading.

4. For the conducted experiments, the accuracy of NDRA recognition was improved from 61% using the spatial stream and 78% using the temporal stream to 90.5% using the two streams.

5. The inclusion of the ROI automatic selection improves the overall performance from 72.5% to 90.5%.

It should be noted that the proposed system can only be applied to NDRAs required physical interaction with the device or object, such as drinking, playing an instrument. A further study is required to tackle other NDRAs such as listening to music where other sensors are required.

3.6 Reference

[1] L. Yang, K. Dong, A. J. Dmitruk, J. Brighton, and Y. Zhao, “A Dual-Cameras-Based Driver Gaze Mapping System With an Application on Non-Driving Activities Monitoring,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 10, pp. 4318–4327, Oct. 2020, doi: 10.1109/TITS.2019.2939676.

[2] B. W. Smith, “SAE levels of driving automation,” *Cent. Internet Soc. Stanford Law Sch.*, p. 1, 2014.

[3] T. Ersal, H. J. A. Fuller, O. Tsimhoni, J. L. Stein, and H. K. Fathy, “Model-Based Analysis and Classification of Driver Distraction Under Secondary Tasks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 692–701, Sep. 2010, doi: 10.1109/TITS.2010.2049741.

[4] J. Kim, W. Kim, H.-S. Kim, and D. Yoon, “Effectiveness of Subjective Measurement of Drivers’ Status in Automated Driving,” in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, Aug. 2018, vol. 2018-Augus, pp. 1–2, doi: 10.1109/VTCFall.2018.8690557.

[5] M. Bueno, E. Dogan, F. Hadj Selem, E. Monacelli, S. Boverie, and A. Guillaume, “How different mental workload levels affect the take-over control after automated driving,” in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov. 2016, pp. 2040–2045, doi: 10.1109/ITSC.2016.7795886.

[6] S. H. Yoon, Y. W. Kim, and Y. G. Ji, “The effects of takeover request modalities on highly automated car control transitions,” *Accid. Anal. Prev.*, vol. 123, pp. 150–158, Feb. 2019, doi: 10.1016/j.aap.2018.11.018.

[7] K. Zeeb, A. Buchner, and M. Schrauf, “Is take-over time all that matters? The impact of visual-cognitive load on driver take-over quality after conditionally automated driving,” *Accid. Anal. Prev.*, vol. 92, pp. 230–239, Jul. 2016, doi: 10.1016/j.aap.2016.04.002.

[8] B. Wandtner, G. Schmidt, N. Schömig, and W. Kunde, “Non-driving related tasks in highly automated driving - Effects of task modalities and cognitive workload on take-over performance,” *AmE 2018 - Automot. meets Electron. 9th GMM-Symposium*, pp. 1–6, 2018.

[9] C. Wu, H. Wu, N. Lyu, and M. Zheng, “Take-Over Performance and Safety Analysis Under Different Scenarios and Secondary Tasks in Conditionally Automated Driving,” *IEEE Access*, vol. 7, pp. 136924–136933, 2019, doi: 10.1109/ACCESS.2019.2914864.

[10] N. Li and C. Busso, “Detecting Drivers’ Mirror-Checking Actions and Its Application to Maneuver and Secondary Task Recognition,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 980–992, Apr. 2016, doi: 10.1109/TITS.2015.2493451.

[11] L. Jin, B. Guo, Y. Jiang, F. Wang, X. Xie, and M. Gao, “Study on the Impact Degrees of Several Driving Behaviors When Driving While Performing Secondary Tasks,” *IEEE Access*, vol. 6, pp. 65772–65782, 2018, doi: 10.1109/ACCESS.2018.2878150.

[12] M. Martin, J. Popp, M. Anneken, M. Voit, and R. Stiefelhagen, “Body Pose and Context Information for Driver Secondary Task Detection,” in *2018 IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2018, vol. 2018-June, no. Iv, pp. 2015–2021, doi: 10.1109/IVS.2018.8500523.

[13] Y. Xing *et al.*, “Identification and Analysis of Driver Postures for In-Vehicle Driving Activities and Secondary Tasks Recognition,” *IEEE Trans. Comput.*

Soc. Syst., vol. 5, no. 1, pp. 95–108, Mar. 2018, doi: 10.1109/TCSS.2017.2766884.

[14] Bangpeng Yao and Li Fei-Fei, “Recognizing Human-Object Interactions in Still Images by Modeling the Mutual Context of Objects and Human Poses,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1691–1703, Sep. 2012, doi: 10.1109/TPAMI.2012.67.

[15] M. Ziaeefard and R. Bergevin, “Semantic human activity recognition: A literature review,” *Pattern Recognit.*, vol. 48, no. 8, pp. 2329–2345, Aug. 2015, doi: 10.1016/j.patcog.2015.03.006.

[16] T. H. N. Le, C. Zhu, Y. Zheng, K. Luu, and M. Savvides, “DeepSafeDrive: A grammar-aware driver parsing approach to Driver Behavioral Situational Awareness (DB-SAW),” *Pattern Recognit.*, vol. 66, no. December 2016, pp. 229–238, Jun. 2017, doi: 10.1016/j.patcog.2016.11.028.

[17] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning Spatiotemporal Features with 3D Convolutional Networks,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, vol. 2015 Inter, pp. 4489–4497, doi: 10.1109/ICCV.2015.510.

[18] S. R. Sreela and S. M. Idicula, “Action Recognition in Still Images using Residual Neural Network Features,” *Procedia Comput. Sci.*, vol. 143, pp. 563–569, 2018, doi: 10.1016/j.procs.2018.10.432.

[19] K. Simonyan and A. Zisserman, “Two-Stream Convolutional Networks for Action Recognition in Videos,” *Biochem. Pharmacol.*, vol. 32, no. 5, pp. 849–855, Jun. 2014, doi: 10.1016/0006-2952(83)90587-7.

[20] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, “HMDB: A large video database for human motion recognition,” in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 2556–2563, doi: 10.1109/ICCV.2011.6126543.

[21] K. Soomro, A. R. Zamir, and M. Shah, “UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild,” no. November, Dec. 2012,

[Online]. Available: http://arxiv.org/abs/1212.0402.

[22] J. Carreira and A. Zisserman, “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, vol. 2017-Janua, pp. 4724–4733, doi: 10.1109/CVPR.2017.502.

[23] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.322.

[24] T. Pfister, J. Charles, and A. Zisserman, “Flowing convnets for human pose estimation in videos,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2015 Inter, pp. 1913–1921, 2015, doi: 10.1109/ICCV.2015.222.

[25] Joe Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, “Beyond short snippets: Deep networks for video classification,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, vol. 07-12-June, pp. 4694–4702, doi: 10.1109/CVPR.2015.7299101.

[26] L. Sevilla-Lara, Y. Liao, F. Güney, V. Jampani, A. Geiger, and M. J. Black, “On the Integration of Optical Flow and Action Recognition,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11269 LNCS, 2019, pp. 281–297.

[27] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, “DeepFlow: Large Displacement Optical Flow with Deep Matching,” in *2013 IEEE International Conference on Computer Vision*, Dec. 2013, no. Section 2, pp. 1385–1392, doi: 10.1109/ICCV.2013.175.

[28] C. Bailer, B. Taetz, and D. Stricker, “Flow Fields: Dense Correspondence Fields for Highly Accurate Large Displacement Optical Flow Estimation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1879–1892, Aug. 2019, doi: 10.1109/TPAMI.2018.2859970.

[29] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, “FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, vol. 2017-Janua, pp. 1647–1655, doi: 10.1109/CVPR.2017.179.

[30] G. Chantas, T. Gkamas, and C. Nikou, “Variational-Bayes Optical Flow,” *J. Math. Imaging Vis.*, vol. 50, no. 3, pp. 199–213, Nov. 2014, doi: 10.1007/s10851-014-0494-3.

[31] J. Donahue *et al.*, “Long-Term Recurrent Convolutional Networks for Visual Recognition and Description,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 677–691, Apr. 2017, doi: 10.1109/TPAMI.2016.2599174.

[32] K. Hara, H. Kataoka, and Y. Satoh, “Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet?,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6546–6555, doi: 10.1109/CVPR.2018.00685.

[33] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[34] L. Bottou, “Large-Scale Machine Learning with Stochastic Gradient Descent,” in *Proceedings of COMPSTAT’2010*, Heidelberg: Physica-Verlag HD, 2010, pp. 177–186.

[35] M. Sivak and B. Schoettle, “Motion Sickness in Self-Driving Vehicles,” no. April, 2015, [Online]. Available: https://deepblue.lib.umich.edu/handle/2027.42/111747.

4 Dual-stream 3D residual network for spatio-temporal representations learning

This chapter is based on the published paper*: L. Yang, X. Shan, C. Lv, J. Brighton, and Y. Zhao, "Learning spatio-temporal representations with a dual-stream 3D residual network for non-driving activity recognition," IEEE Trans. Ind. Electron., vol. 0046, no. c, pp. 1–1, 2021, doi: 10.1109/TIE.2021.3099254.*

4.1 Introduction

More and more level 3 automated driving vehicles will be on road in the coming years [1], and such vehicles allow drivers to take their hands and eyes off the road. However, according to SAE (J3016) Automation Levels, in level 3, drivers are still expected to take control of the vehicle if there is a request to intervene [2]. The driver's situation awareness in terms of driving environment and vehicle condition is reduced since they do not need to pay full attention to road and dashboard, which could bring a risk when the driver takes over the vehicle control without the right process in place. Therefore, it is of great importance to monitor the driver's behaviour during the level 3 automated driving and design the specific takeover request modality or Human Machine Interface (HMI) for different states to ensure a smooth and safe control transition [3].

There are two kinds of activities that the driver could engage in inside the vehicle cabin, which are driving related activities (DRAs) and non-driving related activities (NDRAs). Similar to distraction and fatigue, the engagement of NDRAs could reduce the driver's situation awareness. Normally, the methods of detecting NDRAs engagement is based on the driver's attention [4]. Since the drivers always check the road or surrounding environment when they are conducting DRAs, while during NDRAs engagement, they pay more attention to the object they are engaging with. Moreover, different NDRAs could lead to different impacts on the driver's take-over performance [5]–[7]. A refined classification of NDRAs could help to design an intelligent take-over process to improve driving safety. During NDRAs engagement, the driver's hand movement contains information about the interaction between the driver and the object, which can be used for

further classification. Therefore, both visual attention and behaviour are necessary for the recognition of the driver's activity in the vehicle.

The recognition of the driver's NDRAs has been widely researched in the last few years. With the rapid development of deep learning in activity recognition based on videos, computer vision-based methods have become the focus for NDRAs recognition [3], [8], [9]. The methods for action recognition using videos can be roughly divided into two categories: spatio-temporal attention mechanisms and 3D convolutional neural network (CNN). Both methods employ CNN for spatial feature extraction due to its great learning capability in the spatial domain. The spatio-temporal attention mechanisms learn the temporal features by employing the sequence-based signal processing methods like Recurrent Neural Network, Long Short-Term Memory and transformer [10], [11]. 3D CNN extends the 2D spatial features into 3D features by adding a convolutional kernel in the temporal domain [12]–[15]. For the NDRAs recognition, unlike the traditional activities in the action recognition dataset [16], [17], such as Tai Chi, Basketball, Diving, etc., which contains diverse spatial information in the background and large-scale body movement, NDRAs are constrained in the vehicle cabin. Normally, the movement that matters is the driver's hand. The hand movement is more complex in the temporal domain and the background is similar in the spatial domain, which poses a challenge to the existing 3D CNN models [12], [18] for activity recognition. Considering that, a proper design of the 3D CNN model could enhance the spatio-temporal representations of the activity with less 3D convolutional computation to achieve good recognition performance. Furthermore, the driver's head movement is also needed to be evaluated, since the driver visual attention is also a key factor to determine the NDRAs engagement. In this chapter, we propose a 2-feed 3D CNN based driver behaviour recognition system. This system focuses on both driver's head and hand movement to recognise whether the driver is engaging with an NDRA or not, and further determine the type of NDRA or DRA. We design a dual-stream 3D residual network, named DS3D ResNet, to enhance the short-time spatial representation and small-region temporal representation learned on separate streams. A novel NDRA dataset has been produced to evaluate the proposed

model and other state-of-the-art models. This study also visualises the hidden layers of the proposed model to further verify and explain the semantic features that the model learned.

4.2 Related work

NDRAs recognition: The methods of activity recognition can be roughly divided into 2 categories from the perspective of feature extraction, which are hand-crafted features based methods and deep learning-based methods. The first kind of method classifies the activities based on some hand-crafted features like driver's gaze direction, hand movement and body pose. Martin *et al*. [19] extracted features of the driver's upper body pose and proposed a 3-stream recurrent neural network (RNN) system. This system evaluates the spatial relationship of body joints, the temporal skeleton movement and the context of the driver's surroundings to recognise the selected NDRAs, including drinking, phone texting, calling, reading and eating. Furthermore, Xing *et al*. [20] combined the depth information inside the vehicle cabin with the features mentioned above and established a feedforward neural network (FFNN) to identify the activities. Yang *et al*. [4] proposed a dual-camera gaze estimation system and addressed the NDRA recognition problem from the perspective of the driver's eye. With the development of CNN in the field of activity recognition [12], [13], [21]–[23], deep learning-based methods have attracted increasing attention in NDRA recognition in recent years. Xing *et al*. [8] removed the image background and used the drivers' body as the input of the CNN model to recognise their behaviours. Yang *et al.* [3] employed a 2-stream CNN model to extract the spatial features from the original image and the driver's hand movement features from the corresponding optical flow images. Moreover, Eraqi *et al*. [24] trained different CNNs on multiple inputs including raw images, skin-segmented images, face images, hands images, and "face+hands" images. The final prediction is obtained by using a genetic algorithm based on the outputs of all the CNN models.

3D CNN: CNN has been widely researched in recent years and made great achievements on spatial representation, particularly in the scope of computer vision. CNN has been mainly applied to 2D images that lack temporal

representation, which is especially crucial for the application of video classification. To address this challenge, 3D CNN was employed to learn the spatio-temporal representations and extract the motion information hidden in the video frames [12], [25]. The residual structure [26] was implemented to tackle the training difficulty in the deeper 3D CNN model [18]. Since the computation cost of the deep 3D CNN is expensive and the model size is relatively large, Qiu *et al*. [27] proposed a Pseudo-3D network to factorise 3D convolutions into spatial convolutions and temporal convolutions to reduce the computational complexity. They compacted the model with 3 different forms of the spatio-temporal residual blocks. Similarly, Tran *et al*. [28] used only a spatial convolution followed by a temporal convolution residual block in the proposed R(2+1)D network and achieved better action recognition performance.

Unlike other deep learning-based methods for NDRA recognition, which mainly focus on the 2D image domain, our work attempts to extract the spatio-temporal features from the driver behaviour in the video domain. Considering the characterisation of NDRAs, the capability of 3D CNN has not been fully exploited with the existing architecture mentioned above. In this work, we improve the spatial-temporal representation of residual blocks in the network with a designed dual-stream structure by enhancing the small-region temporal representation and the short-time spatial representation in different scales. The idea of this work is not only to revise the network structure but also to develop a framework to recognise and classify the type of NDRA engagement during level 3 automated driving. The proposed framework is given in detail in the next section.

4.3 Methodology

The proposed 2-feed dual-stream 3D residual network-based driver activity recognition framework is illustrated in Figure 4-1. There are 2 feeds in this framework, which are the frames from the front camera and the rear camera. The front camera captures the driver's head movement and estimates the visual

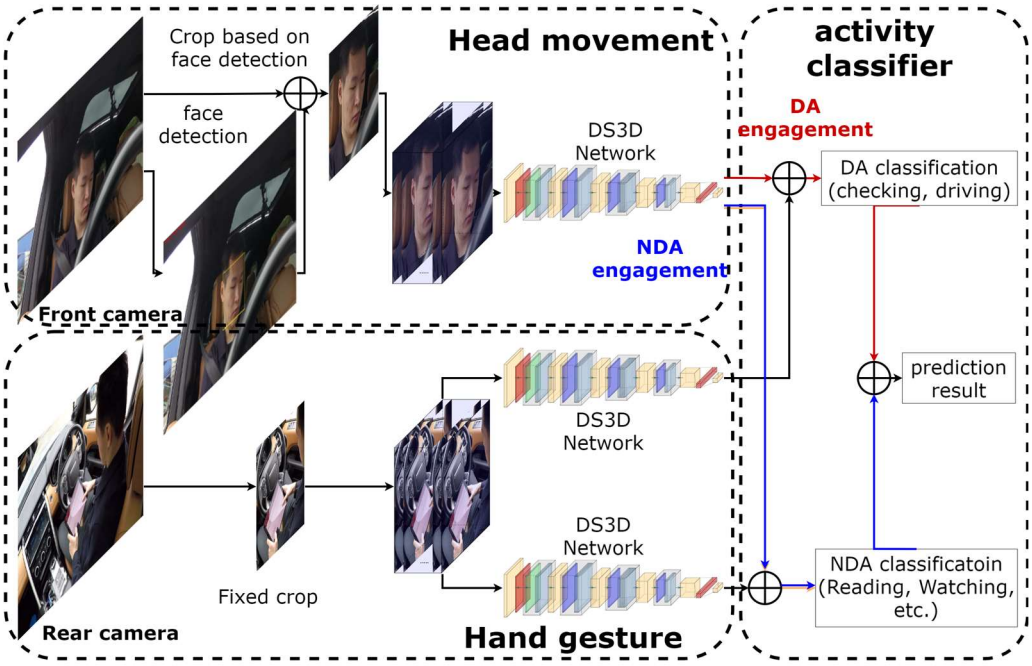


Figure 4-1 Two-feed driver activity recognition framework. The head movement module estimates the driver's visual attention and the hand gesture module captures the driver's hand behaviour. Activity classifier module fuses these two feeds to classify the NDRA or DRA.

attention, which is used to recognise whether the driver is engaging with NDRAs or not. The input of the 3D CNN model for this feed a stack of frames, which are cropped based on the location of the detected face from raw frames. The rear camera focuses on the driver's behaviour in the cabin mainly the hand movement, which aims to further classify the specific NDRAs or DRAs. The final activity classification is obtained by combining these two results.

4.3.1 3D Residual Block

3D convolution is the most natural method to extract the spatio-temporal features from videos [12], [27]. It has the capability to model the temporal connection among the spatial information encoded frames. For the 3D convolution, the filter is denoted as $d \times k \times k$, where d and k are the temporal depth and the spatial size of the filter respectively.

Following the success of the Residual Networks (ResNets) in encoding the spatio-temporal information for action recognition task [18], [26]. We propose 2

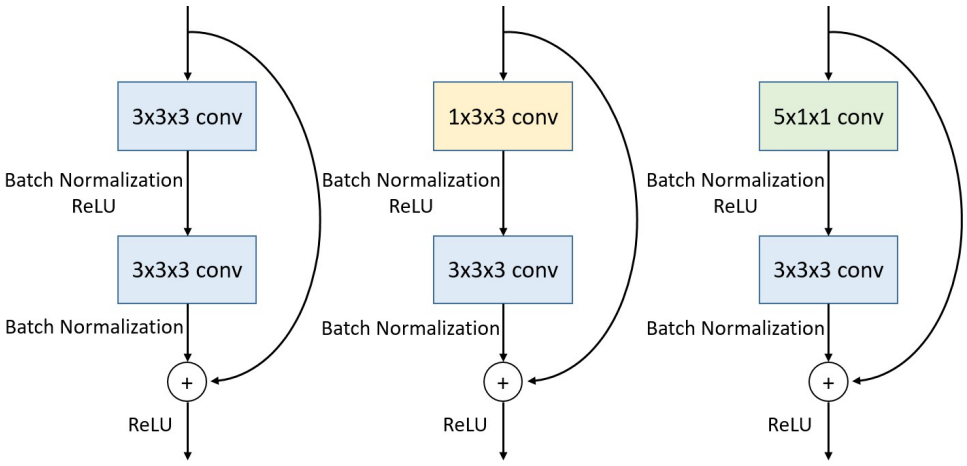


Figure 4-2 The basic residual block and the proposed blocks

different residual blocks to enhance the short-time spatial representation and the small-region temporal representation of the model, as illustrated in Figure 4-2 (b) and (c), based on the basic residual block in Figure 4-2 (a). There are 2 convolutional layers in a basic residual block. Each layer is followed by batch normalization [29]. The filter size of each convolutional layer is $3 \times 3 \times 3$.

The output of the l-th residual block can be expressed as:

$$x_{l+1} = F(x_l, \{W_i\}) + x_l \tag{4-1}$$

where x_{l+1} and x_l are the output and input of the block. The function $F(x_l, \{W_i\})$ is the learned residual mapping of the block and weight $\{W_i\}$ is for multiple convolutional layers.

The short-time spatial block (see Figure 4-2 (b)) aims to encode the change of spatial information in a short time. Unlike the basic residual block, the size of the filter S used in the first convolutional layer of the proposed block is $1 \times 3 \times 3$. This filter compresses the temporal dimension, which is equivalent to the 2D convolutional filter on the spatial domain. The filter R of the second convolutional layer is still a $3 \times 3 \times 3$ filter to expand the receptive field in both temporal and spatial domains. The block can be expressed as:

$$x_{l+1} = R(S(x_l, W_s), W_r) + x_l \tag{4-2}$$

The small-region temporal block, shown in Figure 4-2 (c), concentrates on a small area and captures its change over a long period. The size of the first convolutional filter (T) is $5 \times 1 \times 1$, which can be considered as a 1D convolutional filter on the temporal domain. It is followed by a $3 \times 3 \times 3$ filter (R). The output of this block can be expressed as:

$$x_{l+1} = R(T(x_l, W_t), W_r) + x_l \tag{4-3}$$

The ReLU activation function is employed after the first convolutional layer and the output of all these blocks.

4.3.2 Architecture of the 3D CNN Model

The architecture of the network is illustrated in Figure 4-3. For simplicity, the size of the given video clip is denoted as $c \times l \times h \times w$, where c is the number of channels, l is the number of frames in the clip, h and w are the height and width of images, respectively. The input of the network is a $3 \times 16 \times 112 \times 112$ tensor. The parallel structure is employed after the first convolution block. The upper spatial stream uses a sequence of 4 spatial blocks to emphasise the short-time spatial information in different scales. The bottom temporal stream has 4 temporal blocks connected in series, which focus on the change in the small-region temporal domain. After pooling, the size of the feature map for each stream is $256 \times 1 \times 1 \times 1$. The final 512-dimensional vector is obtained by concatenating

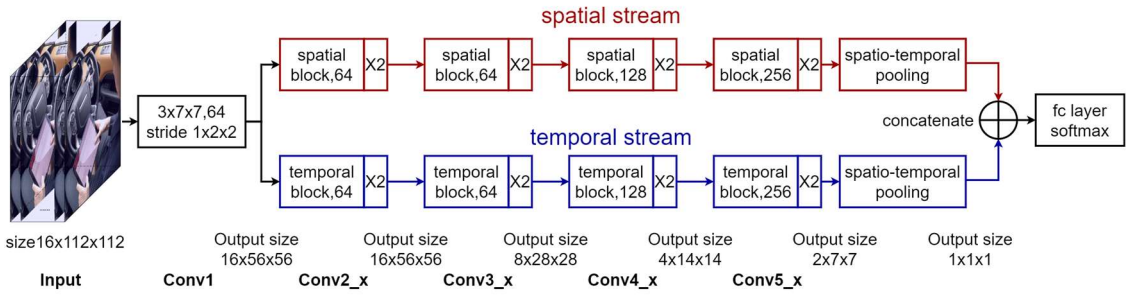


Figure 4-3 The proposed network architecture. The layer name is the bolded word at the bottom. The output size of each layer is on the top right of the layer name. The details of the each used blocks is introduced in Figure 4-2. Downsampling is employed on conv3_1, conv4_1, conv5_1 with a stride of 2

the feature maps produced in both 2 streams and fed into a fully connected layer, which outputs the final prediction probabilities through the Softmax function.

4.3.3 Prediction Process for the Framework

As illustrated in Figure 4-1, the prediction of the driver activity recognition framework combines the outputs from 3 separate models. The prediction probability of NDRA engagement recognition based on the driver's head movement is denoted as P_e, which has two states: DRA engagement and NDRA engagement, denoted as c_D and c_N, respectively. The prediction probability for these two classes is represented by $P_e(c_D)$ and $P_e(c_N)$. Two different 3D CNN models have been trained separately for NDRA and NDRA classification based on hand movement. The prediction probabilities for these 2 models are denoted as P_{Dc} and P_{Nc}. The final scores of the DRA classification and NDRA classification are denoted as Y_{d} and Y_{N}.

The score of a single DRA can be expressed as:

$$Y_D(i_D) = P_{Dc}(i_D)P_e(c_D) \tag{4-4}$$

where i_D is the index of the DRAs. The score of a single NDRA can be expressed as:

$$Y_N(i_N) = P_{Nc}(i_N)P_e(c_N) \tag{4-5}$$

where i_N is the index of the NDRAs. The final prediction scores for all NDRAs and DRAs classes, denoted by Y, can be expressed as:

$$Y = Y_{\mathrm{D}} \cup Y_{\mathrm{N}} \tag{4-6}$$

4.3.4 Visual Explanations of CNN Model Predictions

With the effort of visual explanation for CNN [30]–[32], we can explain the prediction of the instance made by the evaluated 3D CNN models, which allows a better understanding of the features learned. In this study, Grad-CAM++ [31] was employed for visualisation. This method provides the visual explanation of the model based on the pixel-wise weighting of the gradients of the convolution

feature map. It measures the importance of each pixel in the convolutional feature map towards the final prediction of the model.

The classification score Y^c for class c can be expressed as:

$$Y^c = \sum_k w_k^c \sum_i \sum_j \sum_h A_{ijh}^k \tag{4-7}$$

where A_{ijh}^k is the feature map of a particular spatial location (i, j, h), w_k^c is the weight for the feature map A^k and class c.

The class-based saliency map M^c used for the final visual explanation can be expressed as:

$$M_{ijh}^c = relu\left(\sum_k w_k^c A_{ijh}^k\right) \tag{4-8}$$

In the Grad-CAM++ [31], the weights w_k^c is calculated by a weighted average of the pixel-wise gradients, which can be written as:

$$w_k^c = \sum_i \sum_j \sum_h \alpha_{ijh}^{kc} \, relu\left(\frac{\partial Y^c}{\partial A_{ijh}^k}\right) \tag{4-9}$$

where α_{ijh}^{kc} is the weighting coefficients and the $\frac{\partial Y^c}{\partial A_{ijh}^k}$ is the pixel-wise gradient for feature map A^k and class c.

Considering Equation (4-9), Equation (4-7) can be rewritten as:

$$Y^c = \sum_k \left[\sum_a \sum_b \sum_d \alpha_{abd}^{kc} \, relu\left(\frac{\partial Y^c}{\partial A_{abd}^k}\right)\right] \sum_i \sum_j \sum_h A_{ijh}^k \tag{4-10}$$

where (a, b, d) and (i, j, h) are iterators for the same activation map A^k for avoiding confusion. $relu$ has been dropped in the derivation since the function of which is as a threshold for allowing the gradients to flow back. Taking partial derivative A_{ijh}^k twice on both sides:

$$\frac{\partial^2 Y^c}{(\partial A_{ijh}^k)^2} = 2\alpha_{ijh}^{kc} \frac{\partial^2 Y^c}{(\partial A_{ijh}^k)^2} + \sum_a \sum_b \sum_d A_{abd}^k \left(\alpha_{ijh}^{kc} \frac{\partial^3 Y^c}{(\partial A_{ijh}^k)^3} \right) \quad \textbf{(4-11)}$$

Based on Equation (4-11), α_{ijh}^{kc} can be calculated as:

$$\alpha_{ijh}^{kc} = \frac{\frac{\partial^2 Y^c}{\left(\partial A_{ijh}^k\right)^2}}{2\frac{\partial^2 Y^c}{\left(\partial A_{ijh}^k\right)^2} + \sum_a \sum_b \sum_d \left(A_{abd}^k \frac{\partial^3 Y^c}{\left(\partial A_{ijh}^k\right)^3} \right)} \quad \textbf{(4-12)}$$

Considering Equation (4-11), Equation (4-9) can then be rewritten as:

$$w_k^c = \sum_i \sum_j \sum_h \frac{\frac{\partial^2 Y^c}{\left(\partial A_{ijh}^k\right)^2}}{2\frac{\partial^2 Y^c}{\left(\partial A_{ijh}^k\right)^2} + \sum_a \sum_b \sum_d \left(A_{abd}^k \frac{\partial^3 Y^c}{\left(\partial A_{ijh}^k\right)^3} \right)} relu\left(\frac{\partial Y^c}{\partial A_{ijh}^k} \right) \quad \textbf{(4-13)}$$

4.4 Dataset and Training

To evaluate the proposed method, this study produced a new dataset, which contains the driver's head and hand movement footages captured by 2 cameras during the experiment. There are 6 classes in this dataset, including 4 types of NDRAs and 2 types of DRAs. 14 participants (12 male and 2 female) were recruited for this experiment who are from 8 different counties. The participants' age is in the range from 23 to 35. They were required to hold a valid UK driving license. The videos were recorded in different weather and lighting conditions including sunny, cloudy, rainy and snowy.

4.4.1 Experiment Design

The vehicle used in the experiment was an instrumented Land Rover Discovery 5. The car was modified to accommodate both automated driving and human driving. During the experiment, the vehicle is in automated driving mode and following a designed route on the enclosed roads. To ensure safety, a steering wheel and a set of pedals were added in the back seat of the vehicle, which allows

the safety driver to intervene and override the autonomous system. The participants were required to engage in some activities while the vehicle is under the automated driving mode. After a period of time, the driver was asked to take over the vehicle and drive for 2 minutes. Four types of NDRA investigated in this study are *reading news, watching videos, playing games* and *answering questionnaires* using a tablet. These activities were selected by considering the outcomes from surveys [33], [34]. The DRAs considered in this study are *road checking* and *driving*. For each participant, the engagement of each activity (4 types of NDRA and road checking) lasted 5 to 9 minutes followed by a 2 minutes driving process, which is considered as one single trial. There are 5 trials per participant. The data of 4 NDRA classes were extracted from the corresponding trials. The data for the road checking class contains the data extracted from the road checking trial and the data of the road checking behaviour during the NDRA engagement trials. The data for driving was obtained by extracting the data where the participant was driving the vehicle after the take-over.

4.4.2 Camera Setup

The employed 2 cameras for monitoring the driver's behaviour in the experiment were Garmin Virb Action Camera, which provides the videos with 1920 × 1440 pixels spatial resolution and frames were sampled at 30 frames per second (fps). The front camera, facing the driver's face, is used to extract the driver's head

Figure 4-4 Location of the mounted cameras

movement and recognise whether the driver is engaging with NDRAs or DRAs. The rear camera was mounted on the roof of the vehicle between two front seats to record the driver's hand movement. The location of the cameras is shown in Figure 4-4. A flashing red LED light was employed for synchronisation, which can be seen in the view of both cameras.

4.4.3 Data Pre-processing

In the dataset for the driver activity recognition framework, a single instance, denoted by I, contains a pair of synchronised frame stacks (I_f, I_r) from the front camera and rear camera, respectively. The recorded video from each camera was split into several clips. We removed some bad clips which contain the participant's behaviour during the activity transition. The activity is difficult to be determined in such clips, such as the mixture of *road-checking*, *playing games*, etc. As shown in Figure 4-5, there are 48 frames in each clip, which were cropped with a 600×600 region of interest and further resized into 128×128. The dimension of the frames for each clip is $3 \times 48 \times 128 \times 128$. Then the 16 adjacent frames were randomly sampled and used as an input instance of I_f or I_r. The size of an input instance is $3 \times 16 \times 112 \times 112$. There are 7960 pairs of instances for 6 classes in total. The distribution of all these classes is *answering questionnaires* (1336), *road checking* (1320), *driving* (1268), *playing games*

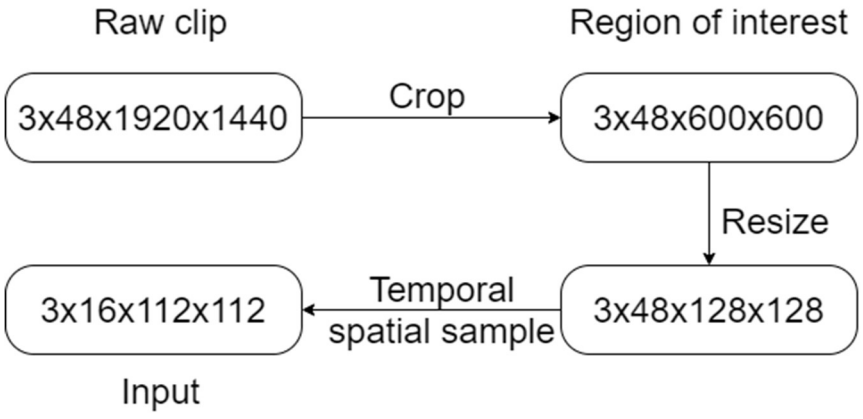


Figure 4-5 Data pre-process flowchart. The data format is presented as a four-dimensional tensor as $c \times l \times h \times w$, where c is the number of channels, l is the number of frames in the clip, h and w are the height and width of images, respectively

(1356), *reading* (1422) and *watching videos* (1258). The data were randomly split into 5 different segments for cross-validation based on participants. For each split, the data of 11 participants were used for training and the data of 3 participants were used for testing. The data distribution for 5 splits is split 1 (6158 for training and 1802 for testing), split 2 (6332 for training and 1628 for testing), split 3 (6176 for training and 1784 for testing), split 4 (6222 for training and 1738 for testing) and split 5 (6186 for training and 1774 for testing).

4.4.4 Training setup

The proposed method is compared with 3 state-of-the-art methods, including

(1) 3D ResNets (R3D) [18] that mainly utilises the basic $3 \times 3 \times 3$ residual block in the whole network to model the spatial-temporal information. Frequent usage of 3D convolution causes a higher computational cost.

(2) (2+1)D ResNets (R(2+1)D) [28] that factorises the 3D convolution of the residual block in R3D into two separate operations, which are a 2D spatial convolution and a 1D temporal convolution. Although such a structure doubles the number of nonlinearities to improve the model's capability of representing complex functions, the number of parameters and the computational cost is not decreased in comparison to the 3D CNN.

(3) Pseudo-3D ResNets (P3D) [27] that has the same method of factorisation with R(2+1D) but develops 3 blocks with different types of connection. It also adapts the bottleneck block in the network. However, the performance is not significantly improved than the simple and homogenous R(2+1D) network.

(4) The proposed DS3D ResNet.

Table 4-1 Comparison of the model size and the computational complexity. All models are based on ResNes-18 architecture

Model	Parameters ($\times 10^6$)	FLOPs ($\times 10^9$)
R3D	33.1	83.1
R2+1D	33.2	85.2
The proposed DS3D	11.8	72.5

For a fair comparison, all networks adapt 18 layers except P3D. Considering the specific design of the P3D architecture, the input size is $3 \times 16 \times 160 \times 160$. We also keep the same crop ratio from the raw frames as other models. The evaluated P3D model was built based on ResNets-50 architecture. All four models were trained from scratch on the same dataset. The size and computational complexity for these models are provided in Table 4-1, which shows the proposed model has the lowest computational cost and smallest model size.

In the training process, Adam was used for parameter optimisation with a mini-batch size of 32. The initial learning rate was set as 0.001, which was divided by 10 after every 10 epochs. The whole training was completed in 35 epochs. The task of NDRA engagement recognition adapts all the head movement dataset I_f. The tasks of NDRA classification and DRA classification use the corresponding data in the hand movement dataset I_r.

Table 4-2 Accuracy of the evaluated models on the produced dataset

Term	NDRAs engagement recognition				DRAs classification			
	R3D	R2+1D	P3D	DS3D	R3D	R2+1D	P3D	DS3D
Split 1	83.74%	87.79%	88.95%	93.90%	90.67%	93.08%	90.67%	95.71%
Split 2	87.78%	90.41%	89.07%	94.71%	91.70%	92.38%	92.21%	96.71%
Split 3	88.96%	90.92%	89.35%	93.57%	87.29%	90.28%	90.65%	90.46%
Split 4	88.15%	88.90%	92.28%	92.87%	91.36%	92.57%	89.46%	92.57%
Split 5	88.84%	90.19%	92.27%	93.63%	90.65%	86.06%	87.30%	93.47%
Mean	87.49%	89.64%	90.38%	**93.74%**	90.33%	90.87%	90.06%	**93.78%**
Term	**NDRAs classification**				**Fusion result**			
	R3D	R2+1D	P3D	DS3D	R3D	R2+1D	P3D	DS3D
Split 1	80.96%	83.81%	81.12%	87.20%	70.31%	74.64%	73.81%	83.46%
Split 2	84.19%	86.95%	84.95%	89.62%	75.43%	82.74%	77.27%	87.59%
Split 3	83.58%	84.38%	84.06%	85.10%	76.12%	78.98%	78.59%	82.90%
Split 4	81.10%	82.48%	82.14%	84.30%	74.51%	77.62%	76.41%	80.32%
Split 5	80.20%	82.60%	82.43%	83.10%	75.08%	76.16%	76.10%	82.47%
Mean	82.01%	84.04%	82.94%	**85.86%**	74.29%	78.03%	76.44%	**83.35%**

4.5 Results

The comparison results, based on the testing data for each spilt, are presented in Table 4-2, which shows the models' accuracy for 3 tasks and the final fusion results. For the task of NDRA engagement recognition (NDRA or DRA), the average accuracy of R3D for 5 splits is 87.49%. The performance of R2+1D and P3D is similar and around 90%. The proposed DS3D model achieves 93.74% average accuracy on this task. For the task of DRAs classification (*driving* or *road checking*), all 3 state-of-the-art methods achieve similar performance while our model has at least 3% improvement than them. For the task of NDRA classification (*reading news, watching videos, playing games* or *answering questionnaires*), the average accuracy of R3D, R2+1D and P3D models is 82.01%, 84.04% and 82.94%, respectively, while the accuracy of our model is 85.86%. For the final fusion result for the classification of all 6 activities, it can be observed that the proposed model achieves the best performance among the evaluated methods with at least 5% improvement.

The confusion matrices of the final fusion predictions are presented in Figure 4-6. Precision and recall are used to evaluate the model in this study. Precision is the fraction of correct instances among the detected instances, while recall is the fraction of correctly detected instances [35]. For the category checking, the precisions of the 3 state-of-the-art models are around 50%~60%. The main contribution of the false positive examples is from NDRAs. It means that some NDRAs have been predicted as DRAs by being misclassified as *checking*, which suggests the poor performance of NDRA engagement recognition for these models based on the participants head movement. For both DRAs (*checking* and *driving*), the proposed DS3D achieves the best performance, specifically, 90.2% precision and 90.6% recall for *driving*. For NDRA classification, *answering questionnaires* and *playing games* have a better performance than the other two activities for all 4 models. his is because these activities normally involve a high-frequency interaction between the participant's hand and the device. The superior performance of our model is benefited from the new structure design that enhances the spatial-temporal representations. The detailed contribution will be given in the next section with the saliency map. The recall of the other activities

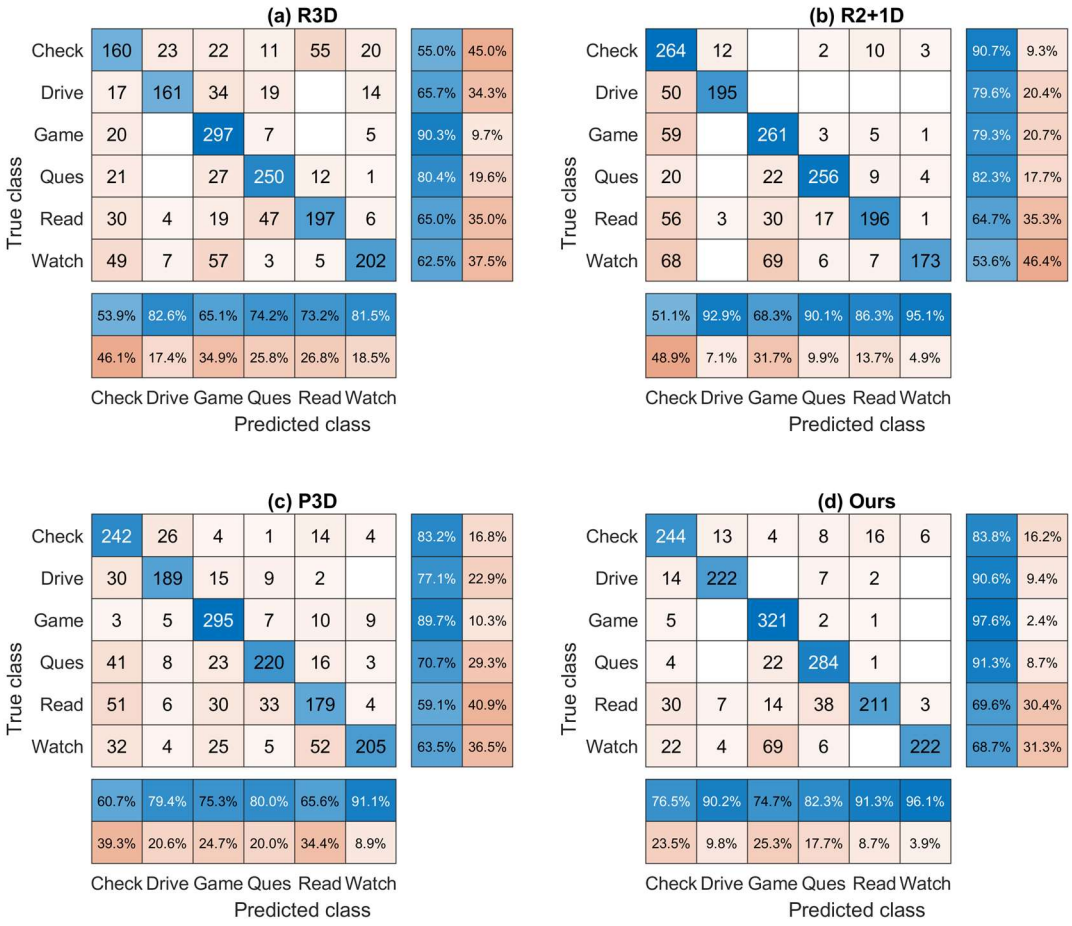


Figure 4-6 Confusion matrix of the fusion results. The models used are trained on split 1. The precision and recall for each class are presented in the bottom and right of the figures, respectively. The classes presented in the figure refer to the activities named: *road checking, driving, playing games, answering questionnaires, reading news* and *watching videos*, successively

reading and watching videos for R3D, R2+1D, and P3D is around 60%~65%. The poor performance of these activities is due to similar observations associated with limited human-object interaction or hand movement in the temporal domain. The frames do not contain sufficient spatial-temporal information to make the right prediction for these activities. Even though, our model also outperforms the other evaluated models.

4.6 Visualisation and discussion

This section provides the visualisation results of the class-based saliency map in the hidden layer of the model trained on the dataset containing hand movement

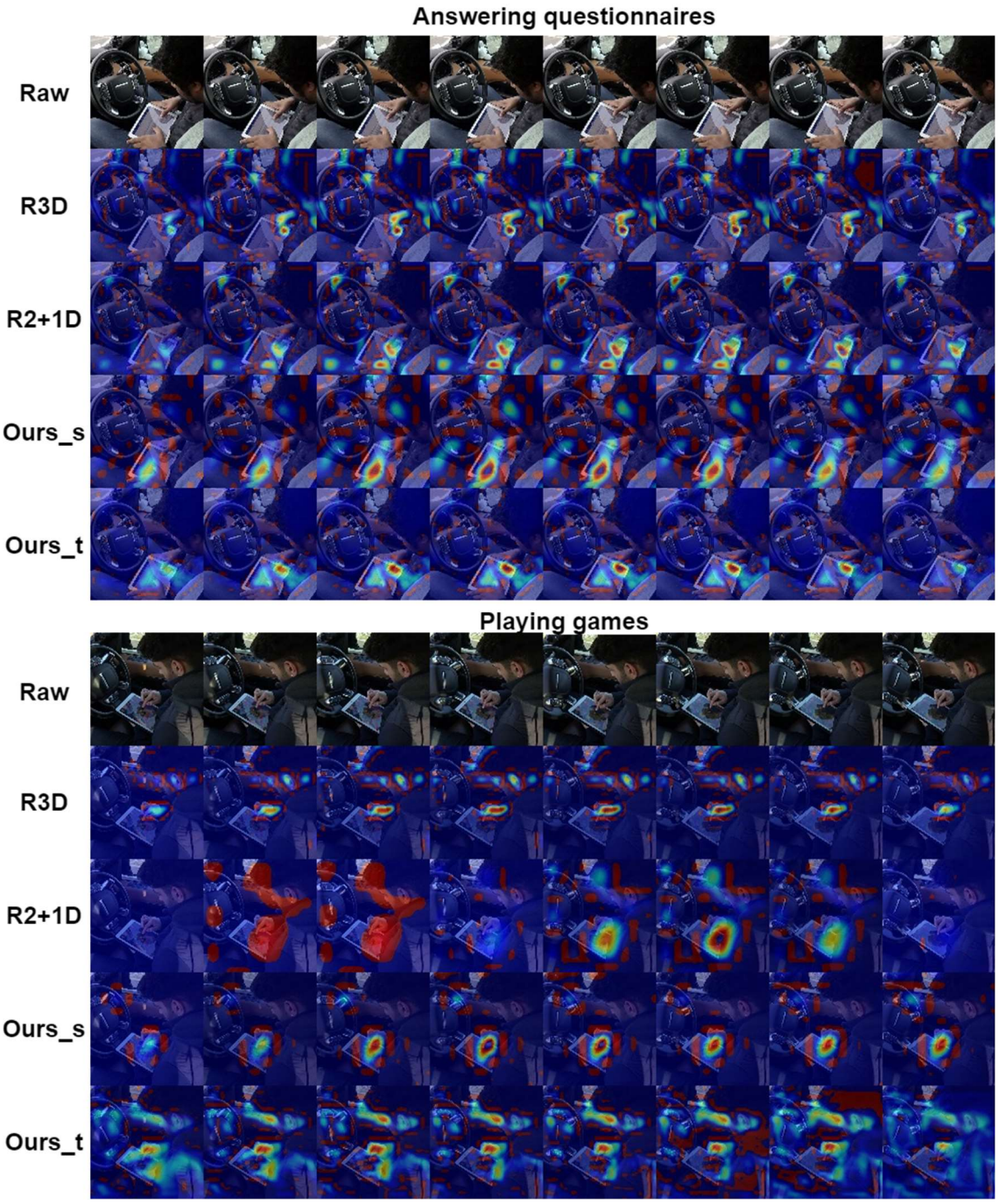


Figure 4-7 Saliency maps of the prediction based on the last convolutional layer of Conv3 by using Grad-CAM++ [31] for *answering questionnaires* and *playing games*. The first row of each activity is the raw frames imported into the network to explain the learned spatio-temporal feature. The images that contain facial information are not presented in this section due to the data protection policy.

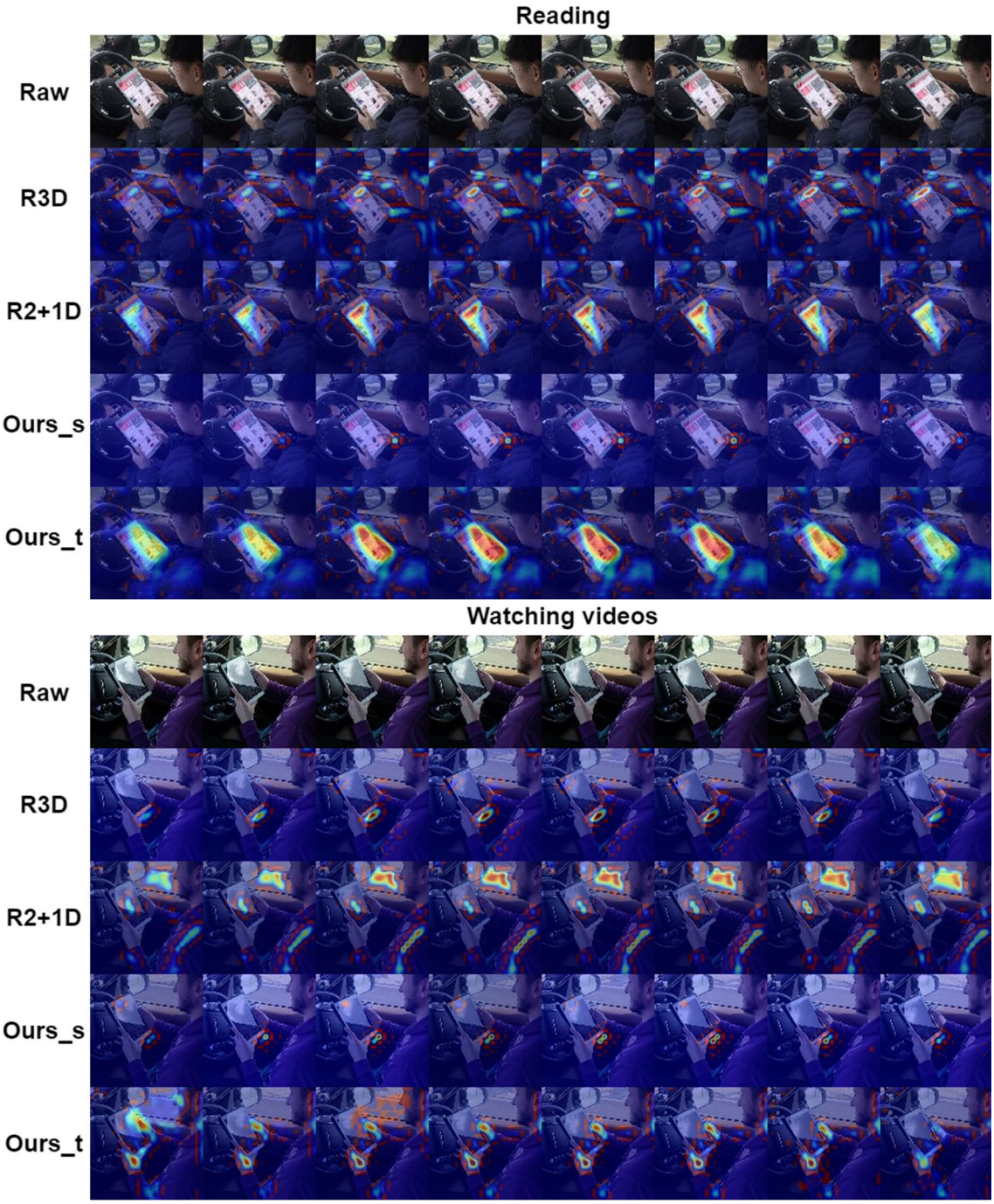


Figure 4-8 Saliency maps of the prediction based on the last convolutional layer of Conv3 for *reading* and *watching movies*

In Figure 4-7, Figure 4-8 and Figure 4-9, the class-discriminative regions contributed from the hidden layer, Conv3, have been located, where the 16 frames are subsampled to 8 frames to save space. The regions in red correspond to a higher association for the class while the regions in blue represent weak

relevance. It can be seen that the saliency regions have been highlighted on the frames based on the importance of the pixels. Specifically, for NDRAs (in Figure 4-7 and Figure 4-8), the R3D model could learn the participant's hand movement when there is high-frequency interaction in the activity (*answering questionnaires* and *playing games*). For the activity like *reading* and *watching movies*, the

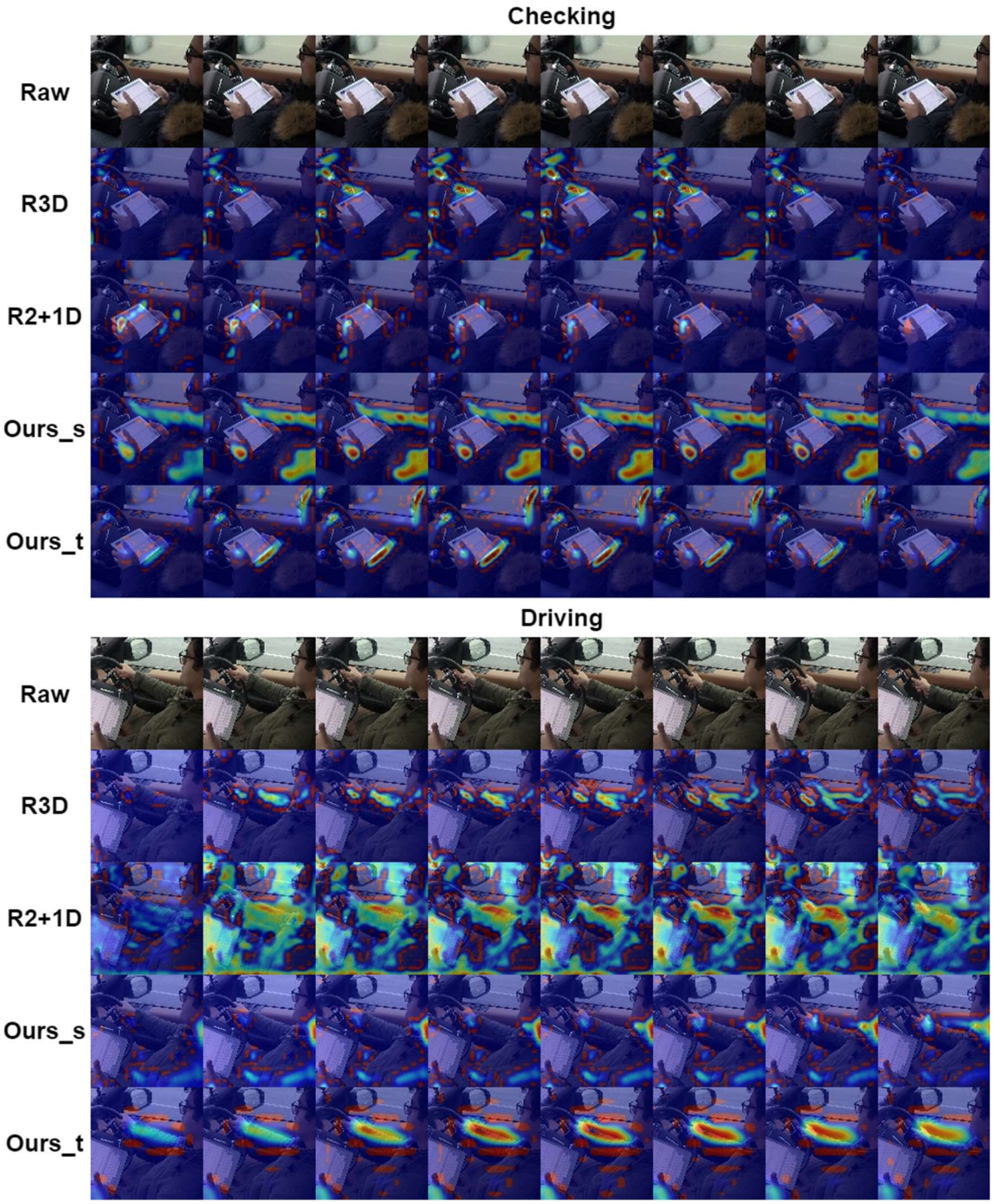


Figure 4-9 Saliency maps of the prediction based on the last convolutional layer of Conv3 for all the DAs

learned features are mainly the edge of the object. The features used in the R2+1D model to predict are based on the context of the tablet. Both two models contain some noise such as steering wheel movement and background change of the side window. Comparing with these two models, the proposed DS3D model highlights the region of the hand movement concentratedly. The spatial stream of the proposed model (denoted as *Ours_s*) focuses on short-time spatial feature learning. The temporal stream of the model (denoted as *Ours_t*) is to learn the small-region temporal feature. It can not only learn the short-time spatial feature, which is the high-frequency hand movement for the activities like *answering questionnaires* and *playing games*, but also the temporal feature, which is low-frequency interaction in the *reading*. Furthermore, it can give the right prediction based on the hand pose when there is a limited interaction during *watching videos*.

The saliency map results for DRA engagement are presented in Figure 4-9. For *checking*, the R3D model focuses on the edge of the steering wheel and the object. The R2+1 model highlights the region of the left hand. The spatial stream of our model encodes the information of the hand pose and the door, while the temporal stream focuses on the edge of the head and the device. It explains the participant's road-checking behaviour during the NDRA engagement where the participant headed up while holding the device on hands. In the *driving* category, the participant quickly steered the steering wheel with the right hand. The R3D model highlights the arm movement with its edge. The R2+1D model also learns the feature of the arm movement but with lots of noise. For the proposed model, the spatial stream captures the fast right-hand movement since it enhances the extraction of the short-time spatial change while the temporal stream mainly focuses on the participant's slight arm movement. From the perspective of the model, the R3D model learns the semantically relevant features of the high-frequency interaction activity. But for the activities like *reading, watching movies* or *road checking*, the semanteme of feature is not clear. The R2+1D shows a better classification performance than R3D, however, the explainability of the learned feature is relatively weak. Collectively, it can be observed that, for all

types of activity, the highlighted features learned by our model are more semantically relevant comparing with other models.

4.7 Conclusion

In this chapter, we propose a 2-feed 3D CNN based driver behaviour recognition system for the conditionally automated driving vehicle. Demonstrated by the testing results on the collected data, the introduced novel dual-stream 3D residual network (DS3D ResNet) presents a strong capability of encoding the spatial-temporal information for driver's behaviour. Specifically, the spatial stream extracts the short-time spatial features while the temporal stream focuses on learning the small-region temporal representation. This hypothesis has been successfully tested by visualising the saliency maps. Quantitative results demonstrate the superior performance of the proposed DS3D model against three state-of-the-art methods. From the perspective of NDRA recognition, the activities with more human-object interaction can be classified more accurately due to the contained abundant spatial-temporal features. It should be noted that the evaluation was conducted on a novel driver activity dataset. Based on the visualisation results, we believe that the capability of the proposed DS3D model has not been fully explored using the current NDRA dataset. The recognition of other NDRAs with interaction in a higher frequency, for instance, phone typing, could benefit from this model. The application of the proposed method on a comprehensive list of NDRAs requires further study.

4.8 Reference

[1] C. Lv, X. Hu, A. Sangiovanni-Vincentelli, Y. Li, C. M. Martinez, and D. Cao, "Driving-Style-Based Codesign Optimization of an Automated Electric Vehicle: A Cyber-Physical System Approach," *IEEE Trans. Ind. Electron.*, vol. 66, no. 4, pp. 2965–2975, Apr. 2019, doi: 10.1109/TIE.2018.2850031.

[2] B. W. Smith, "SAE levels of driving automation," *Cent. Internet Soc. Stanford Law Sch.*, p. 1, 2014.

[3] L. Yang *et al.*, "A refined non-driving activity classification using a two-stream convolutional neural network," *IEEE Sens. J.*, vol. XX, no. XX, pp.

1–1, 2020, doi: 10.1109/JSEN.2020.3005810.

[4] L. Yang, K. Dong, A. J. Dmitruk, J. Brighton, and Y. Zhao, “A Dual-Cameras-Based Driver Gaze Mapping System With an Application on Non-Driving Activities Monitoring,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 10, pp. 4318–4327, Oct. 2020, doi: 10.1109/TITS.2019.2939676.

[5] J. Kim, H. S. Kim, W. Kim, and D. Yoon, “Take-over performance analysis depending on the drivers’ non-driving secondary tasks in automated vehicles,” *9th Int. Conf. Inf. Commun. Technol. Converg. ICT Converg. Powered by Smart Intell. ICTC 2018*, pp. 1364–1366, 2018, doi: 10.1109/ICTC.2018.8539431.

[6] S. H. Yoon, Y. W. Kim, and Y. G. Ji, “The effects of takeover request modalities on highly automated car control transitions,” *Accid. Anal. Prev.*, vol. 123, no. September 2017, pp. 150–158, 2019, doi: 10.1016/j.aap.2018.11.018.

[7] K. Zeeb, A. Buchner, and M. Schrauf, “Is take-over time all that matters? the impact of visual-cognitive load on driver take-over quality after conditionally automated driving,” *Accid. Anal. Prev.*, vol. 92, pp. 230–239, 2016, doi: 10.1016/j.aap.2016.04.002.

[8] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, and F.-Y. Wang, “Driver Activity Recognition for Intelligent Vehicles: A Deep Learning Approach,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5379–5390, Jun. 2019, doi: 10.1109/TVT.2019.2908425.

[9] L. Yang, K. Dong, Y. Ding, J. Brighton, Z. Zhan, and Y. Zhao, “Recognition of visual-related non-driving activities using a dual-camera monitoring system,” *Pattern Recognit.*, vol. 116, p. 107955, Aug. 2021, doi: 10.1016/j.patcog.2021.107955.

[10] L. Meng *et al.*, “Interpretable Spatio-Temporal Attention for Video Action Recognition,” in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, Oct. 2019, pp. 1513–1522, doi:

10.1109/ICCVW.2019.00189.

[11] H. Chen and Z. Shi, “A Spatial-Temporal Attention-Based Method and a New Dataset for Remote Sensing Image Change Detection,” *Remote Sens.*, vol. 12, no. 10, p. 1662, May 2020, doi: 10.3390/rs12101662.

[12] S. Ji, W. Xu, M. Yang, and K. Yu, “3D Convolutional Neural Networks for Human Action Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013, doi: 10.1109/TPAMI.2012.59.

[13] J. Carreira and A. Zisserman, “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, vol. 2017-Janua, pp. 4724–4733, doi: 10.1109/CVPR.2017.502.

[14] A. Ullah, K. Muhammad, J. Del Ser, S. W. Baik, and V. H. C. de Albuquerque, “Activity Recognition Using Temporal Optical Flow Convolutional Features and Multilayer LSTM,” *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9692–9702, Dec. 2019, doi: 10.1109/TIE.2018.2881943.

[15] T. Huynh-The, C.-H. Hua, and D.-S. Kim, “Encoding Pose Features to Images With Data Augmentation for 3-D Action Recognition,” *IEEE Trans. Ind. Informatics*, vol. 16, no. 5, pp. 3100–3111, May 2020, doi: 10.1109/TII.2019.2910876.

[16] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, “HMDB: A large video database for human motion recognition,” in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 2556–2563, doi: 10.1109/ICCV.2011.6126543.

[17] K. Soomro, A. R. Zamir, and M. Shah, “UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild,” no. November, Dec. 2012, [Online]. Available: http://arxiv.org/abs/1212.0402.

[18] K. Hara, H. Kataoka, and Y. Satoh, “Learning Spatio-Temporal Features with 3D Residual Networks for Action Recognition,” in *2017 IEEE*

International Conference on Computer Vision Workshops (ICCVW), Oct. 2017, vol. 2018-Janua, pp. 3154–3160, doi: 10.1109/ICCVW.2017.373.

[19] M. Martin, J. Popp, M. Anneken, M. Voit, and R. Stiefelhagen, “Body Pose and Context Information for Driver Secondary Task Detection,” in *2018 IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2018, vol. 2018-June, no. Iv, pp. 2015–2021, doi: 10.1109/IVS.2018.8500523.

[20] Y. Xing *et al.*, “Identification and Analysis of Driver Postures for In-Vehicle Driving Activities and Secondary Tasks Recognition,” *IEEE Trans. Comput. Soc. Syst.*, vol. 5, no. 1, pp. 95–108, Mar. 2018, doi: 10.1109/TCSS.2017.2766884.

[21] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, “Large-Scale Video Classification with Convolutional Neural Networks,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1725–1732, doi: 10.1109/CVPR.2014.223.

[22] K. Simonyan and A. Zisserman, “Two-Stream Convolutional Networks for Action Recognition in Videos,” *Biochem. Pharmacol.*, vol. 32, no. 5, pp. 849–855, Jun. 2014, doi: 10.1016/0006-2952(83)90587-7.

[23] H. Xu, A. Das, and K. Saenko, “R-C3D: Region Convolutional 3D Network for Temporal Activity Detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2319–2332, Mar. 2017, doi: 10.1109/TPAMI.2019.2921539.

[24] H. M. Eraqi, Y. Abouelnaga, M. H. Saad, and M. N. Moustafa, “Driver Distraction Identification with an Ensemble of Convolutional Neural Networks,” *J. Adv. Transp.*, vol. 2019, pp. 1–12, Feb. 2019, doi: 10.1155/2019/4125865.

[25] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning Spatiotemporal Features with 3D Convolutional Networks,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, vol. 2015 Inter, pp. 4489–4497, doi: 10.1109/ICCV.2015.510.

[26] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[27] Z. Qiu, T. Yao, and T. Mei, “Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, vol. 2017-Octob, pp. 5534–5542, doi: 10.1109/ICCV.2017.590.

[28] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, “A Closer Look at Spatiotemporal Convolutions for Action Recognition,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6450–6459, doi: 10.1109/CVPR.2018.00675.

[29] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” *32nd Int. Conf. Mach. Learn. ICML 2015*, vol. 1, pp. 448–456, Feb. 2015, [Online]. Available: http://arxiv.org/abs/1502.03167.

[30] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, “Striving for Simplicity: The All Convolutional Net,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Work. Track Proc.*, pp. 1–14, Dec. 2014, [Online]. Available: http://arxiv.org/abs/1412.6806.

[31] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, “Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks,” in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2018, vol. 2018-Janua, pp. 839–847, doi: 10.1109/WACV.2018.00097.

[32] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization,” *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 336–359, Feb. 2020, doi: 10.1007/s11263-019-01228-7.

[33] M. Sivak and B. Schoettle, “Motion Sickness in Self-Driving Vehicles,” no. April, 2015, [Online]. Available: https://deepblue.lib.umich.edu/handle/2027.42/111747.

[34] F. Naujoks, D. Befelein, K. Wiedemann, and A. Neukum, “A Review of Non-driving-related Tasks Used in Studies on Automated Driving,” in *Advances in Intelligent Systems and Computing*, vol. 597, 2018, pp. 525–537.

[35] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.

5 Lightweight temporal attention-based module for efficient 3D CNN

This chapter is based on a paper submitted to *neurocomputing*.

5.1 Introduction

The automated driving vehicle could become a commercial reality in the near future, which has the potential of reducing traffic accidents [1] since it has the capability of eliminating human error to avoid the accident considering 94% of fatal crashes are caused by human error reported by the National Highway Traffic Safety Administration (NHTSA) [2]. However, fully automated driving has not been achieved yet. The automated driving vehicles that are currently on the road testing are mainly in the level 3 or 4 driving automation, defined by the Society of Automotive Engineers (SAE) [3]. In such driving automation levels, the drivers only need to take control of the vehicle if an intervention is requested, which gives them some tolerance to do some non-driving related activities (NDRAs) rather than focusing on driving. Comparing with the distraction in conventional human driving vehicles, the engagement of such activities could further reduce their surrounding monitoring and situation awareness [4], [5], which could bring an even higher risk for the driver to take over the vehicle and cause accidents. In the reported accidents that involve the automated driving vehicle, the lack of situation awareness is the main factor where the driver does not have enough time to sense the environment and conducts proper manoeuvres to avoid the accident [6]. Therefore, monitoring the drivers' state and activities that they are engaging in is crucial for the design of a smart human-machine interface (HMI) to improve their situation awareness before the take-over process.

NDRAs are defined as the tasks or activities that could happen in the automated driving vehicle but are not related to driving [7], such as reading, playing games, chatting with passengers, eating or even sleeping. Some of the NDRAs are similar to the secondary tasks in a conventional vehicle. However, the secondary task requests the driving as the primary task while the driver's engagement of NDRAs shows very limited interaction between the vehicle and the driver.

Therefore, the well-used manoeuvre-based or vehicle state-based secondary task detection approaches [8], [9] are not capable of recognising NDRAs. Other approaches such as gaze tracking-based method [10], [11], electroencephalogram (EEG)-based method [12]–[14], seat pressure measurement-based method [15], are either intrusive or costly, which limits its applicability to commercial usage for the NDRAs recognition. With the rapid and significant progress of the human action recognition made by the computer vision community in recent years, some deep learning-based computer vision approaches have been widely researched and developed to monitor the driver's behaviour and recognise NDRAs. Yang *et al*. [16] mapped the driver's gaze into a view of the vehicle cabin, which is then combined with object recognition to determine the visual-related NDRAs. Such a method has high prediction confidence of the NDRAs recognition since it directly locates the driver's visual attention, nevertheless, lacks the capability of classifying the activities with the same object. Xing *et al*. [17] extracted the driver's head rotation angles and the joint positions of the upper body then used a feedforward neural network to classify NDRAs. Similarly, Martin et al. [18] used the joint positions of the upper body and the image of the movement as the inputs then employed 3 recurrent neural networks (RNNs) to detect NDRAs. Apart from the skeleton features, Xing *et al*. [19] further extracted the driver's upper body through image segmentation and used a convolution neural network (CNN) to recognise NDRAs. All of these methods adopted the hand-crafted feature extraction and followed by the neural network-based classifier. Yang *et al*. [20] employed the CNN-based ResNet-50 to extract the features from images and combined the optical flow of the images, which presents the driver's hand movement to achieve the NDRAs recognition. Eraqi et al. [21] captured the image inside the vehicle cabin and employed multiple CNNs with the inputs of the raw image, hand image, face image, skin segmented image and used a genetic algorithm to achieve the weighted ensemble classification. Such methods are usually based on the image input, which is mainly in the spatial domain and lacks the temporal representation of the driver's behaviour during the NDRAs engagement. Yang *et al*. [22] employed a dual-stream 3D CNN, which extracts the spatio-temporal representation of the

driver's behaviour with the designed short-time spatial block and small-region temporal block, to recognise the NDRAs in the video stream. However, the 3D convolution is normally computational costly and not appropriate for real-time applications. There is a lack of efficient network architecture that monitors the driver's behaviour and recognises NDRAs from videos. Furthermore, the existing researches are mainly based on offline analysis. The methods were developed and tested on high-performance workstations with GPU, which cannot be directly used in real road-testing scenarios. There is a lack of evaluation of the inference latency, which refers to the time cost for inferring one instance, on the on-vehicle edge computing devices. This study is important because the fast inference or even prediction of NDRAs could support HMI to rapidly determine an intelligent take-over strategy and achieve a safe control transition.

This chapter proposed a novel lightweight 3D CNN-based temporal attention module for efficient CNN in video-based NDRAs recognition. Unlike the conventional 3D convolution module with the limited receptive field, the proposed module models the global information in the time domain. Specifically, the proposed module uses the 3D convolution operation in the spatial domain and further employs the attention mechanisms-based temporal weighting function to enhance the representation in the temporal domain. This module tends to achieve high accuracy with much less computational cost. The proposed module can be trained end-to-end and used as a plugin module for the existing efficient 3D CNN. In this study, the MobileNet V3 is used as the backbone architecture. The performance of the network is tested in an NDRAs dataset. Moreover, the saliency map is employed to visualise the features learned in the hidden layer of the network to validate its capability of learning the semantic representations of the activities. To further evaluate the applicability of the model on the real driving scenarios, the performance regarding inference latency and accuracy of the proposed model and several state-of-the-art has been compared on 3 types of edge computing devices from the family of the NVIDIA Jetson AI platform.

5.2 Methodology

This section introduces the network used for NDRAs recognition in detail. The sections (5.2.1, 5.2.2 and 5.2.3) elaborate the design of the proposed module in the network, which aims to reduce the computational complexity and improve the capability of learning valuable representations. Section 5.2.4 shows the structure of the proposed module and the backbone network with the module plugged. Section 5.2.5 illustrates the technique used to visualise the learned features in the network. Section 5.2.6 introduces the NDRAs dataset used in this study and the data pre-processing steps. The last section 5.2.7 presents the edge computing devices, which are used for the evaluation of on-vehicle inference latency. From now on, the 3D convolution kernel is denoted as $d \times k \times k$, where d and k are the temporal depth and the spatial size respectively.

5.2.1 Depthwise Separable Convolution

Depthwise separable convolution is a widely used convolution operation in different efficient neural network-based models [23]–[25], which factorize the conventional convolution into two operations, depthwise convolution and pointwise convolution. Unlike the conventional convolution, whose kernel computes the feature map across all channels of the input, the kernel of the depthwise convolution only applies the convolution operation for one single input channel. After applying the depthwise convolution to each input channel, the pointwise convolution, $1 \times 1 \times 1$ convolution, will build a linear combination of the output of the depthwise convolution.

The n^{th} feature map O of the conventional convolution can be calculated as:

$$O_{(i,j,h,n)} = \sum_{k,l,m,c}^{K,L,M,C} W_{(k,l,m,n)} \cdot F_{(k+i,l+j,m+h,c)} \quad \textbf{(5-1)}$$

The n^{th} feature map O_{dw} of the depthwise convolution for a single input channel can be calculated as:

$$O_{dw(i,j,h,n)} = \sum_{k,l,m}^{K,L,M} W_{(k,l,m,n)} \cdot F_{(k+i,l+j,m+h,n)} \tag{5-2}$$

The feature map O_{pw} of the following pointwise convolution can be calculated as:

$$O_{pw(i,j,h,n)} = \sum_{c}^{C} W_{(n)} \cdot F_{(i,j,h,c)} \tag{5-3}$$

where i, j, h and k, l, m are the 3-dimensional location indicators. K, L are the spatial size of the kernel, M is the temporal size of the kernel. W is the convolutional kernel, F is the input feature map, C is the number of the input channel.

The computation ratio R_{ds} between depthwise separable convolution and conventional convolution for 3D CNN can be expressed as:

$$\begin{aligned} R_{ds} &= \frac{S_W \cdot S_W \cdot S_W \cdot C \cdot S_{O_i} \cdot S_{O_j} \cdot S_{O_h} + N \cdot C \cdot S_{O_i} \cdot S_{O_j} \cdot S_{O_h}}{S_W \cdot S_W \cdot S_W \cdot N \cdot C \cdot S_{O_i} \cdot S_{O_j} \cdot S_{O_h}} \\ &= \frac{1}{N} + \frac{1}{S_W^3} \end{aligned} \tag{5-4}$$

where S_W is the kernel size and $S_{O_i} \cdot S_{O_j} \cdot S_{O_h}$ are the size of the output feature map in 3-dimension.

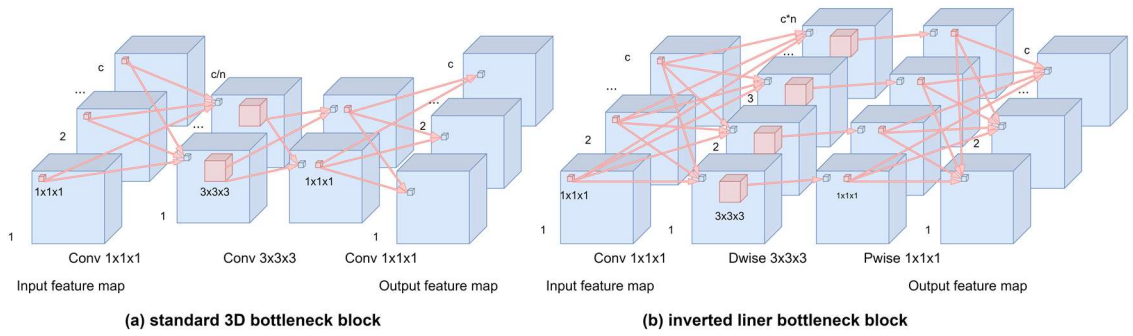


Figure 5-1 (a) standard 3D bottleneck block; (b) inverted liner bottleneck block where c is the number of the channels, n is the ratio of the channel expansion. Light red cube is the 3D convolution kernel and light blue cube is the pixel in the feature map, which is conducted by convolution operation

5.2.2 Inverted Residuals and Linear Bottlenecks

A bottleneck architecture, presented in Figure 5-1 (a), is designed to improve the model efficiency in the deep neural network [26]. $1 \times 1 \times 1$ convolution kernel is employed to compress or expand the dimensions. By this mean, the $3 \times 3 \times 3$ convolution has fewer channels, which reduces the computational complexity of the model. In the inverted liner bottleneck (Figure 5-1 (b)), since the depthwise separable convolution is employed to replace the conventional convolution, which has already significantly reduced the computational complexity, the number of channels of the $3 \times 3 \times 3$ depthwise convolution is increased to improve the capability of feature extraction. Furthermore, both batch normalization and ReLU6 activation are used after each layer. The usage of ReLU6 is due to its robustness when used with low-precision computation [27]. However, the non-linear activation transformation, ReLU6, could result in an inevitable information loss of spatial information, specifically in low-dimensional space encoding [27]. Therefore, after the dimensional compress at the end of the bottleneck block, the linear activation is employed to replace the non-linear activation.

5.2.3 Channel weighting and temporal weighting

3D CNN has been widely used to extract the spatio-temporal features from the video for human action recognition [28]–[31]. However, compared to 2D convolution, 3D convolution is more computationally expensive. Factorisation of the 3D convolution [32], [33] is a way to reduce the computational complexity. It factorises the 3D convolution operation into 2D spatial convolution and 1D temporal convolution. In this study, instead of using a $3 \times 3 \times 3$ depthwise convolution kernel, a $1 \times 3 \times 3$ depthwise convolution kernel and a 1D temporal weight function have been employed to extract the spatio-temporal features. A channel weighting module, Squeeze-and-Excitation [34] has been employed to compute the channel-wise importance.

The Squeeze-and-Excitation module introduced a way to improve the channel interdependencies with very limited computational cost. It provides the attention mechanism at the channel level. Such a module compresses the spatial and temporal information in a single channel and then adds a channel weight

function C_W, to achieve the fusion of the spatio-temporal and the channel information. The weighting function C_W can be expressed as:

$$C_W(F_{se}, W_c) = \sigma\left(W_{c_2}\delta\left(W_{c_1}AAP_{3D}(F_{se})\right)\right) \quad \textbf{(5-5)}$$

where $W_{c_1} \in \mathbb{R}^{\frac{c}{4}\times c}$ and $W_{c_2} \in \mathbb{R}^{c\times\frac{c}{4}}$, F_{se} is the input feature map for the module.

Temporal weight function T is designed base on the attention mechanism, which compresses the features in the spatial domain and weights the temporal information in a global way to extract the valuable temporal representation rather than focusing on the limited receptive field in the time domain. Such a design could further reduce the computational complexity and improve the capability of feature extraction in the time domain. The output of the temporal weight function $O_t = \left(O_{t_1}, O_{t_2}, \dots, O_{t_c}\right)$ can be expressed as:

$$\left(O_{t_1}, O_{t_2}, \dots, O_{t_c}\right) = T\left(F_{t_1}, F_{t_2}, \dots, F_{t_c}\right) \quad \textbf{(5-6)}$$

where c is the number of the feature map channel. The input of the weight function is applied with a 2D adaptive average pooling (AAP_{2D}) first to compress the spatial information. Then a 1D adaptive average pooling (AAP_{1D}) is employed, which is applied at the channel level to integrate the channel information. The pooled output goes through two fully connected layers. ReLU and sigmoid activation function was employed to add nonlinearity for the first and second fully connected layer, respectively. The weight function T can be calculated as:

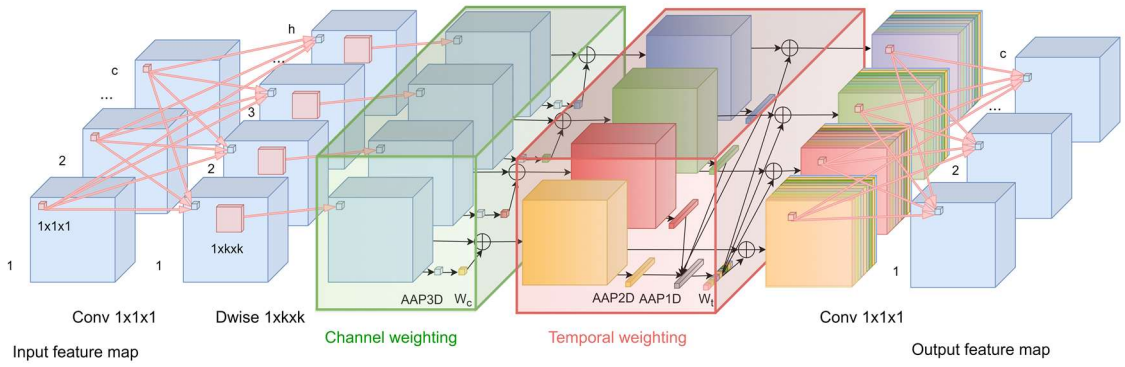


Figure 5-2 Proposed lightweight temporal attention-based module structure. where k is the kernel size and h is the number of the channels for depthwise convolution

$$T(F_t, W_t) = \sigma\left(W_{t_2}\delta\left(W_{t_1}AAP_{1D}\left(AAP_{2D}(F_t)\right)\right)\right) \quad \textbf{(5-7)}$$

where δ and σ are the ReLU and sigmoid functions. $W_{t_1}, W_{t_2} \in \mathbb{R}^{t\times t}$.

The computation ratio R_{tw} between temporal weight enabled depthwise separable convolution and 3D depthwise convolution can be expressed as:

$$R_{tw} = \frac{S_W \cdot S_W \cdot C \cdot S_{O_i} \cdot S_{O_j} \cdot S_{O_h} + C \cdot S_{O_h} \cdot S_{O_h} \cdot S_{O_h}}{S_W \cdot S_W \cdot S_W \cdot C \cdot S_{O_i} \cdot S_{O_j} \cdot S_{O_h}} \quad \textbf{(5-8)}$$

5.2.4 Model structure

The lightweight temporal attention-based module is presented in Figure 5-2. $1 \times 1 \times 1$ convolution kernel is used to expand the channel dimension. A $1 \times 3 \times 3$ depthwise convolution kernel is then employed to extract the spatial features and followed by a channel weighting function is employed to improve the learned representation at the channel level. Then a temporal weighting function, which extracts the most valuable representation in the time domain among the weighted channels. In the end, a $1 \times 1 \times 1$ convolution kernel is used to compress the dimension.

The specification of the model is presented in Figure 5-3. The MobileNet V3 is used as the backbone of the model. The inputs, including a stack of head movement frames and a stack of hand movement frames, are fed into convolution

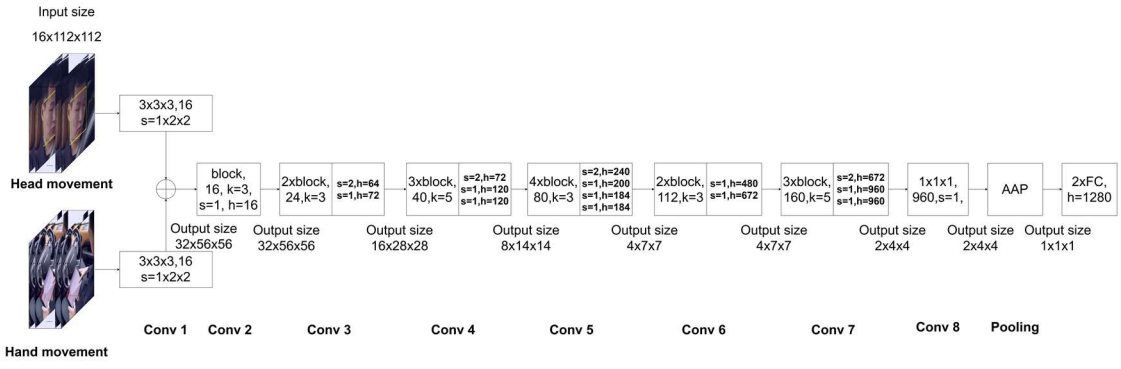


Figure 5-3 Specification of the proposed model where s is the stride of the convolution operation, h is the number of the channel for the depthwise convolution layer. AAP refers an adaptive average pooling layer; FC stands for the fully connected layer

layer1 for the initial feature extraction, separately. Then we concatenate the output together and go through 6 convolution layers with the designed block. The number of the blocks and input channels, kernel size, stride and number of the expanded hidden channels in the block are presented in Figure 5-3. A pooling layer is employed after the feature extraction to adjust the dimension of the output feature. Two fully connected layers are used for the final classification.

5.2.5 Saliency map visualisation

The method and equation can be found in section 4.3.4.

5.2.6 Dataset and pre-processing

The NDRA recognition dataset used in this study can be found in section 4.4. The data pre-processing method is introduced in section 4.4.3.

5.2.7 Hardware

NVIDIA Jetson is a high-performance AI platform for edge computing. Three Jetson modules (Figure 5-4), including Jetson Nano, Jetson TX2 and Jetson AGX Xavier, were used to test the inference latency of the model for further in-vehicle implementation. Jetson Nano is an entry-level AI development module, which could process multiple neural networks in parallel with the data acquired from

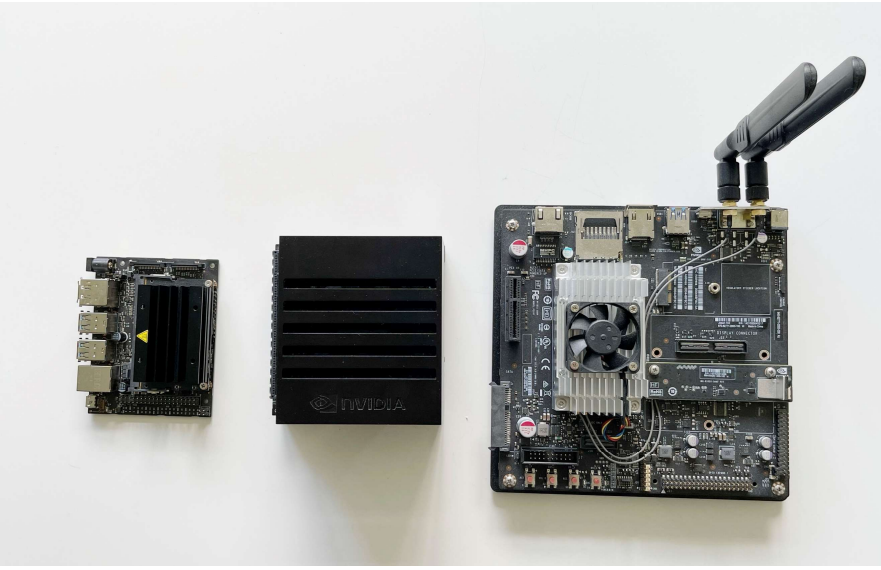


Figure 5-4 Edge computing module used in the latency test. Left: Jetson Nano. Middle: Jetson AGX Xavier. Right: Jetson TX2

Table 5-1 Technical Specifications of the Hardware

	Nano	TX2	AGX Xavier
CPU	Quad-Core Arm® Cortex®-A57 MPCore processor	Dual-Core NVIDIA Denver 2 64-Bit CPU and Quad-Core Arm® Cortex®-A57 MPCore processor	8-core NVIDIA Carmel Arm®v8.2 64-bit CPU
GPU	128-core NVIDIA Maxwell™ GPU	256-core NVIDIA Pascal™ GPU	512-core NVIDIA Volta™ GPU with 64 Tensor Cores
Memory (GB)	4	8	32
Storage (GB)	16	32	32
Power (W)	5 | 10	7.5 | 15	10 | 15 | 30

high-resolution sensors. Jetson TX2 upgrades the power efficiency and performance to another level than Nano. Jetson AGX Xavier is an edge computer designed specifically for autonomous machines. It provides the hardware acceleration for the entire AI pipeline and multiple high-speed inputs and outputs. The comparison of the technical specification is shown in Table 5-1.

5.3 Results

5.3.1 Training

Several classical efficient CNNs and state-of-the-art backbones were revised to 3D convolutions and compared with the proposed model, including

- MobileNet V1 [25]: it replaces the conventional convolution with depthwise separable convolution in the network to reduce the computational cost.
- MobileNet V2 [27]: it utilises the inverted residual structure to enhance the capability of feature learning and removes the non-linear activation in the narrow layers to maintain representational power.
- ShuffleNet V1 [35]: it employs the pointwise group convolution and channel shuffle operation to address the computational expensiveness of the pointwise convolutions.

Table 5-2 Comparison of the Model Size and the Computational Cost with Different Model Size

Model	Parameters ($\times 10^6$)				FLOPs ($\times 10^9$)			
Channel multiplier	**0.5**	**1**	**1.5**	**2**	**0.5**	**1**	**1.5**	**2**
MobileNet V1	1.73	6.6	14.61	25.76	0.19	0.48	0.85	1.32
MobileNet V2	1.30	4.3	9.33	16.28	0.42	1.12	2.1	3.37
ShuffleNet V1	0.52	1.89	4.13	7.22	0.15	0.4	0.69	1.07
ShuffleNet V2	0.53	2.13	4.37	8.83	0.25	0.39	0.58	0.87
MobileNet V3	2.51	7.68	16.71	28.91	0.31	0.73	1.45	2.22
The proposed	1.45	4.31	9.64	16.94	0.25	0.63	1.29	2.02

- ShuffleNet V2 [36]: it further introduces a channel split operator to decrease the latency on the work device.
- MobileNet V3 [37]: it is produced by the network architecture search techniques, which strike the best trade-off between performance and latency. It also employs the squeeze and excitation structure to improve accuracy.

The methods' performance was evaluated with multiple sizes of the model that are controlled by the channel multiplier, which is used to adjust the channel number in the convolutional layer. The channel multiplier was set as 0.5, 1, 1.5, and 2. Since the input of the network is a pair of head and hand movements, the networks extract the features separately and combine both feeds at the high level with the adaptive average pooling before the classifier. The model size and the computational cost for all the networks are presented in Table 5-2. It can be seen that the proposed model has a similar level of model size with MobileNet V2.

In the training process, Adam was adapted as a parameter optimisation with a mini-batch size of 64. The initial learning rate was set as 0.001. The whole training epoch was set as 60.

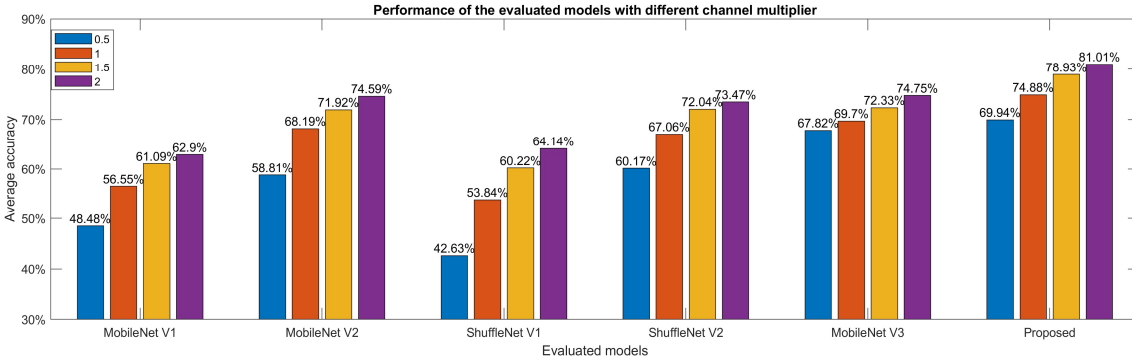


Figure 5-5 Average accuracy of all the splits for evaluated models with a set of channel multiplier (0.5, 1, 1.5, 2)

5.3.2 Results

The performance of the evaluated models has been presented in Figure 5-5. With the channel multiplier increase from 0.5 to 2, the classification accuracy of the models also increases from 22% (ShuffleNet V1) to 7% (MobileNet V3). The MobileNet V2, V3 and ShuffleNet V2 have a similar level of performance when the channel multiplier is larger than 0.5, which is around 68% (1), 72% (1.5) and 74% (2). ShuffleNet V1 and MobileNet V1 also achieve similar performance when the model is relatively large. ShuffleNet V1 (0.5) shows the lowest accuracy (42.63%), which is due to the limited number of channels that restricts its capability of feature extraction. Comparing with other models, the proposed model shows the best performance among all the channel multipliers and achieves 81.01% average accuracy when the multiplier is 2. To present the

Table 5-3 Performance of the Model When the Channel Multiplier set as 2 for All Splits in Dataset

	Accuracy					
	Split1	**Split2**	**Split3**	**Split4**	**Split5**	**Mean & Std**
MobileNet V1	62.14%	61.68%	62.77%	64.21%	63.69%	62.90% ± 1.05%
MobileNet V2	73.99%	74.63%	75.87%	72.84%	75.64%	74.59% ± 1.24%
ShuffleNet V1	63.32%	64.25%	64.09%	65.69%	63.34%	64.14% ± 0.97%
ShuffleNet V2	73.90%	74.41%	73.24%	72.25%	73.54%	73.47% ± 0.81%
MobileNet V3	74.52%	75.32%	73.56%	74.05%	76.38%	74.75% ± 1.10%
The proposed	80.32%	79.80%	81.70%	82.56%	80.66%	81.01% ± 1.11%

Table 5-4 Comparison of Latency for Different Device and Different Channel Multiplier

	Latency (ms)								
Device	**Jetson Nano**	**Jetson TX2**				**Jetson AGX Xavier**			
Channel Multiplier	**0.5**	**0.5**	**1**	**1.5**	**2**	**0.5**	**1**	**1.5**	**2**
MobileNet V1	N/A	304	397	482	531	98	131	148	163
MobileNet V2	634	339	488	679	867	133	206	232	278
ShuffleNet V1	532	251	275	300	361	114	119	132	145
ShuffleNet V2	374	243	355	541	772	116	128	145	162
MobileNet V3	509	323	423	507	686	148	168	210	258
The proposed	417	262	297	376	458	121	135	155	174

generalization capability of the model, Table 5-3 shows the results of the cross-validation for each split when the multiplier is 2. It can be observed that the standard deviation of all the models is under 1.5%. The values for ShuffleNets are below 1%. The standard deviation for the rest modes is around 1.1%

To evaluate the latency, the models were evaluated on 3 edge computing devices and the results are shown in Table 5-4. Due to the limited computational capability of the GPU on Jetson Nano, only the smallest model (channel multiplier is 0.5) can be implemented. MobileNet V1 cannot be implemented because of its poor memory efficiency on GPU. The proposed method achieves a relatively low latency, which is 417ms among all the evaluated models. For the Jetson TX2, ShuffleNet V1 achieves the least latency across all sizes of the model. The proposed method has a slightly higher latency than ShuffleNet V1 and is below 0.5s for the largest model. MobileNet V2 achieves the highest latency. Similar results can be found on the Jetson AGX Xavier implementation. However, with the increase of the computation capability of GPU on the device, ShuffleNet V1 shows a slight faster inference than the proposed method. On this device, the proposed method could complete at least 5 inferences in a second. Combining Table 5-2 and Table 5-4, it can be observed that the computational complexity (FLOPs) of the network dominates the cost of the inference. However, in the real

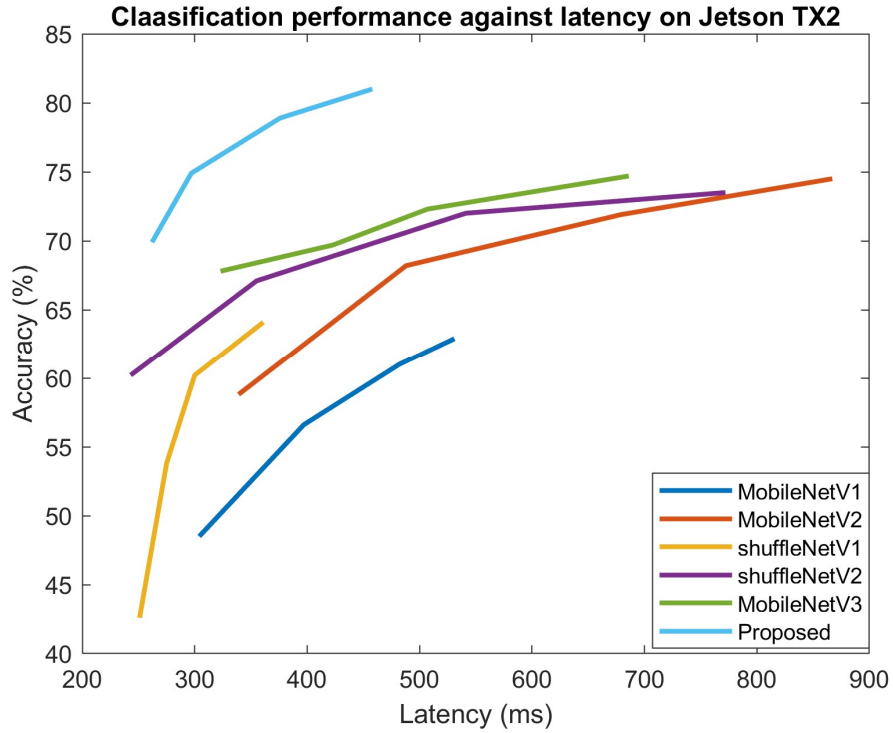


Figure 5-7 Classification and inference performance of the model on Jetson TX2

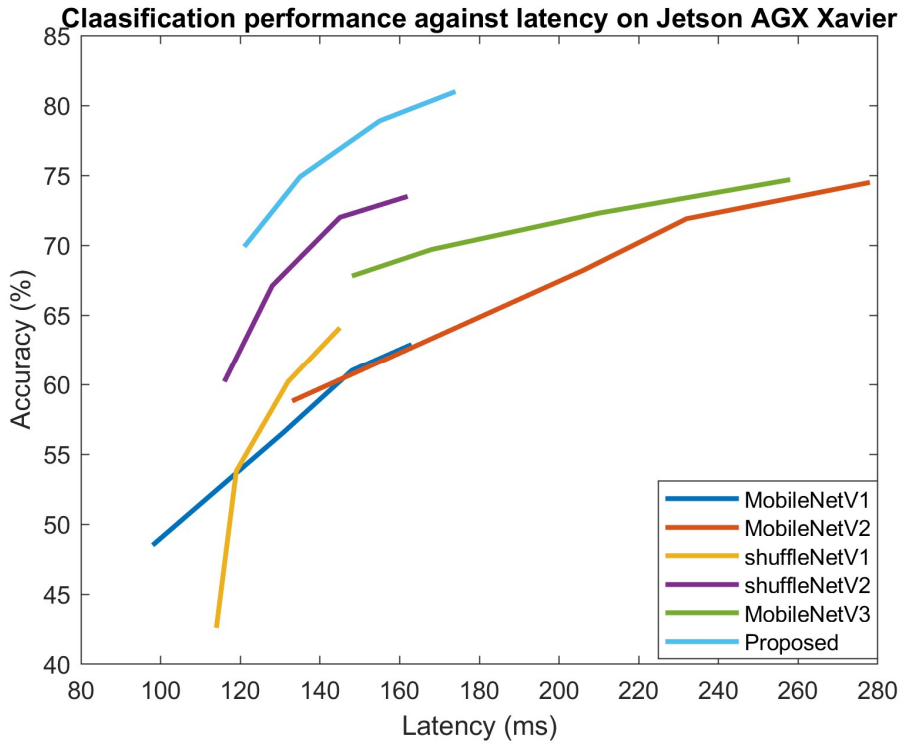


Figure 5-6 Classification and inference performance of the model on Jetson AGX

GPU implementation, the proper design of the network could speed up the inference process and further reduce the latency. Figure 5-7 and Figure 5-6 plot

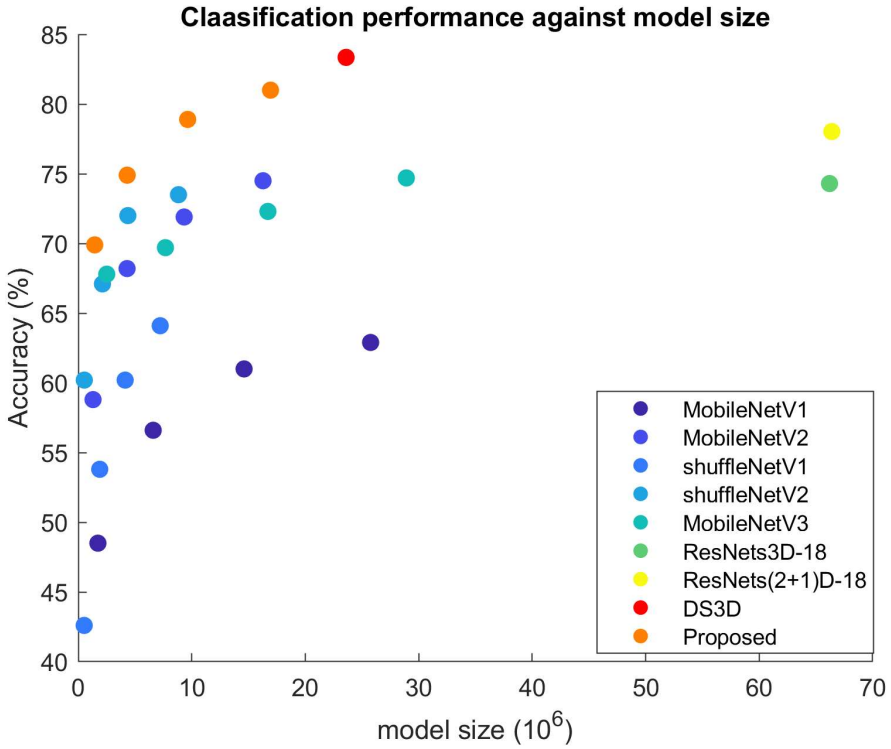


Figure 5-8 Classification performance against the model size

the model's accuracy against latency on Jetson TX2 and Jetson AGX Xavier, respectively. It can be seen that, as the model becomes larger, the inference latency and accuracy also increase. On a different device, the overall performance could be different. For instance, on Jetson TX2, MobileNet V3 achieves a higher accuracy with a similar level of latency than ShuffleNet V2. However, on Jetson AGX Xavier, the inference latency of ShuffleNet V2 is much less than MobileNet V3, which is crucial for the NDRAs detection. In both figures, the proposed model outperforms all the evaluated state-of-the-art models. Figure 5-8 presents a performance comparison between the evaluated efficient models with the conventional 3D CNN models in terms of model size. It has been observed that DS3D [22] achieves the best classification accuracy. In terms of accuracy, the proposed method is only 2.5% less than DS3D, however, it only has a 75% model size of DS3D. The FLOPS of the proposed model is only 2.8% of DS3D.

5.3.3 Saliency may visualisation

This section visualises the features learned in the hidden layer through the saliency map. The input of the network contains both head and hand movements. Due to the data protection policy, the data contains facial information is not included in this section. The 16 hand movement frames are downsampled to 8 for an easy display. The class-discriminative saliency map of the last convolution layer in Conv4 is presented in Figure 5-9. The results of ShuffleNet V2 (SV2) and MobileNet V3 (MV3) are used for comparison. It can be seen that, for the first 3

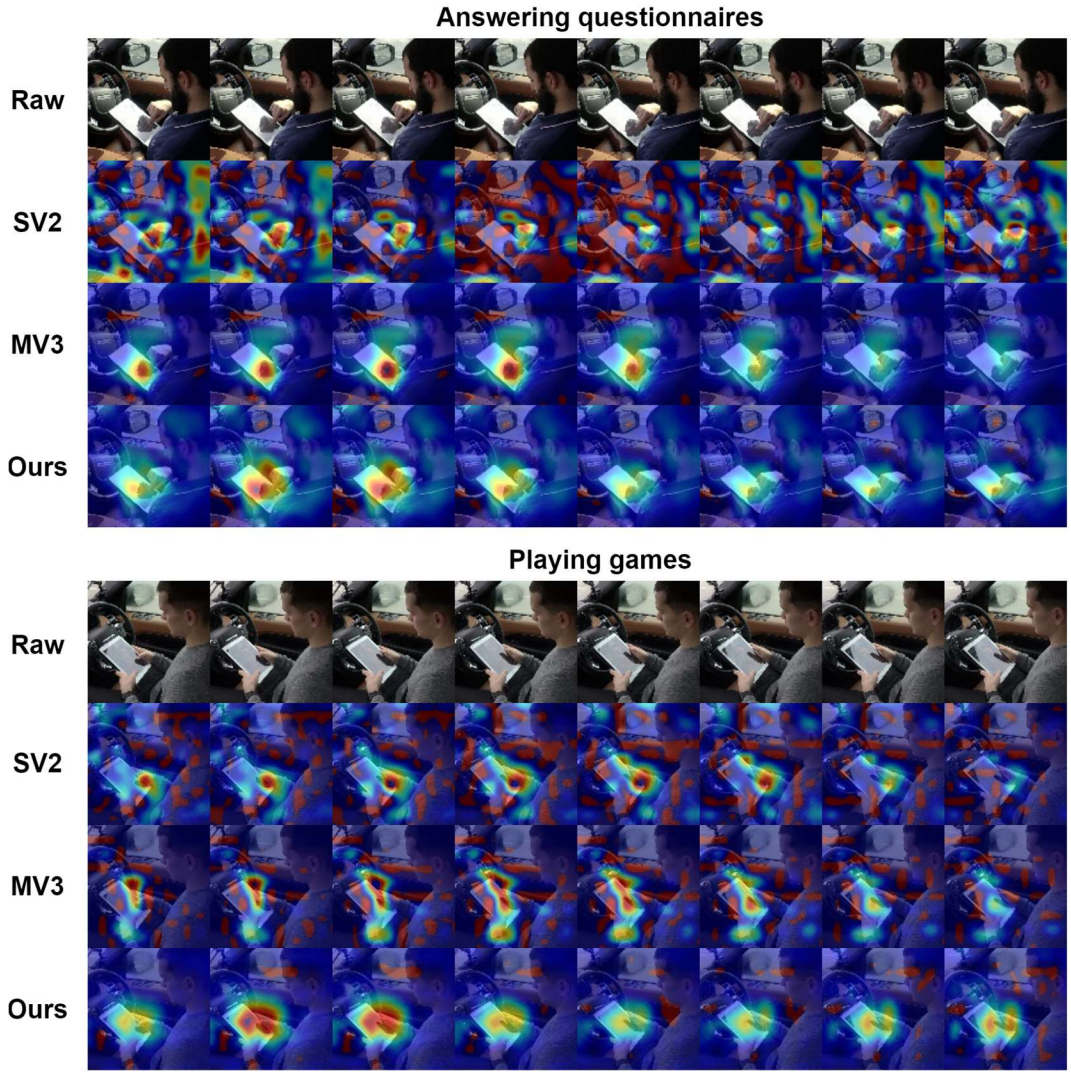


Figure 5-9 Class-discriminative saliency maps of the last convolutional layer of Conv4 for first 2 NDRAs. The first row of each activity is the raw frames imported into the network. The red regions refer to a higher association with the final classification while the regions in blue show the weak relevance

NDRAs, the hand movement has been highlighted in SV2. However, it contains more noise comparing with the other two methods. For *answering questionnaires* and *playing games*, both MV3 and the proposed method learned the spatio-

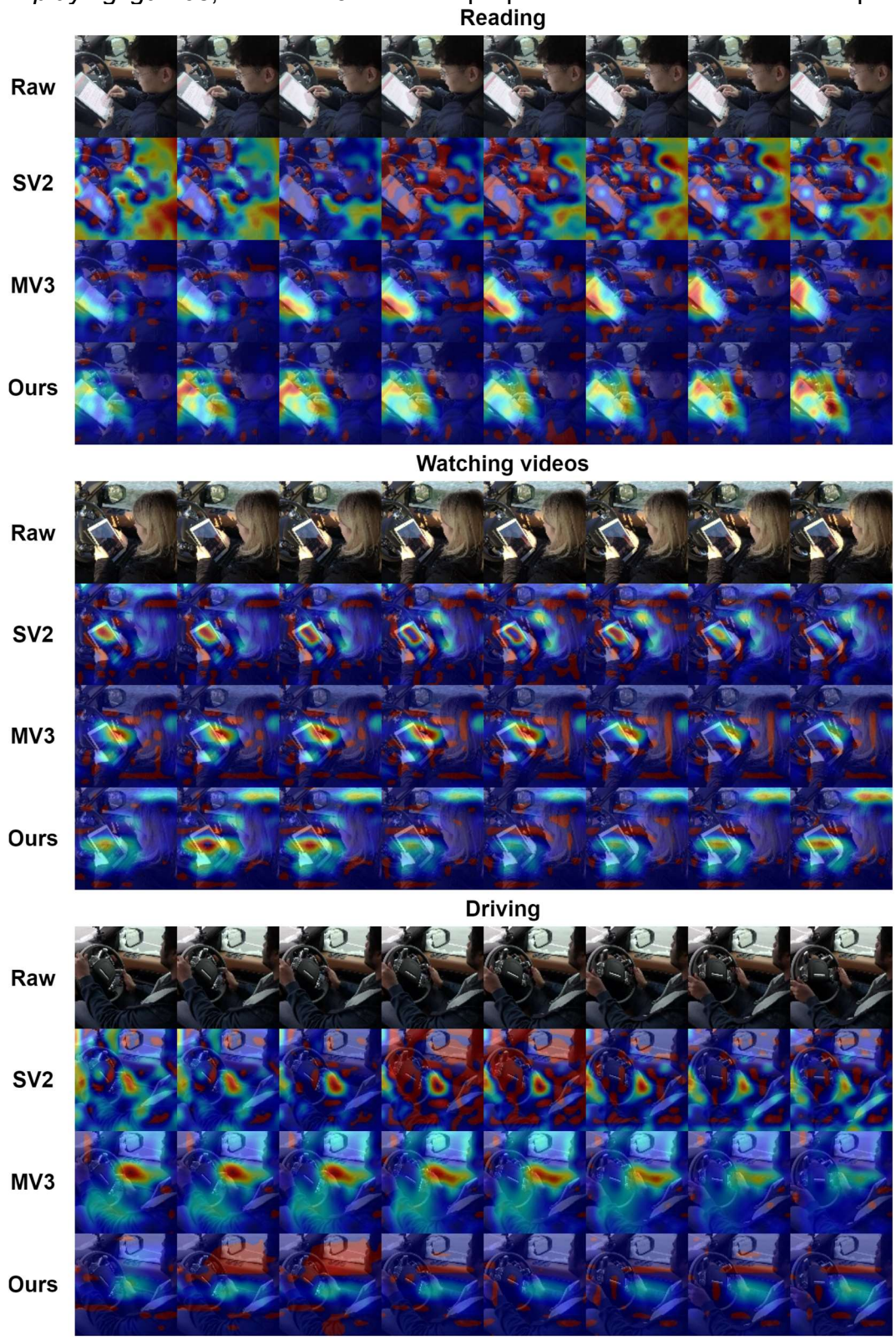


Figure 5-10 Class-discriminative saliency maps for last 3 NDRAs

temporal features of hand movement. The results of the proposed method show a higher sensitiveness in the time domain. It highlights a couple of frames with the key finger movement in the action classification while the features learned by MV3 cover a longer period. For *reading*, the MV3 model mainly focuses on the tablet location while the proposed method covers both tablet and hand features. For the activity of *watching videos*, the learned features of SV2 are mainly from a part of the tablet and the MV3 highlights the region which is above the driver's hand. The features that the proposed method learned are the spatial relationship between both hands and the tablet. For *driving*, SV2 captures the driver's both hands movement. Even though MV3 highlights the driver left arm movement, the main focus is around the door. The proposed method focuses on the driver's right-hand movement. To sum up, comparing these two models, the proposed method shows a higher semantic relevance of the extracted spatio-temporal features. The proposed temporal attention module presents a stronger capability of extracting semantic representation of the hand movement in the time domain.

5.4 Conclusion

In this chapter, an efficient and low-latency CNN based temporal attention module has been proposed, which learns the spatio-temporal representation through spatial convolution and the attention enhanced temporal weighting. Unlike the conventional 3D convolution operation, the proposed module could enhance the learned representation in the temporal domain with lower computational complexity. In this study, the performance of the proposed module with MobileNet V3 as backbone has been evaluated on an NDRAs recognition dataset. The results demonstrate a significant improvement of accuracy against several state-of-the-art methods. The saliency map of the learned features shows its capability of extracting the key spatial-temporal representation from the activity specifically in the time domain. The evaluation of the inference latency on three edge computing devices also presents the advance of the proposed network regards computational efficiency, which is crucial for real-time applications.

5.5 Reference

[1] Đ. Petrović, R. Mijailović, and D. Pešić, “Traffic Accidents with Autonomous Vehicles: Type of Collisions, Manoeuvres and Errors of Conventional Vehicles’ Drivers,” *Transp. Res. Procedia*, vol. 45, no. 2019, pp. 161–168, 2020, doi: 10.1016/j.trpro.2020.03.003.

[2] “Automated Driving Systems: A Vision for Safety 2.0,” Feb. 2017. [Online]. Available: https://www.nhtsa.gov/sites/nhtsa.gov/files/documents/13069a-ads2.0_090617_v9a_tag.pdf.

[3] B. W. Smith, “SAE levels of driving automation,” *Cent. Internet Soc. Stanford Law Sch.*, p. 1, 2014.

[4] J. C. F. de Winter, R. Happee, M. H. Martens, and N. A. Stanton, “Effects of adaptive cruise control and highly automated driving on workload and situation awareness: A review of the empirical evidence,” *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 27, no. PB, pp. 196–217, Nov. 2014, doi: 10.1016/j.trf.2014.06.016.

[5] S. H. Yoon and Y. G. Ji, “Non-driving-related tasks, workload, and takeover performance in highly automated driving contexts,” *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 60, pp. 620–631, Jan. 2019, doi: 10.1016/j.trf.2018.11.015.

[6] Y. Song, M. V. Chitturi, and D. A. Noyce, “Automated vehicle crash sequences: Patterns and potential uses in safety testing,” *Accid. Anal. Prev.*, vol. 153, p. 106017, Apr. 2021, doi: 10.1016/j.aap.2021.106017.

[7] B. Pfleging, M. Rang, and N. Broy, “Investigating user needs for non-driving-related activities during automated driving,” in *Proceedings of the 15th International Conference on Mobile and Ubiquitous Multimedia*, Dec. 2016, vol. 21, no. 1, pp. 91–99, doi: 10.1145/3012709.3012735.

[8] O. A. Osman, M. Hajij, S. Karbalaieali, and S. Ishak, “A hierarchical machine learning classification approach for secondary task identification from observed driving behavior data,” *Accid. Anal. Prev.*, vol. 123, no.

November 2018, pp. 274–281, Feb. 2019, doi: 10.1016/j.aap.2018.12.005.

[9] A. Aksjonov, P. Nedoma, V. Vodovozov, E. Petlenkov, and M. Herrmann, “Detection and Evaluation of Driver Distraction Using Machine Learning and Fuzzy Logic,” *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2048–2059, Jun. 2019, doi: 10.1109/TITS.2018.2857222.

[10] S. H. Kwon and M. Y. Kim, “Selective attentional point-tracking through a head-mounted stereo gaze tracker based on trinocular epipolar geometry,” in *2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, May 2015, vol. 2015-July, pp. 1617–1621, doi: 10.1109/I2MTC.2015.7151521.

[11] A. Kar and P. Corcoran, “A Review and Analysis of Eye-Gaze Estimation Systems, Algorithms and Performance Evaluation Methods in Consumer Platforms,” *IEEE Access*, vol. 5, pp. 16495–16519, 2017, doi: 10.1109/ACCESS.2017.2735633.

[12] H. Almahasneh, W.-T. Chooi, N. Kamel, and A. S. Malik, “Deep in thought while driving: An EEG study on drivers’ cognitive distraction,” *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 26, pp. 218–226, Sep. 2014, doi: 10.1016/j.trf.2014.08.001.

[13] A. Chaudhuri and A. Routray, “Driver Fatigue Detection Through Chaotic Entropy Analysis of Cortical Sources Obtained From Scalp EEG Signals,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 185–198, Jan. 2020, doi: 10.1109/TITS.2018.2890332.

[14] D. Jing, D. Liu, S. Zhang, and Z. Guo, “Fatigue driving detection method based on EEG analysis in low-voltage and hypoxia plateau environment,” *Int. J. Transp. Sci. Technol.*, vol. 9, no. 4, pp. 366–376, Dec. 2020, doi: 10.1016/j.ijtst.2020.03.008.

[15] M. Zhao, G. Beurier, H. Wang, and X. Wang, “Driver posture monitoring in highly automated vehicles using pressure measurement,” *Traffic Inj. Prev.*, vol. 22, no. 4, pp. 278–283, 2021, doi: 10.1080/15389588.2021.1892087.

[16] L. Yang, K. Dong, Y. Ding, J. Brighton, Z. Zhan, and Y. Zhao, “Recognition of visual-related non-driving activities using a dual-camera monitoring system,” *Pattern Recognit.*, vol. 116, p. 107955, Aug. 2021, doi: 10.1016/j.patcog.2021.107955.

[17] Y. Xing *et al.*, “Identification and Analysis of Driver Postures for In-Vehicle Driving Activities and Secondary Tasks Recognition,” *IEEE Trans. Comput. Soc. Syst.*, vol. 5, no. 1, pp. 95–108, 2018, doi: 10.1109/TCSS.2017.2766884.

[18] M. Martin, J. Popp, M. Anneken, M. Voit, and R. Stiefelhagen, “Body Pose and Context Information for Driver Secondary Task Detection,” in *2018 IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2018, vol. 5, no. 1, pp. 2015–2021, doi: 10.1109/IVS.2018.8500523.

[19] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, and F.-Y. Wang, “Driver Activity Recognition for Intelligent Vehicles: A Deep Learning Approach,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5379–5390, Jun. 2019, doi: 10.1109/TVT.2019.2908425.

[20] L. Yang *et al.*, “A refined non-driving activity classification using a two-stream convolutional neural network,” *IEEE Sens. J.*, vol. 21, no. 14, pp. 1–1, 2020, doi: 10.1109/JSEN.2020.3005810.

[21] H. M. Eraqi, Y. Abouelnaga, M. H. Saad, and M. N. Moustafa, “Driver Distraction Identification with an Ensemble of Convolutional Neural Networks,” *J. Adv. Transp.*, vol. 2019, pp. 1–12, Feb. 2019, doi: 10.1155/2019/4125865.

[22] L. Yang, X. Shan, C. Lv, J. Brighton, and Y. Zhao, “Learning spatio-temporal representations with a dual-stream 3D residual network for non-driving activity recognition,” *IEEE Trans. Ind. Electron.*, vol. 0046, no. c, pp. 1–1, 2021, doi: 10.1109/TIE.2021.3099254.

[23] F. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions,” in *2017 IEEE Conference on Computer Vision and Pattern*

Recognition (CVPR), Jul. 2017, vol. 2017-Janua, pp. 1800–1807, doi: 10.1109/CVPR.2017.195.

[24] L. Kaiser, A. N. Gomez, and F. Chollet, “Depthwise Separable Convolutions for Neural Machine Translation,” *Iclr*, pp. 1–10, Jun. 2017, [Online]. Available: http://arxiv.org/abs/1706.03059.

[25] A. G. Howard *et al.*, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” Apr. 2017, [Online]. Available: http://arxiv.org/abs/1704.04861.

[26] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[27] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[28] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning Spatiotemporal Features with 3D Convolutional Networks,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, vol. 2015 Inter, pp. 4489–4497, doi: 10.1109/ICCV.2015.510.

[29] K. Hara, H. Kataoka, and Y. Satoh, “Learning Spatio-Temporal Features with 3D Residual Networks for Action Recognition,” in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, Oct. 2017, vol. 2018-Janua, pp. 3154–3160, doi: 10.1109/ICCVW.2017.373.

[30] S. Ji, W. Xu, M. Yang, and K. Yu, “3D Convolutional Neural Networks for Human Action Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013, doi: 10.1109/TPAMI.2012.59.

[31] Z. Zheng, G. An, D. Wu, and Q. Ruan, “Global and Local Knowledge-Aware Attention Network for Action Recognition,” *IEEE Trans. Neural Networks*

Learn. Syst., vol. 32, no. 1, pp. 334–347, Jan. 2021, doi: 10.1109/TNNLS.2020.2978613.

[32] Z. Qiu, T. Yao, and T. Mei, “Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, vol. 2017-Octob, pp. 5534–5542, doi: 10.1109/ICCV.2017.590.

[33] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, “A Closer Look at Spatiotemporal Convolutions for Action Recognition,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6450–6459, doi: 10.1109/CVPR.2018.00675.

[34] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, “Squeeze-and-Excitation Networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020, doi: 10.1109/TPAMI.2019.2913372.

[35] X. Zhang, X. Zhou, M. Lin, and J. Sun, “ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6848–6856, doi: 10.1109/CVPR.2018.00716.

[36] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, “ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11218 LNCS, pp. 122–138, Jul. 2018, doi: 10.1007/978-3-030-01264-9_8.

[37] A. Howard *et al.*, “Searching for MobileNetV3,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, vol. 2019-Octob, pp. 1314–1324, doi: 10.1109/ICCV.2019.00140.

6 Impact analysis of NDRAs in take-over process

This chapter is based on the published conference paper*: L. Yang, M. B. Semiromi, D. Auger, A. Dmitruk, J. Brighton, and Y. Zhao, "The implication of non-driving activities on situation awareness and take-over performance in level 3 automation," in IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, Oct. 2020, vol. 2020-Octob, pp. 5075–5080, doi: 10.1109/IECON43393.2020.9254533.*

6.1 Introduction

A highly automated driving vehicle could free the driver's eye and hand from controlling the vehicle in some driving scenarios. It could encourage the driver to engage in some non-driving related activities (NDRAs) [1], [2]. However, fully automated driving has not been achieved yet. Automated driving vehicles cannot provide an appropriate response for every driving scenario, which is a potential safety risk and the main concern of the current automated driving system [3]. According to the SAE (J3016) Automation Levels [4], in Level 3 automation, the driver only needs to control the vehicle when the intervene is requested, which means the driver could engage some NDRAs rather than pay full attention to driving under the automated driving mode. Since the engagement of NDRAs could reduce the driver's situation awareness and attention [5]–[7], it is of great importance to evaluate its impact on the take-over performance to achieve a safe and smooth control transition.

Researches suggested that the sufficient take-over interval for drivers should be 5 to 8s [7], [8]. It is affected by different factors such as driver's state including age, gender, driving experience [9], [10], the complexity of the driving scenario [11]–[13], the modality of the take-over request [8], [14], [15] and the NDRAs that drivers engage with [8], [16]. The impact of diverse NDRAs on take-over performance has been widely researched in recent years. Yooh *et al.* [8] investigated the driver's take-over performance with 3 types of NDRA, which are phone conversation, smartphone interaction, and video watching tasks, while Zeeb *et al*. [1] examined the impact of writing an email, reading news, and

watching a video clip. Results from both studies suggested that the NDRA engagement can significantly influence the take-over quality based on the statistical analysis. One of the limitations of existing studies [17], [18] is that NDRAs were investigated specifically and independently, which limits the extendibility of the driver monitoring or take-over assistance system. When considering a new NDRA, such a system needs to conduct the evaluation process again to investigate its impact. There is a lack of a systematic method to group or categorise NDRAs which could have a similar level of impact on the take-over performance. On the other hand, the existing literature of NDRA's impact is normally from the perspective of the driver's workload [3], [7], [19]. The situation awareness before take-over is also considered as a crucial factor of safe take-over transition but has not been discussed associated with NDRAs [20]. There is a knowledge gap in the implication of situation awareness on the take-over process.

The existing literature has claimed that the driver's take-over performance is affected by the type of NDRAs. For instance, visual related activities tend to take a longer reaction time than auditory related activities [21]. However, the number of evaluated NDRAs is limited. Following the survey made by Sivak and Schoettle [22], the common NDRAs are *reading*, *texting*, *working*, *watching movies* and *playing games*. In this study, we picked 4 types of visual-related NDRA which are *playing games, answering questionnaires, watching videos* and *reading news* and further evaluated them on the same device which is a tablet. Based on the way of interaction between human and object, the NDRAs are divided into 2 groups, which are *active interaction mode* and *passive interaction mode*. *Playing games* and *answering questionnaires* can be considered as active interaction mode since the driver and the object respond to each other's action over time during the engagement. However, under *passive interaction mode* like *reading news* or *watching movies*, the driver only receives information passively. This study hypothesises that the workload and demanded attention are different between these two modes. Furthermore, compared with the *passive mode* NDRAs, the *active mode* NDRAs could result in a more negative impact on the driver's take-over performance.

This chapter investigates the implication of NDRAs in different interaction modes on the take-over performance in level 3 automation. Furthermore, the driver's behaviour has been recorded including hand and head movement, which is used to evaluate the driver's road-checking behaviour, which is considered as a factor that reflects the driver's situation awareness. Its motivation associated with NDRAs has been inferred. To ensure a safe take-over transition, haptic feedback has been added to the steering wheel. The haptic feedback in the Human-Machine Interface (HMI) design for the take-over process has been widely researched [23]–[25], specifically implemented on the steering wheel [26]–[28]. In this study, the effectiveness and impact of haptic feedback in take-over performance are also evaluated. The vehicle setting and the experiment design are introduced in Section 6.2. In Section 6.3, the driver's road-checking behaviour and take-over performance of each NDRAs are evaluated and discussed at both group and individual levels. Discussion and conclusion are given in Section 6.4.

6.2 Methodology

6.2.1 Take-over concept

The design of the take-over process in a trial is illustrated in Figure 6-1. During a trial, the vehicle was driving automatically initially while the participant was

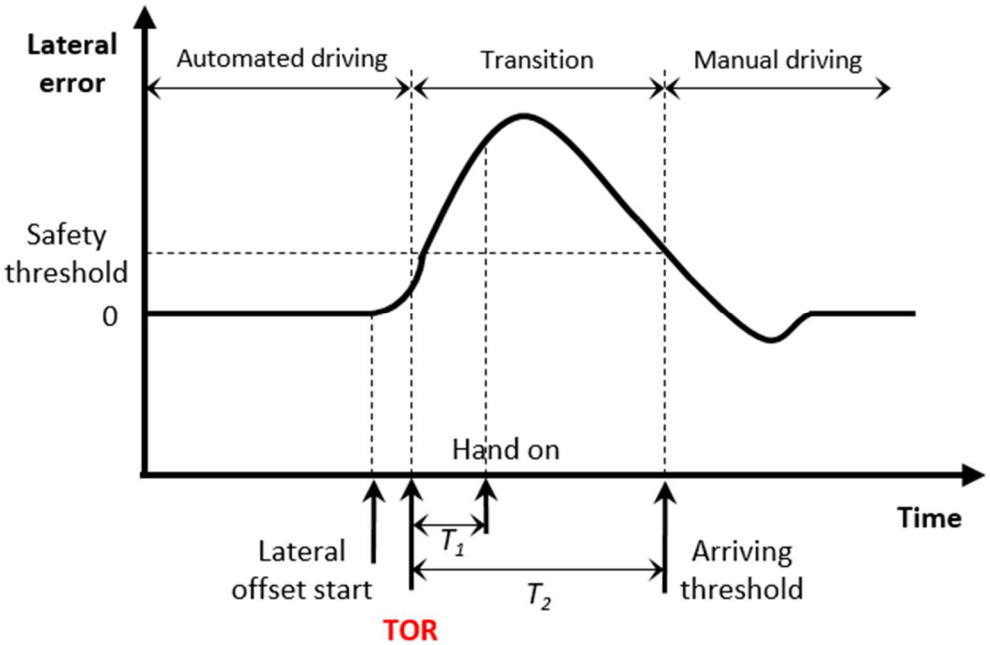


Figure 6-1 Concept of the take-over process

required to do a type of NDRA or checking the road. Then the take-over process started after a lateral offset was implemented to the vehicle. The lateral error is defined as the distance between the vehicle position and the closest point on the path. After a lateral offset was implemented, the vehicle is in an improper position of the road, an acoustic signal as a take-over request (TOR) was then given to the participant. The participant was requested to take control of the vehicle and bring it back to the right position. In Figure 6-1, T_1 indicates the time needed for the driver to put her/his hand on the steering wheel. To achieve a safe and smooth take-over transition, a haptic torque was implemented to help the driver and guide the vehicle to the reference route. The haptic torque was engaged as soon as the driver applies torque to the wheel and gradually fades away. After the lateral error achieves the maximum value, the vehicle will return to the reference route. A threshold of the safety distance is defined, which indicates the control transition is finished and the driver could achieve a safe manual driving afterwards. In this study, the threshold was set as 0.7m, which is the maximum lateral error to keep the vehicle inside the lane. In Figure 6-1, T_2 refers to the time needed from TOR to the time when the vehicle arrives at the threshold, which is considered as a criterion to evaluate the take-over performance in this study.

6.2.2 Experiment setup

6.2.2.1 Vehicle Modification

The vehicle used for the experiments was an instrumented Landrover Discovery 5. The car was modified to accommodate both autonomous and human driving. An electric motor, operating on the steering column, was used for steering and another electric motor was used to control the throttle pedal position. Braking was modified using a pneumatic actuator on the brake pedal. To ensure safety, a steering wheel and a set of pedals were added in the back seat, which allows a safety driver to intervene and override the autonomous system. For path following, the pure pursuit algorithm was used to generate the reference steering angle. The rear steering wheel was controlled using the reference steering angles and the front wheel follows the rear wheel.

6.2.2.2 Participants

A total of 16 participants (14 male and 2 female) from Cranfield University were recruited for this experiment. The participants' age is in a range from 24 to 30. They were required to hold a valid UK driving license while they have no driving experience with automated vehicles.

6.2.2.3 NDRAs

Four types of NDRA are investigated in this study, which include *reading news, watching videos, playing games* and *answering questionnaires* using a tablet. For *reading*, the participant was required to read some articles from BBC News. For *watching videos*, the participant was asked to watch Youtube videos. Temple Run was used as the target game for the game engagement. For the NDRA of *answering questionnaires*, the participant was required to complete a questionnaire, which comprises some objective and subjective questions about this experiment. In the experiment, there were 7 trials per participant. It includes 4 trials for 4 types of NDRA respectively and 1 trial without NDRA (watching road).

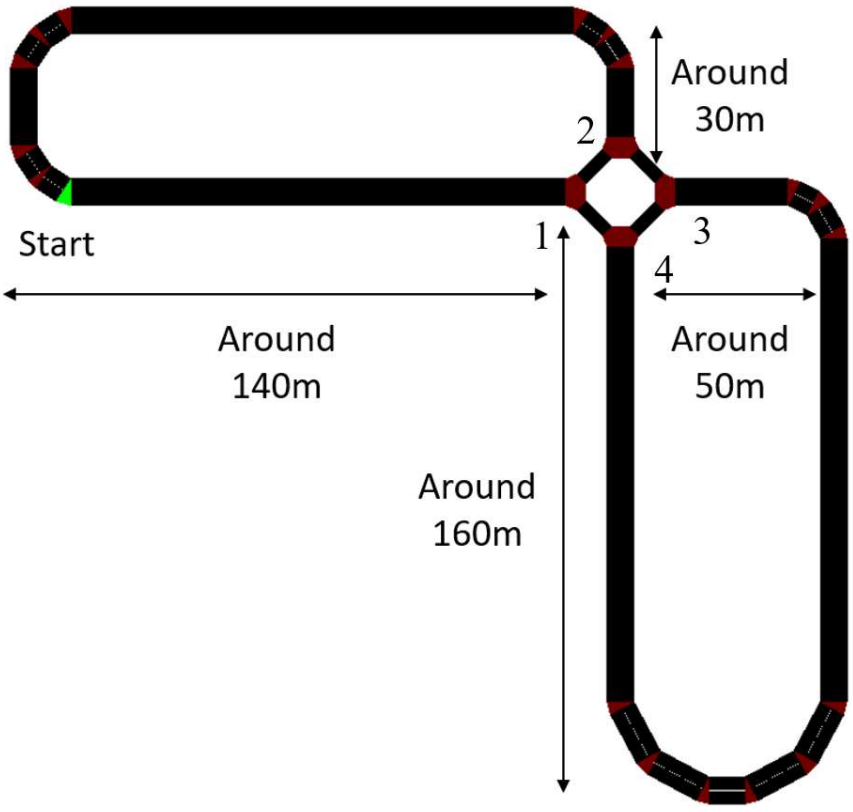


Figure 6-2 Sketch map of the track

For the remaining 2 trials, 2 activities were randomly selected from the 5 activities mentioned above. The order of each activity was randomly selected to reduce the bias.

6.2.2.4 Track and Take-over Scenarios

The testing track is a two-lane road with a mini-roundabout, as shown in Figure 6-2. The start point is highlighted with green colour. In the odd loop, the vehicle enters from exit 1 into the mini-roundabout and leaves from exit 3. Then it enters from exit 4 and leaves from exit 2. In the even loop, the vehicle enters from exit 1 and leaves the mini-roundabout from exit 4. Then it enters into the roundabout from exit 3 and leaves from exit 2. The TOR signal was issued at specific points on the track to avoid the area around the mini roundabout for safety concerns. The lateral offset was set as 1.5m with a small variation in the real trial. The maximum speed of the vehicle was set as 30 mph. The interval between TORs was randomly selected from the range of 5 to 9 minutes.

6.2.3 Data Acquisition

An OXTS RT1003 with RTK GPS was used for positioning and a dSPACE Microautobox I was used as an onboard computer. The RT1003 system provides the global vehicle position with an accuracy of 2cm and the heading angle with an accuracy of less than 1 degree. The data of vehicle status were recorded in the Micorautobox I at a sampling rate of 1kHz. The data include driver steering torque, autonomous steering torque, vehicle position and heading, vehicle velocity, steering angle and take-over signal. The path was recorded beforehand at 1kHz.

Driver's hand-on-wheel time (t_1) was defined as the moment that the driver's applied torque passes a certain threshold. The threshold was experimentally determined to avoid false take-over detection due to sensor noise. An instance of the driver's torque during a take-over process is shown in the top plot of Figure 6-3. The corresponding vehicle route is presented in the bottom plot of Figure 6-3.

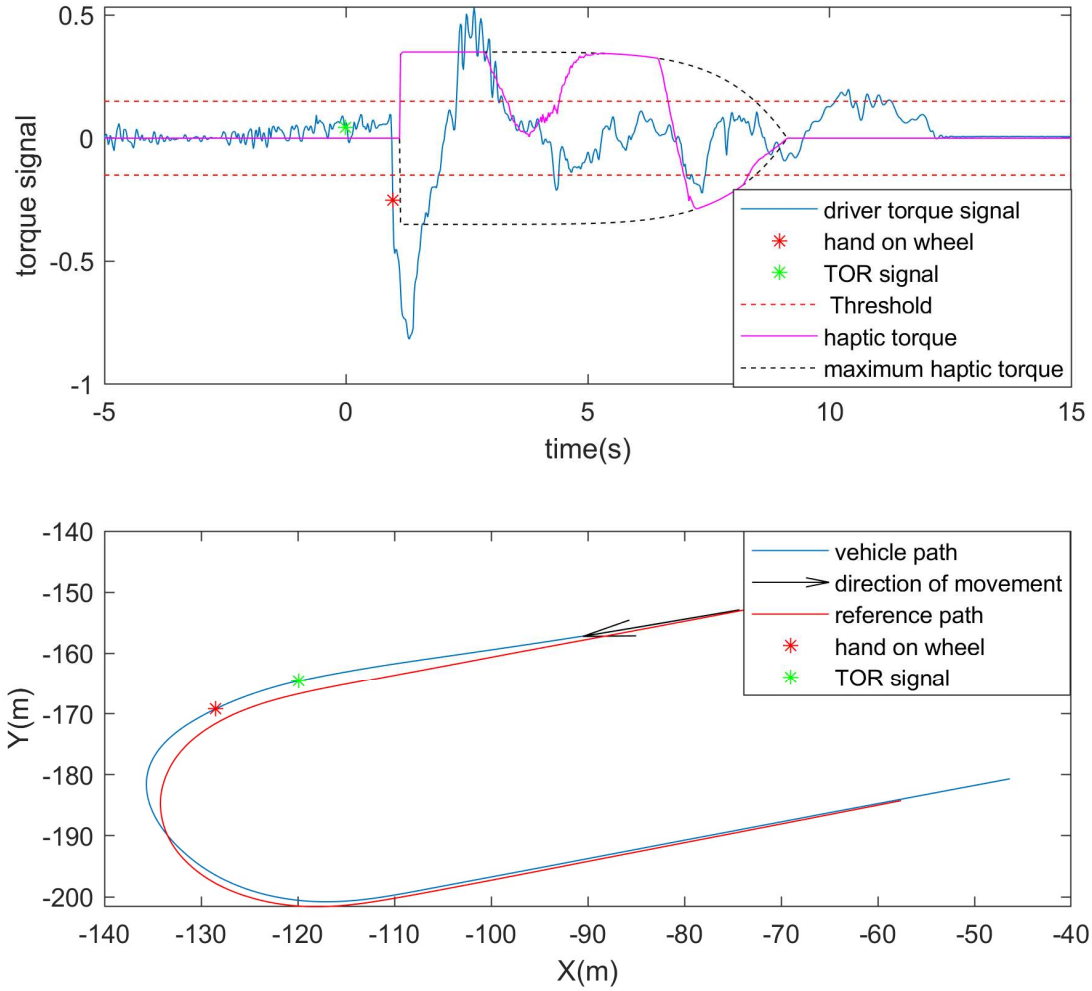


Figure 6-3 Top plot presents the driver's torque and the haptic torque for 1 instance. Bottom plot presents the corresponding vehicle movement in the track

After the driver takes control of the steering wheel, the vehicle provides haptic cues to the driver, in the form of torque on the steering wheel, to increase the driver's awareness of the environment. The haptic decays over a certain amount of time and eventually reaches 0 to give the driver full control. The value of the torque is calculated using

$$\tau_{haptic}(t) = K_t(t)K_p(\delta - \delta_{ref}) \quad \textbf{(6-1)}$$

where δ is the vehicle steering angle; δ_{ref} is the reference steering angle calculated by the path following algorithm; K_p is a constant gain and $K_t(t)$ is a decaying gain which is a function of time starting from 1 and reaching to 0 at the

end of the take-over period. The decaying profile is shown in the top plot of Figure 6-3. The decaying duration chosen for this experiment was 8 seconds. The torque value is normalised between -1 and 1, where 1 indicates the maximum torque of the electric motor in one direction and -1 indicates the maximum torque in another direction. The maximum amplitude of the torque was a tuning parameter. Each participant tried two of three pre-set values: 0.35, 0.45, 0.55.

There were 2 cameras (Garmin Virb Action Camera) employed to monitor the driver's behaviour during the experiment. The resolution of both cameras was set as 1920 × 1440 pixels and images were sampled at 24 frames per second (fps). As shown in Figure 6-4, one camera (Camera 1) was located in the right bottom of the windscreen and faced to the driver's head, which is used to detect whether the driver is engaging in NDRAs or checking the road. Another one (Camera 2) was mounted on the roof of the vehicle between two front seats to record the driver's hand movement engaging with the tablet or steering wheel. The following analysis of situation awareness is based on the process of the recorded videos. The video clip captured from Camera 1 was used to evaluate if the driver conducts road-checking behaviours. The inferred motivation was manually labelled based on the videos recorded from Camera 2.

Figure 6-4 A illustration of the two cameras inside the vehicle

Table 6-1 Road-checking behaviour evaluation

NDRAs	Checking period (s)	Percentage of checking for corresponding motivation			
		Bumping	Approaching junctions	Breakpoint	Others
Watching videos	37.10	19.88%	52.05%	5.85%	22.22%
Reading news	51.64	16.78%	51.75%	7.69%	23.78%
Playing games	79.13	3.61%	26.50%	59.04%	10.84%
Answering questionnaires	123.00	18.18%	50.00%	13.64%	18.18%

6.3 Results

6.3.1 Road-checking behaviour analysis

The statistic results of driver road-checking behaviour for all participants are presented in Table 6-1. The checking period is calculated by the duration of the NDRA trial and the total number of checking behaviour in this trial. The motivation for road-checking behaviour is inferred by reviewing the videos from two cameras. *Bumping* refers to the vehicle vibration due to uneven road surface. For *approaching junctions*, the driver's glance is counted when approaching the roundabout and turning. *Breakpoint* indicates the road-checking behaviour due to a short break during the NDRA engagement. For instance, the driver sometimes checks the environment after she/he finishes watching a video clip or a round of game. *Other*s covers the road-checking behaviour without unclear motivation or regular road-checking.

It has been observed that the checking period is lowest (37.1s) when the driver was *watching videos*. For this NDRA, the main motivations of road-checking are *Approaching junction* (52.05%) and *Bumping* (19.88%). *Reading news* has the second-lowest period (51.64s), where the proportion of motivations is similar to that of *watching videos. Answering questionnaires* has the least road-checking behaviour. Normally only once or twice in a trial. *Approaching junction* (50%) still

dominates the motivation. As one of the typical NDRAs under the *active interaction mode*, *playing games* has a relatively high road-checking period (79.13s), where *Breakpoint* (59.04%) dominates the motivation. The proportion of *Approaching junctions* and *Bumping* are 26.5% and 3.61%, respectively. Compared to the NDRAs in *active interaction mode*, the NDRAs in *passive interaction mode* leads to more frequent road-checking, which suggests drivers have more awareness for the situations of vehicle vibration, turning or slowing down when approaching junctions. These road-checking behaviours are important to ensure a safe transition if the take-over is required under these scenarios. The observation also suggests that the driver has a relatively low workload under *passive interaction mode*, which potentially leads to a smoother and better-quality take-over process. For the NDRAs under active interaction mode, the results show that the driver paid a high level of engagement on the activity, particularly *for answering questionnaires*, evident by much less frequent road-checking. For *playing games*, the road-checking normally happens during

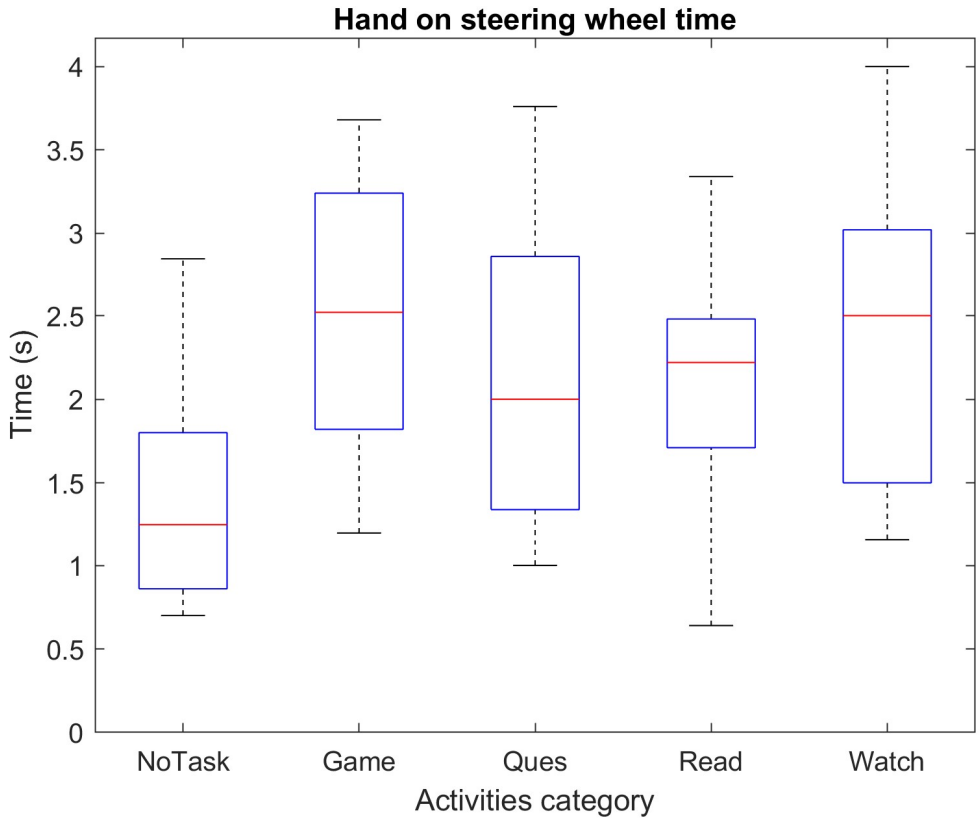


Figure 6-5 The hand-on-wheel time performance. NoTask refers to the performance in watching road trial

Breakpoint and the driver is not sensitive to the driving-situation change during a game. Therefore, for this type of NDRA, the driver is more difficult to complete a high-quality take-over transition due to lack of situation awareness.

6.3.2 Take-over performance

The driver's performance during the take-over process is presented in this section. The driver's hand-on-wheel time (T_1) is shown in Figure 6-5. As expected, no NDRA engagement achieved the shortest T_1 with an average value around 1.3s. For the selected 4 NDRAs, the average T_1 is in the range of 1.9-2.6s, which is more than double than without NDRA. Playing games seems to result in the longest T_1. From Figure 6-6, it can be observed the maximum lateral error for each activity is similar, which is in the range of 2 to 3.5m. In most of the trials, after receiving the TOR signal, the driver can obtain the control of the vehicle and prevent the situation from getting worse within a 3.5m lateral error. However, NDRA engagement affects the driver's controlling performance after the vehicle achieving the maximum lateral error. It can be seen from Figure 6-7 that the time

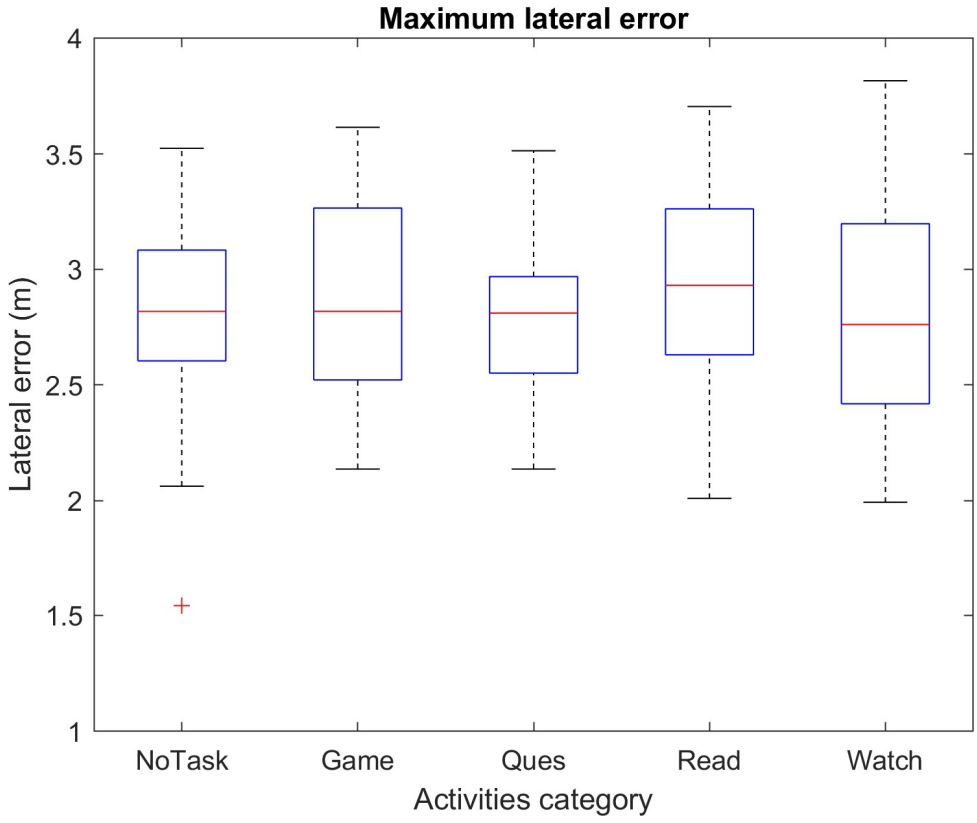


Figure 6-6 Maximum lateral error achieving

Table 6-2 Time to threshold for all activities

Time to Threshold	Activities				
	No task	Watch	Read	Ques	Game
Mean (s)	4.16	4.74	4.96	5.45	5.43
Standard deviation (s)	0.67	1.12	0.87	1.23	1.14

needed to arrive the safe position without NDRA engagement (T_2) is around 4.16s, while for all types of NDRA engagement T_2 is at least 0.5s more, which suggests that the vehicle could stay in a dangerous position for a longer time. Mean and standard deviation of T_2 for each activity are presented in Table 6-2. The mean values of *watching videos* and *reading news* (*passive interaction mode*) are 4.74s and 4.96s, representatively, which are higher than those of *answering questionnaires* and *playing games* (5.45s and 5.43s respectively) in *active interaction mode*. The standard deviation of the NDRAs is higher than the *NoTask*, which suggests higher individual differences of the take-over performance in NDRAs engagement. Through combining Figure 6-5 and Figure

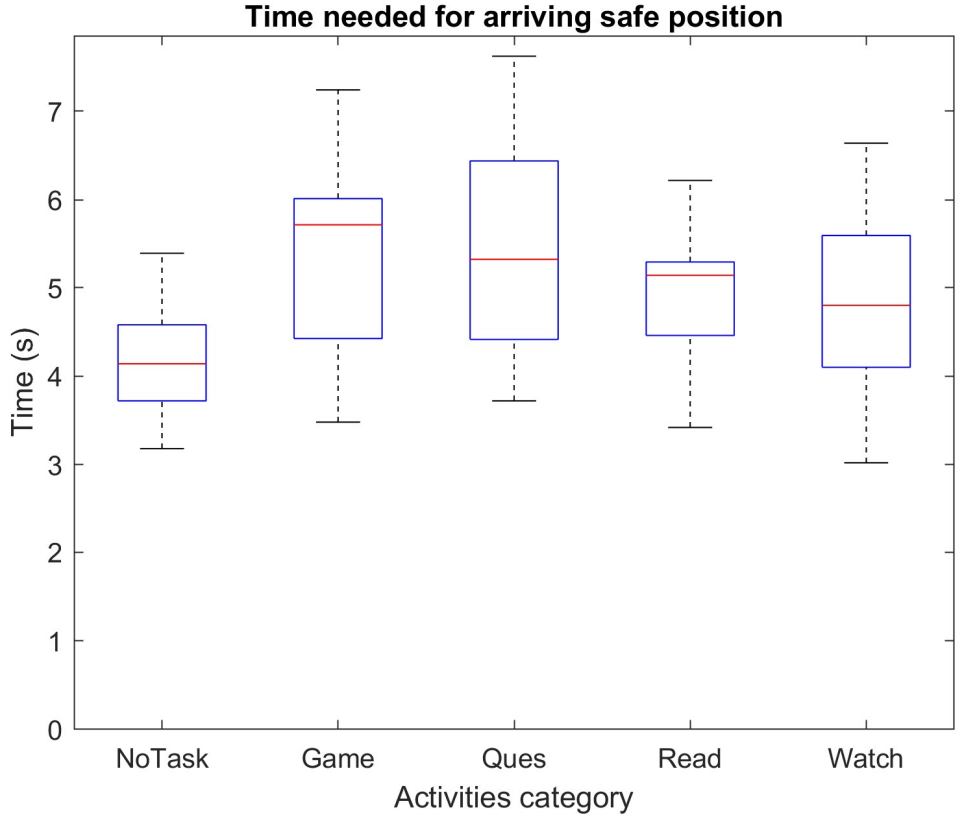


Figure 6-7 Time cost for the vehicle back to the safe position

6-7, it has been suggested that NDRAs in *active interaction mode* request more time to control the vehicle during the take-over process. The reason could be that for this type of NDRA the driver needs more time to develop the awareness of driving-environment after receiving the TOR signal and is more difficult to recover from the previous NDRA mentally.

The take-over performance of haptic feedback is presented in **Error! Reference source not found.**. For a low level of haptic torque, the mean value of T_2 is 5.32s, which is the lowest among all the evaluated levels. The standard deviation is 1.12s, which suggests that all the participants have higher tolerance on this level of haptic torque assistance. It can be seen that the increase of the torque level could result in the decrease of the mean value of T_2, which means a higher level of haptic torque could support the driver to reduce T_2 and improve their take-over performance. However, the standard deviation increases (1.55s for medium level and 1.32s for high level). It suggests that some of the participants could distrust and resist the higher level of haptic torque and take a longer T_2.

6.4 Conclusion

In level 3 automated driving, one of the most important challenges for driving safety is the take-over process. It is affected by many factors but dominated by the driver's state before take-over. This study investigated the implication of four selected NDRAs, grouped into *active and passive interaction modes*, on situation awareness during the NDRA engagement associated with its motivation and the following take-over performance. The approach of grouping aims to extend the application of this study on a wide range of NDRA. Furthermore, the effectiveness of steering wheel haptic assistance system for the take-over process has been evaluated.

From the situation awareness point of view, drivers always check the environment to ensure driving safety during the NDRAs engagement. Compared to the NDRAs in *active interaction mode*, the NDRAs in *passive interaction mode* leads to more frequent road-checking, which suggests drivers have more awareness for the situations of vehicle vibration, turning or slowing down when approaching

junctions. The motivation study also suggests that for the NDRAs in *active interaction mode* the driver is not sensitive to the driving situation change. Drivers should be warned when they engage with this kind of NDRA and do not check the road for a long period.

For the take-over process, the engagement of NDRAs could result in a negative effect. It has been observed that the type of the NDRAs could affect the driver's takeover performance. Specifically, the driver who engages with NDRAs in *active interaction mode* requests more time to achieve a safe take-over transition. Therefore, identifying the NDRAs type rather than detecting the driver's distraction could help to predict his/her takeover performance and determine the proper takeover strategy or modality for the current type of NDRAs if the intervention is requested. Moreover, haptic torque assistance could improve the take-over performance evidenced by decreasing T_2. However, a higher level of haptic torque could result in the driver's resistance.

In summary, the type of NDRA determines the level of demanded attention of the driver, which influences the situation awareness and take-over quality. The observed results also suggest that the take-over process could benefit from the high-frequency road-checking and haptic feedback assistance. The investigation of these factors helps us develop a deep understanding of the implication of human behaviour on the take-over performance, which could help for further take-over strategy and HMI design to achieve the safe control transition.

6.5 Reference

[1] K. Zeeb, A. Buchner, and M. Schrauf, "Is take-over time all that matters? The impact of visual-cognitive load on driver take-over quality after conditionally automated driving," *Accid. Anal. Prev.*, vol. 92, pp. 230–239, Jul. 2016, doi: 10.1016/j.aap.2016.04.002.

[2] L. Yang, K. Dong, A. J. Dmitruk, J. Brighton, and Y. Zhao, "A Dual-Cameras-Based Driver Gaze Mapping System With an Application on Non-Driving Activities Monitoring," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 10, pp. 4318–4327, Oct. 2020, doi: 10.1109/TITS.2019.2939676.

[3] D. Choi, T. Sato, T. Ando, T. Abe, M. Akamatsu, and S. Kitazaki, “Effects of cognitive and visual loads on driving performance after take-over request (TOR) in automated driving,” *Appl. Ergon.*, vol. 85, no. February, p. 103074, May 2020, doi: 10.1016/j.apergo.2020.103074.

[4] B. W. Smith, “SAE levels of driving automation,” *Cent. Internet Soc. Stanford Law Sch.*, p. 1, 2014.

[5] J. C. F. de Winter, R. Happee, M. H. Martens, and N. A. Stanton, “Effects of adaptive cruise control and highly automated driving on workload and situation awareness: A review of the empirical evidence,” *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 27, no. PB, pp. 196–217, Nov. 2014, doi: 10.1016/j.trf.2014.06.016.

[6] E. Dogan, M.-C. Rahal, R. Deborne, P. Delhomme, A. Kemeny, and J. Perrin, “Transition of control in a partially automated vehicle: Effects of anticipation and non-driving-related task involvement,” *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 46, pp. 205–215, Apr. 2017, doi: 10.1016/j.trf.2017.01.012.

[7] S. H. Yoon and Y. G. Ji, “Non-driving-related tasks, workload, and takeover performance in highly automated driving contexts,” *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 60, pp. 620–631, Jan. 2019, doi: 10.1016/j.trf.2018.11.015.

[8] S. H. Yoon, Y. W. Kim, and Y. G. Ji, “The effects of takeover request modalities on highly automated car control transitions,” *Accid. Anal. Prev.*, vol. 123, no. November 2018, pp. 150–158, Feb. 2019, doi: 10.1016/j.aap.2018.11.018.

[9] H. Clark and J. Feng, “Age differences in the takeover of vehicle control and engagement in non-driving-related activities in simulated driving with conditional automation,” *Accid. Anal. Prev.*, vol. 106, pp. 468–479, Sep. 2017, doi: 10.1016/j.aap.2016.08.027.

[10] S. Li, P. Blythe, W. Guo, and A. Namdeo, “Investigating the effects of age

and disengagement in driving on driver's takeover control performance in highly automated vehicles," *Transp. Plan. Technol.*, vol. 42, no. 5, pp. 470–497, Jul. 2019, doi: 10.1080/03081060.2019.1609221.

[11] J. Radlmayr, C. Gold, L. Lorenz, M. Farid, and K. Bengler, "How Traffic Situations and Non-Driving Related Tasks Affect the Take-Over Quality in Highly Automated Driving," *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 58, no. 1, pp. 2063–2067, Sep. 2014, doi: 10.1177/1541931214581434.

[12] M. S. L. Scharfe, K. Zeeb, and N. Russwinkel, "The Impact of Situational Complexity and Familiarity on Takeover Quality in Uncritical Highly Automated Driving Scenarios," *Information*, vol. 11, no. 2, p. 115, Feb. 2020, doi: 10.3390/info11020115.

[13] C. Wu, H. Wu, N. Lyu, and M. Zheng, "Take-Over Performance and Safety Analysis Under Different Scenarios and Secondary Tasks in Conditionally Automated Driving," *IEEE Access*, vol. 7, pp. 136924–136933, 2019, doi: 10.1109/ACCESS.2019.2914864.

[14] B. Wandtner, N. Schömig, and G. Schmidt, "Effects of Non-Driving Related Task Modalities on Takeover Performance in Highly Automated Driving," *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 60, no. 6, pp. 870–881, Sep. 2018, doi: 10.1177/0018720818768199.

[15] H. Jeong and Y. Liu, "Effects of non-driving-related-task modality and road geometry on eye movements, lane-keeping performance, and workload while driving," *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 60, pp. 157–171, Jan. 2019, doi: 10.1016/j.trf.2018.10.015.

[16] J. Kim, H.-S. Kim, W. Kim, and D. Yoon, "Take-over performance analysis depending on the drivers' non-driving secondary tasks in automated vehicles," in *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, Oct. 2018, pp. 1364–1366, doi: 10.1109/ICTC.2018.8539431.

[17] S. Petermeijer, F. Doubek, and J. de Winter, “Driver response times to auditory, visual, and tactile take-over requests: A simulator study with 101 participants,” in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2017, vol. 2017-Janua, pp. 1505–1510, doi: 10.1109/SMC.2017.8122827.

[18] H. Kim, W. Kim, J. Kim, and D. Yoon, “A Study on the Control Authority Transition Characteristics by Driver Information,” in *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, Dec. 2019, pp. 1562–1563, doi: 10.1109/CSCI49370.2019.00297.

[19] M. Bueno, E. Dogan, F. Hadj Selem, E. Monacelli, S. Boverie, and A. Guillaume, “How different mental workload levels affect the take-over control after automated driving,” in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov. 2016, pp. 2040–2045, doi: 10.1109/ITSC.2016.7795886.

[20] L. Petersen, L. Robert, X. J. Yang, and D. Tilbury, “Situational Awareness, Driver’s Trust in Automated Driving Systems and Secondary Task Performance,” *SAE Int. J. Connect. Autom. Veh.*, vol. 2, no. 2, pp. 12-02-02–0009, May 2019, doi: 10.4271/12-02-02-0009.

[21] B. Wandtner, G. Schmidt, N. Schoemig, and W. Kunde, “Non-driving related tasks in highly automated driving - Effects of task modalities and cognitive workload on take-over performance,” in *AmE 2018 - Automotive meets Electronics; 9th GMM-Symposium*, Mar. 2018, pp. 1–6.

[22] M. Sivak and B. Schoettle, “Motion Sickness in Self-Driving Vehicles,” no. April, 2015, [Online]. Available: https://deepblue.lib.umich.edu/handle/2027.42/111747.

[23] S. M. Petermeijer, D. A. Abbink, M. Mulder, and J. C. F. de Winter, “The Effect of Haptic Support Systems on Driver Performance: A Literature Survey,” *IEEE Trans. Haptics*, vol. 8, no. 4, pp. 467–479, Oct. 2015, doi: 10.1109/TOH.2015.2437871.

[24] J. Wan and C. Wu, “The Effects of Vibration Patterns of Take-Over Request and Non-Driving Tasks on Taking-Over Control of Automated Vehicles,” *Int. J. Human–Computer Interact.*, vol. 34, no. 11, pp. 987–998, Nov. 2018, doi: 10.1080/10447318.2017.1404778.

[25] C. Lv *et al.*, “Characterization of Driver Neuromuscular Dynamics for Human–Automation Collaboration Design of Automated Vehicles,” *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 6, pp. 2558–2567, Dec. 2018, doi: 10.1109/TMECH.2018.2812643.

[26] S. M. Petermeijer, J. C. F. de Winter, and K. J. Bengler, “Vibrotactile Displays: A Survey With a View on Highly Automated Driving,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 897–907, Apr. 2016, doi: 10.1109/TITS.2015.2494873.

[27] C. Lv *et al.*, “A Novel Control Framework of Haptic Take-Over System for Automated Vehicles,” in *2018 IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2018, vol. 2018-June, no. Iv, pp. 1596–1601, doi: 10.1109/IVS.2018.8500480.

[28] C. Lv *et al.*, “Human-Machine Collaboration for Automated Vehicles via an Intelligent Two-Phase Haptic Interface,” *arXiv*, Feb. 2020, [Online]. Available: http://arxiv.org/abs/2002.03597.

7 Overall discussion, conclusion, and future work

This chapter presents the key findings and the contribution to the knowledge of this research, discusses its implication on the industry and further gives the conclusion and future work of this research.

7.1 Research gaps filled

Research Gap 1: Lack of research on the DRAs recognition in the AVs and visual attention estimation:

To recognise the driver's DRAs engagement, this research utilised the driver's head movement, since when the driver is engaging in the DRAs, he/her always checks the road and the driving environment, while the NDRAs engagement requests the driver to focus on the object such as phone, tablet. Furthermore, the driver's gaze focus has been extracted through a gaze mapping system in Chapter 2. Such a system could locate the driver's visual attention, which can be used to recognise the visual attention-related NDRAs with the recognition of the object that the driver is interacting with. The head movement monitoring and the visual attention estimation can be used as an indicator to reflect the driver's situation awareness, which is important to be evaluated before the takeover process.

Research Gap 2: Lack of research on video-based driver's NDRAs recognition, specifically, high-similarity activities:

To differentiate the high-similarity activities, this research produced an NDRAs dataset, which includes 4 activities, *reading news, watching videos, playing games* and *answering questionnaires.* All of the activities are performed on a tablet. Unlike most of the existing research, which is usually image-based, this research proposed a 3D CNN based model that extracts the spatio-temporal features from the driver's behaviour in videos in Chapter 4. It could learn the representation of the movement pattern in the time domain.

Research Gap 3: Lack of the lightweight driver behaviour monitoring model for in-vehicle application and its implementation and evaluation on the edge computing device:

Unlike the accuracy-oriented studies in the field of action recognition, this research focuses on real-time driver behaviour monitoring and its in-vehicle implementation. In Chapter 5, a temporal attention-based lightweight 3D CNN module has been developed for this purpose. Its performance of the activity recognition in terms of accuracy and inference latency has been evaluated on the NVIDIA Jetson family in comparison with other state-of-the-art lightweight models.

Research Gap 4: Lack of research on the exploration of the high-level implication of NDRAs on the takeover process:

Finally, in Chapter 6, the research categories the NDRAs into 2 groups based on the interaction mode between the driver and the object employed for the engagement. Then investigates the impact of the NDRAs on the takeover process at the group and individual levels. The results prove that the engagement of the NDRAs could lead to a negative impact on the takeover process and also proves the feasibility of using this proposed category method to group the NDRAs with a similar level of impact.

7.2 Contribution to the knowledge

The contribution to the knowledge of this research can be divided into 3 parts: The exploration of the methods for the driver's behaviour characterisation, the optimisation of deep learning-based methods for NDRAs/DRAs recognition, and the evaluation of NDRAs impact on the takeover process.

The exploration of the method for the driver's behaviour characterisation:

- In this research, the pattern of the driver's behaviour in the vehicle is characterised by the driver's head and hand movement. A two-feed system based on these features has been proposed to monitor the driver's activity engagement in the AVs.

- A novel two-camera based system has been developed to estimate the driver's visual attention. One camera is used to extract the driver's facial and gaze features while another camera visualises the estimated gaze. The estimated gaze is used to detect the driver's NDRAs engagement and identify the NDRAs with the recognised object. Comparing with other deep learning-based NDRAs recognition methods, such a method converts the classification problem into a recognition problem, which is more transparent and interpretable.
- Using a two-stream CNN model to recognise the NDRAs, for which the spatial stream differentiates the objects from the raw image, the temporal stream employs the optical flow to represent the motion between images. Furthermore, an ROI model is designed based on the human-object interaction, which could reduce the noise of the optical flow and increase the processing speed.

The optimisation of deep learning-based methods for NDRAs/DRAs recognition:

- This research proposed a dual-stream 3D residual network, named DS3D ResNet, which is able to enhance the learning of spatio-temporal representation and improve the activity recognition performance. Specifically, a parallel 2-stream structure is introduced to focus on the learning of short-time spatial representation and small-region temporal representation of the activity. Moreover, the saliency map of the hidden layer is employed to present the semantic correlation of the learned representation.
- This research also developed a lightweight temporal attention-based module for CNN. Such a module factorises the conventional 3D convolution as a spatial convolution and a temporal attention function. With the implemented channel weighting function, the proposed module could learn the spatio-temporal representation in an efficient way. The performance, especially, the inference latency of the proposed model and other state-of-the-art models have been evaluated on the NVIDIA Jetson family.

The evaluation of NDRAs impact on the takeover process:

- This research proposed a category method for the NDRAs based on the interaction mode. The results show that the NDRAs in the same interaction mode have a similar level of impact on the takeover performance. It also suggests that the *active* NDRAs engagement takes more time for the driver to complete the control transition, compare with the passive NDRAs engagement. Such a method could predict the level of the impact for unevaluated NDRAs.
- In this research, driver's road checking behaviour during the NDRAs engagement is considered as an important factor that reflects the driver's situation awareness before the takeover process. This research highlights the importance of road checking behaviour. The behaviour and its motivation can be used to evaluate the level of the driver's NDRAs engagement and the driver's mental demands of the NDRAs

7.3 Real world application or Impact on the industry

This research investigates different approaches and uses the driver's head and hand movement to characterise the driver's behaviour and demonstrates the effectiveness of the two-feed system for the monitoring of the driver activity engagement in the vehicle cabin. Specifically, by using the head movement information, a gaze mapping system has been developed to visualise the driver's visual attention. Such a system can not only be used to detect the NDRAs engagement and classify the visual related NDRAs but also be used for the driver distraction detection for the human-driven vehicle. Compare with other facial information based driver distraction detection methods, the advantage of this system is that it can locate the driver's visual attention and provide the details of the activity that distracts the driver. For instance, the impact level on the driving for NDRAs engagement distraction and visual contact with passengers are different. Based on the distraction type and level, it can be further determined if the driver needs to be warned to secure driving safety. However, monitoring the driver's facial information could be controversial. It could involve some concerns about privacy and ethics. Compare with the facial information-based methods,

the hand gesture-based NDRAs recognition methods will have less concern in this field. Even though the NDRAs can be classified based on hand gestures, such a method still cannot determine whether the driver is engaging in the NDRAs without visual attention, since the driver could hold some device while checking the road or sleeping without any hand gesture. Moreover, Tesla attempted to use the driver-applied steering wheel torque as an indicator to monitor the driver's DRAs engagement, which has been proved as an ineffective surrogate measure in the report of Tesla's fatality [1]. Therefore, using the driver's facial information or visual attention to measure the DRAs engagement is straightforward and effective. The way of collecting, processing and using this kind of information in the vehicle could be further discussed and investigated. To sum up, head or hand movement information has been used to identify some types of NDRAs individually in the existing research. However, the limitation of using these features independently is obvious. The head movement cannot be used to refine the NDRAs classification, and the hand movement is not able to indicate the driver's attention. Therefore, combining both features to recognise the driver's behaviour by using a two-feed system is crucial in real driver monitoring applications. Furthermore, the evaluated and proposed approaches for driver behaviour monitoring is not only limited to level 3 automated driving vehicle. It is also important to monitor the driver's behaviour at all levels of automation. For the lower automation level such as level 1 and level 2, it can be used to detect the distraction (level 1) and recognise the secondary task (level 2). The driver's visual attention plays a key role in these levels. In the higher automation level, it gives the driver more tolerance to engage some NDRAs. Monitoring the driver's behaviour could help the vehicle to determine whether the vehicle will give the control back to the driver if the driver requests (level 4). For instance, the driver requests to take over the vehicle, however, he/she is in an unsuitable condition for driving like drunk. The recognised state of the driver could support the vehicle to make the right decision. In level 5 automation, the driver cannot control the vehicle. Even though, the monitoring of the driver or passenger is also necessary to predict the hazard inside the vehicle cabin or the effect of their behaviour could make to the automated driving. Therefore, the monitoring of the driver's behaviour

is of great importance in all levels of driving automation, which should be further implemented in the whole automotive industry.

From the algorithm perspective for the action recognition, this research developed two methods, which are the DS3D model for spatio-temporal representation learning and a lightweight model for the edge computing device based on the attention mechanisms. This proposed DS3D model has demonstrated its capability of extracting spatio-temporal features from the driver's behaviour during the NDRAs engagement. This model not only can be used for this purpose but also be used to solve the general action recognition problem. The short-time spatial stream of the model focuses on the spatial change in a short duration, which is normally the feature of the high-frequency movement while the small-region temporal stream shows the capability of capturing long-term memory from the low-frequency movement. Comparing with other conventional one-stream models, the proposed model adopts the two-stream, which extracts different types of features from the activity. It provides a new thought for the model design in the field of action recognition or classification. The proposed lightweight temporal attention-based module factorised the conventional 3D convolution. It employs the 2D convolution for spatial features and introduces the attention mechanisms into the time domain. The proposed module enhanced the model's capability of learning the temporal attention from the motion with a limited computational cost. Moreover, the module is independent and end-to-end trainable, which can be used as a plugin module for the existing 3D CNN backbone. It could have wide applications in the field of spatio-temporal feature extraction.

For the evaluation of the NDRAs impact on the takeover process, this research proposed a category method to group the activities based on the manner that the information is transferred during the visual attention-related activities. The *active interaction mode* NDRAs requires the bidirectional information transmission between the driver and the object during the NDRAs engagement, while in the *passive interaction mode* NDRAs engagement, the driver passively receives the information from the object. During the *active* NDRAs engagement, the intensive

information transmission normally costs high mental demands for the driver, which reduces his/her road-checking behaviour and situation awareness. It will increase the time needed for completing the takeover process. The *active* NDRAs are not limited to the 2 classes evaluated in this research. The other common *active* NDRAs could be: working on a laptop, chatting via a phone, etc. For the *passive* NDRAs engagement, the driver normally pays less attention and performs more road checking behaviour to sense the environment. This research also found the importance of road-checking behaviour during the NDRAs engagement. The results suggest that the frequently road-checking behaviour could reduce the time needed for completing the takeover process. It should be noted that the evaluated NDRAs are visual attention-related. Some other NDRAs such as eating, drinking, communicating with passengers, etc, are not evaluated. Unlike the visual attention-related NDRAs, during the engagement of these NDRAs, the driver normally checks the road all the time and has a good awareness of the driving environment. Such NDRAs are not suitable for the proposed category method. Furthermore, the finding of the research in terms of the NDRAs impact on the takeover performance is crucial for the design of the takeover strategy and modality in conditional driving automation. In the current research on the takeover strategy and modality, the investigated TOR signals are mainly visual, auditory and tactile [2]. The *active* NDRAs engagement is proved costs more mental demands of the driver. Therefore, an efficient way to convey the takeover message to the driver is to combine the 3 types of signals [3]. Since the driver could lack situation awareness before the takeover, the detected hazard could be presented on the windscreen to help the decision making. However, for *passive* NDRAs engagement, the TOR signal could be gentle and the strategy should be simple. The complicated strategy and strong TOR signal such as vibration (tactile signals) could confuse the driver and make him/her nervous, which could lead to a negative effect on the takeover process.

7.4 Conclusion

This thesis explores the way of characterising the driver's behaviour during the DRAs and NDRAs engagement, particularly the latter one, with the computer

vision and AI-based approach in the AVs. Specifically, the driver's head movement or the facial information can be considered as the source that provides the driver's attention information. With the recognised in-vehicle environment, the located attention can be used to infer the driver's intention, either the NDRAs engagement with the object like a phone, tablet or the DRAs engagement with the object like wing mirrors, road outside. The driver's hand movement is employed to further differentiate the activities since the motion patterns in the time domain are different during the activity engagement. Based on these features, this research proposes a 2-feed system architecture to monitor the driver's behaviour. For the modelling of activity classification and feature extraction, this research proposes a Dual-stream 3D CNN, which learns the short-time spatial representations (high-frequency interaction) and the small-region temporal representations (low-frequency interaction) from the motion during the activity engagement. Moreover, this research evaluates the performance of the state-of-the-art efficient CNN-based action classification models that can be used to recognise the NDRAs/DRAs on the edge computing device for the real-time in-vehicle application. A temporal attention-based lightweight 3D CNN module has been proposed to enhance the model's capability of spatio-temporal representation learning, especially in the time domain, with a relatively low computational cost. The proposed 2-feed system architecture and the CNN based model have been proved effective for the recognition of the driver's behaviour in the vehicle cabin.

From the investigation of NDRAs' impact on the takeover process, the visual attention-related NDRAs can be categorised as active NDRAs and passive NDRAs based on the interaction mode between the driver and the object involved in the engagement. The results suggest that the NDRAs engagement could increase the time cost for the driver to complete the control transition and the active NDRAs engagement could result in an even longer takeover time. This research also finds that the willingness of performing road-checking behaviour during the NDRAs engagement is different between the active NDRAs and passive NDRAs. During the passive NDRAs engagement, drivers perform the road-checking behaviour more frequently and they always check the road if there

are some changes of the vehicle state (velocity change, vehicle bumping, etc.). However, drivers are more concentrated on the activity during the active NDRAs engagement, and they are not sensitive to such changes, which leads to lower situation awareness of the driving environment. The evaluation of the takeover performance and the road-checking behaviour suggests that the engagement of active NDRAs could lead to high mental demands on the driver. This research explores the method of monitoring the driver's behaviour in the AVs and investigates the influence of the NDRAs engagement, which is of great importance for the design of the takeover strategy and modality in the current and following driving automation.

7.5 Future work

The current video-based dataset of the driver's behaviour could be expanded in future work. From the perspective of the evaluated classes, for DRAs, the driver's road checking behaviour can be further classified as forward road checking, wing mirror checking and rear-view mirror checking. The refined classes could provide more details of the driver's attention, which is helpful for the evaluation of the driver's situation awareness of the surrounding environment. Combining with the environment sensing system of the vehicle, the driver's hazard awareness before the emergency takeover can be further evaluated. For NDRAs, more visual related NDRAs can be investigated, such as chatting via a phone, interacting with the centre console (navigating, watching movies, etc.). Some other NDRAs like eating, drinking, sleeping, calling, chatting with passengers, should also be included in this dataset. It should be noted that, unlike the visual related NDRAs, the impact of these NDRAs on the takeover performance needs to be evaluated. The road checking behaviour or the situation awareness of the driver during the engagement of these NDRAs also needs to be investigated. On the other hand, more participants need to be involved in the experiment to make the dataset equal and unbiased in terms of race, gender, age, and driving experience, etc.

This current research of the driver's behaviour monitoring is off-line based using the collected video dataset. The proposed driver behaviour monitoring system can be implemented on an AV simulator or a vehicle with an automated driving

system. Its real-time recognition performance can be further evaluated. The methods evaluated and developed in this research is computer-vision based. The method based on non-camera sensors can be further investigated and integrated into the investigated driver behaviour monitoring system. For instance, using the microphone to extract the sound in the vehicle cabin, using the haptic sensors like the smartwatch to monitor the driver's state, using the centre console to record the application that the driver is interacting with, such as movies, games, navigation. All this information can be integrated together to develop a more robust and comprehensive system.

7.6 Reference

[1] National Transportation Safety Board, "Collision Between a Sport Utility Vehicle Operating With Partial Driving Automation and a Crash Attenuator, Mountain View, California, March 23, 2018," *Natl. Transp. Saf. Board*, 2018.

[2] W. Morales-Alvarez, O. Sipele, R. Léberon, H. H. Tadjine, and C. Olaverri-Monreal, "Automated Driving: A Literature Review of the Take over Request in Conditional Automation," *Electronics*, vol. 9, no. 12, p. 2087, Dec. 2020, doi: 10.3390/electronics9122087.

[3] P. Bazilinskyy, S. M. Petermeijer, V. Petrovych, D. Dodou, and J. C. F. de Winter, "Take-over requests in highly automated driving: A crowdsourcing survey on auditory, vibrotactile, and visual displays," *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 56, pp. 82–98, Jul. 2018, doi: 10.1016/j.trf.2018.04.001.

APPENDICES

Appendix A Supplementary tables for chapter 2

Table A-1 An example of the estimated 2nd order nonlinear model with ERR value for the first test of the indoor experiment

Model	X				Y			
Priority	Model term	Coefficient	ERR	Importance	Model term	Coefficient	ERR	Importance
1	constant	811.25	88.459%	N/A	constant	528.27	91.563%	N/A
2	gaze_angle_x	-1582.93	11.465%	99.34%	gaze_angle_y	474.31	7.605%	90.14%
3	pose_Tz* pose_Rx	53.43	0.011%	0.10%	pose_Tx	-2.03	0.487%	5.77%
4	gaze_angle_y	-917.28	0.011%	0.10%	pose_Tx* pose_Rx	-35.57	0.125%	1.48%
5	pose_Tx	-0.58	0.008%	0.07%	gaze_angle_x	465.13	0.081%	0.96%
6	pose_Tx* pose_Rx	-46.94	0.003%	0.03%	gaze_angle_y* gaze_angle_y	821.15	0.027%	0.32%
7	pose_Rx	853.26	0.002%	0.02%	pose_Ty* pose_Ty	0.21	0.024%	0.28%
8	pose_Rx* pose_Ry	-4419.04	0.002%	0.02%	gaze_angle_y* pose_Rz	16140.28	0.014%	0.17%
9	pose_Ry	450.32	0.002%	0.02%	gaze_angle_x* gaze_angle_y	-2771.10	0.012%	0.14%
10	gaze_angle_y* pose_Tz	-62.47	0.001%	0.01%	pose_Rx	1248.56	0.011%	0.13%

Table A-2 An example of the estimated 2nd order nonlinear model with ERR value for the second test of the indoor experiment

Model	X				Y			
Priority	Model term	Coefficient	ERR	Importance	Model term	Coefficient	ERR	Importance
1	constant	776.76	88.440%	N/A	constant	545.12	91.542%	N/A
2	gaze_angle_x	-2253.33	10.860%	93.94%	gaze_angle_y	1612.76	7.517%	88.88%
3	pose_Tz* pose_Ry	1.92	0.284%	2.46%	gaze_angle_x	-207.97	0.271%	3.21%
4	pose_Tx* pose_Tx	0.04	0.143%	1.24%	pose_Tz* pose_Tz	-0.01	0.212%	2.51%
5	pose_Ry	-863.78	0.066%	0.57%	pose_Tx* pose_Rx	-27.94	0.094%	1.11%
6	pose_Tx	-5.23	0.050%	0.43%	gaze_angle_y* gaze_angle_y	3018.13	0.052%	0.62%
7	gaze_angle_x* pose_Tz	-2.32	0.016%	0.14%	pose_Tx* pose_Ty	0.23	0.045%	0.53%
8	pose_Rz	-62.47	0.008%	0.07%	pose_Ry* pose_Rz	-2826.91	0.043%	0.51%
9	gaze_angle_x* pose_Rz	987.56	0.005%	0.04%	pose_Ry	-432.25	0.033%	0.41%
10	pose_Tz	0.46	0.002%	0.02%	pose_Ty	2.89	0.018%	0.21%