

UNIVERSIDAD POLITÉCNICA DE MADRID
Escuela Técnica Superior de Ingenieros de Telecomunicación



Vision-based human alertness and attention state modeling in driving scenarios

DOCTORAL THESIS

Submitted for the degree of Doctor by:

Juan Diego Ortega Valdivieso
Ingeniero Industrial

Madrid, 2024



UNIVERSIDAD POLITÉCNICA DE MADRID
Escuela Técnica Superior de Ingenieros de Telecomunicación

Doctoral Degree in Electronic Systems Engineering

Vision-based human alertness and attention state modeling in driving scenarios

DOCTORAL THESIS

Submitted for the degree of Doctor by:

Juan Diego Ortega Valdivieso
Ingeniero Industrial

Under the supervision of:

Dr. Luis Salgado Álvarez de Sotomayor
Dr. Marcos Nieto Doncel

Madrid, 2024

Title: Vision-based human alertness and attention state modeling in driving scenarios

Author: Juan Diego Ortega Valdivieso

Doctoral Programme: Electronic Systems Engineering

Thesis Supervision:

Dr. Luis Salgado Álvarez de Sotomayor
Doctor Ingeniero de Telecomunicación
Universidad Politécnica de Madrid (Supervisor)

Dr. Marcos Nieto Doncel
Doctor Ingeniero de Telecomunicación
Vicomtech

External Reviewers:

Thesis Defense Committee:

Thesis Defense Date:

A mi familia y amigos, por su apoyo incondicional.

Acknowledgement

Llegar a este punto no ha sido tarea fácil. A lo largo de estos años he tenido la suerte de contar con el apoyo de muchas personas que me han ayudado a superar los obstáculos que he ido encontrando en el camino para el desarrollo de esta tesis.

En primer lugar, las primeras palabras las quiero dedicar a mi familia que han estado siempre allí para apoyarme y darme ánimos para seguir adelante. A mis padres, Fredy y María Augusta, que han sido mi pilar fundamental para poder llegar hasta aquí. Su exigencia y su cariño a la vez me han enseñado a no rendirme y luchar por lo que quiero. A mis hermanos, Francisco, Pablo y María Belén, por haber estado siempre preocupados por mí y darme ánimos. A mis abuelos, Ulpiano, Lidia, Jorge y Olguita, que han sido muy cariñosos y siempre han querido lo mejor para mí. Su apoyo ha sido fundamental para poder llegar hasta aquí. A Ander, por ser mi compañero de vida, haberme apoyado en todo momento y aguantado mis desahogos y alegrías. A María Jesús, que ha sido mi segunda madre y me ha dado todo su cariño y apoyo. A mis tíos y primos, que siempre han estado pendientes de mí y me han dado ánimos para seguir adelante. Por cercanía, especial mención a mi prima Lucía, que ha estado allí para escucharme y darme ánimos. Ha sido mi paño de lágrimas y a la vez mi alegría, su estancia en San Sebastián me permitió sentirme arropado y que tenía a un pedacito de familia cerca. A mis amigos, por los momentos de diversión y alegría que hemos compartido y han sido también parte de este trabajo.

En segundo lugar, el poder haber realizado esta tesis en Vicomtech ha sido una gran oportunidad para mí y para mi desarrollo profesional. Especial mención se merecen mis compañeros de trabajo cuyo apoyo ha sido importantísimo para poder completar este trabajo. A Nerea, por el soporte y ser mi compañera de fatigas en los momentos duros. A Paola, por haber entrado en el grupo en el momento adecuado para poder formar equipo en el mundo de los DMS. Espero que esta tesis pueda servirte para continuar con el trabajo que hemos empezado y desarrolla aún más los sistemas DMS. A Orti, por su incansable disponibilidad para ayudar en lo que sea necesario y estar dispuesto a compartir nuestras frustraciones y alegrías. Su alegría y compañerismo siempre estrá en mi memoria. A Marcos, que además de ser mi codirector de la tesis ha sido un compañero de trabajo y mentor excepcional. A mis compañeros de cafés, Lara, José Luis, Andoni, Pedro y se han marchado de Vicomtech, por esos momentos de descanso que han permitido dar un respiro a la mente y poder seguir adelante.

En tercer lugar, pero no menos importante, a mis directores de tesis, Luis Salgado y Marcos Nieto, por su paciencia, su consejo y guía durante todos estos años. Me han ayudado no solo a centrar y focalizar el trabajo técnico sino también con consejos de vida que me han ayudado a crecer como profesional y persona. Han estado allí durante los momentos de bloqueo y cuando la motivación ha flaqueado. Muchas gracias por todo. Sois unos grandes directores.

Finalmente, a todas aquellas personas que se han cruzado en mi camino y me han permitido ser quien soy hoy.

Esto va por todos ustedes.

Juan Diego Ortega Valdivieso

Abstract

Humans are constantly striving to improve their transportation resources. The invention of the combustion engine car is considered a milestone in the evolution of modern transportation, as it has reduced distances and facilitated connections between people all around the world. However, engineers and researchers are continually pushing forward the optimization of transportation and vehicles. In this context, the development of advanced driving assistance systems (ADAS) has provided vehicles with additional functions that improve both the security and comfort of vehicles. In fact, the automobile industry has embraced the task of developing increasingly autonomous vehicles, which relieve the human driver of some of the tasks required when driving a car. This has been manifested by the definition of standardized levels for vehicle autonomy (SAE Autonomous Levels).

The evolution of vehicles towards technologically advanced means of transportation requires the inclusion of driver monitoring systems (DMS), which can assess the driver's state. Notably, the inclusion of DMS is a key element in EuroNCAP vehicle safety evaluation and Europe safety regulations, and it is expected that most vehicles will be equipped with different types of DMS. Research on driver monitoring (DM) is crucial to find the adequate methodology to implement these systems in real cars. The goal of DMS is to assess the driver's capabilities to perform driving tasks, so observing the driver is necessary. Unobtrusive methods to assess the driver's state seem to be the most suitable approaches to building DMS. Among these, vision-based systems stand out as methods for observing the driver using camera sensors, as they offer the ability to infer visible human state impairments.

Human state physiognomy is complex and depends on the individual characteristics of each person. However, some common phenomena could be identified, which allow the classification of signs of driver state impairment. The evaluation of driver capacities is usually performed by DM, which can be degraded by episodes of inattention, putting their safety and that of others in the car at risk. This Ph.D. thesis analyzes the most relevant DMS requirements based on human physiognomy signs of inattention to enable the development of unobtrusive camera-based components for DMS, such as DM datasets. The work done in this thesis aims to provide researchers and industrial practitioners with an overview of driver state characteristics that are useful in building DMS components and tools to develop such systems.

Camera-based methods for assessing the driver's state are based on computer vision (CV) and machine learning (ML) technologies. Nowadays, the results of deep learning (DL) have proven to be very useful in many image analysis problems. However, these methods require large datasets to perform their design task well. Therefore, this thesis presents a methodology for building DM datasets that take into consideration the analysis of DMS requirements. To apply the presented methodology, the **DMD** is built and made available to the research community. The DMD is a multi-modal, multi-camera dataset aimed at covering the most relevant characteristics of DM regarding fatigue and distraction states. Moreover, the DMD is currently one of the most diverse datasets related to DM.

Additionally, the analysis of DMS requirements has led to the development of a real-time software framework based on DMS characteristics. This thesis elaborates on the classification

of the typology of DM algorithms and proposes a modular framework for developing complex yet optimized real-time DMS architectures. The **DMSLib** framework is based on the definition of abstract processing units called Analyzers, which perform specific tasks. These units can be interconnected with each other, sharing common information and algorithm analytics results by using defined interfaces. By connecting different components, it is possible to generate complex DMS pipelines that fulfill any monitoring requirements. A key benefit of the proposed framework is the optimization of DMS pipelines by re-utilization of low-level components' results reducing the time to market efforts for OEMs and Tier 1.

The resulting **DMD** dataset and proposed **DMSLib** framework have been used to develop and test different DM vision-based methods, which consolidate the contribution of this thesis. First, this thesis shows the application of the **DMSLib** and the **DMD** in DM use cases, such as eyelid aperture estimation, driver gaze estimation and driver action recognition. Second, a novel method to estimate the driver's eyelid aperture is presented. This method is relevant since it is based on the capacity to adapt to the human physiognomy of eye shape, which could affect the driver's fatigue state. The method was tested on the **DMD** dataset and other publicly available datasets, and implementation was done using the **DMSLib** framework. Finally, as proof of industrial viability, some DM functions have been integrated into real vehicle systems to validate the integration with autonomous driving (AD) functions and communications (IoT) capabilities.

Resumen

Los seres humanos están constantemente esforzándose por mejorar sus formas de transporte. La invención del automóvil es un hito en la evolución del transporte moderno. Sin embargo, los esfuerzos continúan en la mejora de los vehículos. En este contexto, el desarrollo de sistemas ADAS ha proporcionado a los vehículos funciones que mejoran tanto la seguridad como la comodidad. De hecho, la industria tiene la tarea de desarrollar vehículos cada vez más autónomos, que liberan al conductor de algunas de las tareas requeridas al conducir un automóvil.

El avance de los vehículos requiere incluir sistemas de monitorización del conductor (DMS). La inclusión de DMS es un elemento clave en la evaluación de la seguridad realizada por EuroNCAP y en las regulaciones europeas. El objetivo de los DMS es evaluar el estado del conductor, por lo que es necesario observar al conductor. Los métodos no intrusivos para evaluar el estado del conductor son los más adecuados para construir DMS. Entre estos, los sistemas basados en visión destacan, ya que ofrecen la capacidad de inferir aspectos visibles en el conductor.

La fisiología del estado humano es compleja y depende de las características individuales de cada persona. Las capacidades del conductor pueden verse degradadas por episodios de falta de atención. Este trabajo analiza las características de falta de atención en la fisonomía humana para permitir el desarrollo de componentes basados en cámaras y datos para monitorización del conductor. Además, el objetivo es proporcionar una visión general de las características del estado del conductor que son útiles para construir componentes de DMS.

Los métodos basados en cámaras se basan en tecnologías de visión por computador (CV) y aprendizaje automático (ML). Los resultados del Deep Learning (DL) han demostrado ser muy útiles en muchos problemas de análisis de imágenes. Sin embargo, estos métodos requieren conjuntos de datos grandes para realizar bien su tarea. Esta tesis presenta una metodología para construir conjuntos de datos que tienen en cuenta el análisis de los requisitos de DMS. Como ejemplo de aplicación de la metodología propuesta, se pone a disposición el conjunto de datos **DMD**. El **DMD** es un dataset multimodal y multicámara que cubre las características más relevantes de los estados de fatiga y distracción. Además, el **DMD** es actualmente uno de los conjuntos de datos más diversos relacionados con la monitorización del conductor.

Además, el análisis de los requisitos de DMS ha llevado al desarrollo de un framework de software en tiempo real. Se propone una clasificación de la tipología de algoritmos y una metodología para diseñar arquitecturas complejas pero optimizadas. El framework propuesto **DMSLib**, define unidades de procesamiento, que realizan tareas específicas. Estas unidades pueden interconectarse entre sí, compartiendo resultados de algoritmos mediante el uso de interfaces comunes. Un beneficio clave del framework es la optimización del procesamiento mediante la reutilización de los resultados de los componentes, lo que reduce el tiempo de comercialización para los fabricantes de automóviles (OEM) y proveedores de sistemas.

El dataset **DMD** y el framework **DMSLib** se han utilizado para desarrollar y validar métodos

de monitorización del conductor basados en visión artificial. Se muestra la aplicación de DMSLib y **DMD** en casos de uso, como la estimación de apertura de los párpados, la estimación de la dirección de la mirada y el reconocimiento de acciones. Además, se presenta un nuevo método para estimar la apertura de los párpados del conductor que se adapta a la fisonomía de la forma del ojo. El método ha sido probado en el conjunto de datos **DMD** e implementado con el framework **DMSLib**. Como prueba de viabilidad industrial, algunas funciones de DM se han integrado en sistemas de vehículos reales para validar la integración con funciones de conducción autónoma e IoT.

Table of Contents

Acknowledgement	v
Abstract	vi
Resumen	viii
List of Figures	xiii
List of Tables	xvi
Abbreviations and acronyms	xix
1 Introduction	1
1.1 Context and motivation	1
1.2 Challenges of driver state monitoring	4
1.3 Objectives	5
1.4 Outline and contribution of the thesis	6
2 Definitions of driver state and physiology	11
2.1 Introduction	11
2.2 Driver state definitions	12
2.2.1 Driver state	12
2.2.2 Driver inattention	13
Driver distraction	14
Driver fatigue	15
2.3 Driver state physiological patterns	15
2.3.1 Signs of distraction	15
Driver behavior patterns of distraction	15
Other physiological responses	16
Effects of distraction in driving performance	17
2.3.2 Signs of fatigue	17
Driver behavior patterns of fatigue	18
Other physiological responses	19
Effects of fatigue in driving performance	19
2.4 Conclusions	20
3 Eyelid aperture estimation method	21
3.1 Introduction	21
3.2 Related work	22
3.2.1 Eye blink analysis methods	22

3.2.2	Eyelid aperture estimation methods	23
3.3	Definition of eyelid amplitude	25
3.4	Normalized aperture estimation.	26
3.5	Parameter learning	33
3.5.1	Manual parameter selection	33
3.5.2	Parameter training	34
3.6	Experiments	35
3.6.1	DMD annotation for eyelid aperture estimation	35
3.6.2	Results and discussion	38
3.7	Conclusions	40
4	Driver monitoring dataset generation	43
4.1	Introduction	43
4.2	Related work	45
4.2.1	Datasets in the automotive sector	45
4.2.2	Driver monitoring systems and data requirements	46
4.2.3	Datasets for application in driver monitoring systems	47
4.3	Dataset creation strategy	49
4.3.1	Metadata taxonomy definition	49
	Scenario Metadata	50
	Geometrical Features	50
	Temporal Features	52
4.3.2	Scripted protocols	55
	Distraction Protocol	56
	Drowsiness Protocol	57
	Gaze and Hands Protocol	58
4.3.3	Participants selection	59
4.4	Recording of visual data for Driver Monitoring	60
4.4.1	Multi-sensor setup architecture	60
4.4.2	Recording sessions	62
4.5	Dataset post-processing	64
4.5.1	Stream Compression	65
4.5.2	Multi-sensor stream synchronization	66
4.5.3	Semi-automatic labeling	68
4.6	DMD discussion	72
4.6.1	Published material of the DMD: temporal action recognition	72
4.6.2	Ongoing and future work	74
4.7	Conclusions	75
5	Real-time modular software framework	77
5.1	Introduction	77
5.2	Related work	78
5.2.1	Algorithms and frameworks	78
	Subjective measures	79
	Physiological measures	80

Driving performance measures	81
Driver visual measures	81
5.2.2 DMS regulatory environment	83
5.2.3 DMS systems used by OEMs	85
Commercial vehicles equipped with DMS	88
5.2.4 DMS component suppliers	91
Commercial products and technological activities	91
Consumer electronics devices	91
5.3 DMS methods classification	94
5.4 DMS software framework	95
5.5 Implementation details	97
5.5.1 Analyzers	98
5.5.2 Measures	98
5.5.3 Parameters and Models	98
5.5.4 Sensor Data	98
5.5.5 RTMaps integration	99
5.5.6 Data synchronization	100
5.6 Conclusions	101
6 Driver monitoring applications	103
6.1 Introduction	103
6.2 Eyelid aperture estimation	105
6.3 Gaze estimation	109
6.4 Action recognition	112
6.5 Integration and validation in real testing setups	116
6.5.1 Human-Centered methodology for scenario definition	117
6.5.2 Use case selection for validation	120
6.5.3 Integration details	122
6.5.4 Testing campaigns and validation results	125
6.6 Conclusion	127
7 Conclusions and future work	129
7.1 Conclusions	129
7.2 Future work	132
7.3 Bibliographic contributions	134
References	137

List of Figures

1.1	SAE Levels for autonomous driving.	3
1.2	Overall thesis structure	7
3.1	Classification of Vision-based Eye blink Analysis methods	22
3.2	Classification of Vision-based Eyelid Aperture Estimation methods	24
3.3	Eye landmark fitting and estimation of the height (H) and width (W) of the eye	26
3.4	Eyelid amplitude variations for two individuals blinking normally	27
3.5	Eyelid aperture normalization method	28
3.6	Representation of likelihood of the eye to be open	29
3.7	Parameters involved in the computation of the normalized aperture for a sample sequence	30
3.8	Representation of the sigmoid function used to calculate the EWMA learning factors	31
3.9	Sample evolution of eyelid closure metrics	32
3.10	Map of cost function J for each user and for α and β values spanning from 0.0 to 1.0 in steps of 0.01	35
3.11	Map of the cost function J for α and β values spanning from 0.0 to 1.0 in steps of 0.01	35
3.12	Temporal Annotation Tool (TaTo) for annotating eye states of the DMD videos	38
4.1	Examples of activities performed in the DMD	48
4.2	General overview of the DMD creation process	49
4.3	Metadata label taxonomy for the DMD	51
4.4	DMD Participants information	59
4.5	DMD camera setup and recording environments	61
4.6	Total raw data volume in terabytes (TB) of the DMD dataset	64
4.7	Distribution of recordings categorized by the exhibited type of driver behavior	65
4.8	Process for multi-sensor stream synchronization	67
4.9	DMD Frame shift error across groups	69
4.10	DMD Frame shift error across protocols	70
4.11	DMD classes distribution	73
4.12	Procedure to distribute the DMD	74
4.13	DMD file structure	75
5.1	Toyota DMS analyzing face orientation and eyelid analysis	85

5.2	Toyota DMS sequence of system operation	86
5.3	DMS proposed by Mitsubishi	87
5.4	Cadillac CT6 dashboard, DMS and HMI	89
5.5	DS 7 Driver Attention Monitoring	90
5.6	Subaru Forester DMS integration	90
5.7	Image analysis of Seeing Machines DMS	93
5.8	Affectiva DMS detection of emotional state	93
5.9	Proposed framework for DMS pipeline generation	96
5.10	Interrelations between the components of the proposed DMS software framework	97
5.11	DMS pipeline example as a combination of RTMaps components	99
5.12	Representation of RTMaps circular buffer	101
6.1	Example of DMS architecture build based on proposed framework	104
6.2	Eyelid aperture estimation pipeline	105
6.3	Driver gaze estimation pipeline	110
6.4	Driver action detection pipeline	113
6.5	VI-DAS Human Centred design methodology	118
6.6	VI-DAS validation scenarios	119
6.7	VI-DAS AHIDA-M and AHIDA-A Test Cases	122
6.8	Component diagram of the VI-DAS instrumented vehicle	123
6.9	Integration of DMS pipelines in driving simulation	124
6.10	Instrumented vehicle with interior cameras for face, body and hand monitoring	124
6.11	DMS integration in VI-DAS lane change Use Case	125
6.12	DMS signals transmitted to an IoT platform.	127

List of Tables

2.1	Driver state physiological constructs	13
2.2	Driver state classification in terms of inattention	14
3.1	Comparison of frame classification accuracy between open and close states .	39
3.2	Classification results of open and closed states	40
4.1	Comparison of public vision-based driver monitoring datasets	48
4.2	Subset of annotation levels for distraction-related material in the DMD . . .	53
4.3	Subset of annotation levels for drowsiness-related material in the DMD . . .	54
4.4	Subset of annotation levels for gaze-related material in the DMD	55
4.5	Actions performed in the distraction protocol	57
4.6	Actions performed in the drowsiness protocol	57
4.7	Actions performed in the gaze-hands protocol	58
4.8	Specifications of Intel® RealSense™ cameras used to generate the DMD . . .	62
4.9	Camera recording parameters and required bandwidths	63
4.10	Comparison of annotation times using v1 and v2 of TaTo	72
5.1	Karolinska Sleepiness Scale (KSS)	80
5.2	DMS analysis categories and examples	94
5.3	Examples of <i>Measure</i> elements in DMS software	96
6.1	Parameters and Interfaces of the <i>Video Capture</i> component	106
6.2	Parameters and Interfaces of the <i>Face Detection</i> component	107
6.3	Parameters and Interfaces of the <i>Face 2D Landmarks</i> component	108
6.4	Parameters and Interfaces of the <i>Eyelid Amplitude</i> component	108
6.5	Parameters and Interfaces of the <i>Normalized Eyelid Aperture</i> component . .	109
6.6	Parameters and Interfaces of the <i>Face 3D Model Fitting</i> component	111
6.7	Parameters and Interfaces of the <i>Head Pose Estimation</i> component	111
6.8	Parameters and Interfaces of the <i>Eye Tracking</i> component	112
6.9	Parameters and Interfaces of the <i>Eye Gaze Estimation</i> component	112
6.10	Parameters and Interfaces of the <i>Body Detection</i> component	115
6.11	Parameters and Interfaces of the <i>Body 2D Landmarks</i> component	115
6.12	Parameters and Interfaces of the <i>Body 3D Pose Estimation</i> component . . .	116
6.13	Parameters and Interfaces of the <i>Object Detection</i> component	116
6.14	Parameters and Interfaces of the <i>Action Recognition</i> component	117

6.15 Locations where the validation campaigns were conducted for the VI-DAS project 125

6.16 VI-DAS validation checklist for each test case 126

Abbreviations and acronyms

AD Autonomous Driving

ADS Automated Driving Systems

ACC Adaptive Cruise Control

ABS Anti-lock Breaking System

AD Autonomous Driving

ADAS Advanced Driver Assistance Systems

AI Artificial Intelligence

CAS Collision Avoidance System

CCAM Connected and Cooperative Automated Mobility

CNN Convolutional Neural Network

CV Computer Vision

DL Deep Learning

DNN Deep Neural Network

DMD Driving Monitoring Dataset

DMS Driver Monitoring System

ECU Electronic Control Unit

EEG Electroencephalogram

ECG Electrocardiogram

EOG Electrooculogram

FUT Feature under Test

HMI Human Machine Interface

IIVIS In-Vehicle Information Systems

IOT Internet of Things

LDWS Lane Departure Warning System

LSTM Long Short-Term Memory

ML Machine Learning

ODD Operational Desing Domain

RNN Recurrent Neural Network

SUT System Under Test

TC Test Case

Chapter 1

Introduction

1.1 Context and motivation

Safe driving requires a complex amalgamation of driver behaviors called the Dynamic Driving Task (DDT). The DDT requires the driver to successfully and simultaneously perform several complex activities such as route finding, route following, controlling speed and lateral position, avoiding collisions, obeying road rules and monitoring the status of vehicle systems [32]. Despite the significant attentional resources that such activities demand, drivers are often observed to engage in other activities such as using the mobile phone, manipulating the radio or GPS, talking to passengers, etc., that have the potential to divert attention away from activities critical for safe driving.

Over the past decade, there has been a growing interest in developing automotive systems [191] that can enhance the comfort and safety of drivers, particularly in Europe, to eliminate fatalities and injuries on European roads by 2050, as outlined by the 'Vision Zero' initiative [52]. One of the latest trends in the transportation industry is to provide systems that can automate the DDT, thereby minimizing the need for human intervention in operating the vehicle. With the increasing level of vehicle automation, cars are gradually evolving into multi-sensor computational systems that are designed to comprehend (i) the driver's state and (ii) the state of the driving environment [204]. Consequently, a crucial objective is to attain a fully autonomous vehicle system that can evaluate, comprehend, and tackle complex driving scenarios, with human behavior also considered a crucial aspect of the equation [1], [189].

Within vehicle automation, key safety issues under debate in the transition from manual to fully autonomous vehicles are human-related and often described with terms such as Driver engagement, Driver-in-the-loop, Driver Monitoring, Fall-back Readiness, Receptiveness, and Availability [24]. These terms are frequently used to describe driver responsibilities while using automation. It is possible to classify these driver-related safety issues into the following types: (i) recognition error e.g., distraction and inattention; (ii) decision error e.g., aggressive driving behavior; (iii) performance error e.g., overcompensation; and (iv) non-performance error e.g., mental workload and fatigue [195]. All these human factors related to driver behavior capabilities are important to be assessed in automotive systems.

Although automation is not new in vehicle control systems, the automobile industry has made a firm commitment to support developments towards a transformation of automobiles driven by human drivers to intelligent robots who transport humans on the roads. Many vehicle systems already have self-acting or self-regulating mechanisms, such as adaptive cruise control (ACC), anti-lock braking system (ABS), automatic parking, blind spot detection, collision avoidance system (CAS), lane departure warning system (LDWS), parking assistance[148]. These systems have greatly improved the vehicle’s ability to interpret its state and context, significantly enhancing situational awareness.

Some vehicles, like Tesla Autopilot [206] and Cadillac Super Cruise [22] systems, already possess SAE L2 functionalities. Moreover, since 2021, the market has seen the introduction of the first conditional autonomous (SAE L3) vehicles, such as the Mercedes-Benz S-Class and EQS models equipped with the Mercedes-Benz DRIVE PILOT system [140], as well as the Honda Legend featuring Honda’s SENSING Elite technology [89]. However, a fully autonomous vehicle that does not require human intervention (SAE L4-L5) is not yet commercially available [212].

For Highly Automated Vehicles (HAV), “activities critical for safe driving” are some combination of the driver monitoring, the driving environment, supervising the automated driving system and being the fallback operator [86]. The introduction of new autonomous features most likely will produce out-of-the-loop problems, mode confusion and behavioral adaptation issues [142]. Hence, as more and more automated systems are used in the vehicle, the driver needs to allocate fewer attentional resources to the DDT, and this could produce a decrease in driver availability to take over the control of the vehicle. Although the consequences of driving with this type of advanced driver assistance systems (ADAS) are still under research by the Human-Factors and Ergonomics communities, it is clear there is a need to monitor the attentional state of the driver, specifically under different automated levels [14], [62].

Within the industry and research community, there is a concern for driver engagement safety issues while using increasingly sophisticated automation, e.g., SAE Level 2 and 3 systems [24] (see Figure 1.1). There are many open questions in autonomous driving (AD) that are still unresolved: What will be the role of humans in such an approaching future? Would they seat as passive occupants who fully trust their vehicles? Would there be a need for humans to “take over” control in some situations either triggered by the need perceived by the autonomous vehicle or desired by someone in the cabin? How should these autonomous vehicles interact with agents outside the vehicle (other drivers of non-autonomous vehicles, pedestrians, etc.)? [155].

There are several concerns surrounding the feasibility of Level 3 automated vehicles (i.e., conditional driving). While a few OEMs have recently announced the commercial availability of such vehicles, others have abandoned their Level 3 programs. The key difference between Level 2 (L2) and Level 3 (L3) lies in the driver’s role. In L2, the driver is responsible for continuously monitoring the DDT and must be prepared to retake control within a short timeframe if the automation fails. In contrast, L3 vehicles do not require the driver to monitor the DDT. Instead, the system is responsible for predicting when the driver may need to resume control and assessing the driver’s readiness, leading to various insurance and legal concerns.

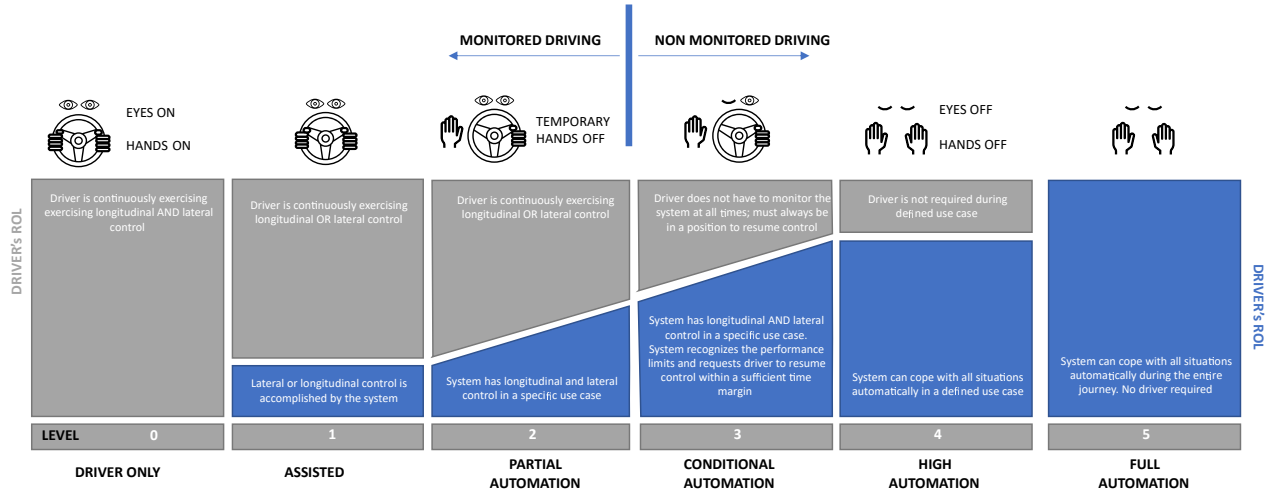


Figure 1.1: SAE Levels for autonomous driving [182]. The gap between L2 and above mainly relies on the need to release the driver from certain driving tasks, but also monitor the driver to return to manual driving when the AD function exits its Operational Design Domain (ODD).

As a result, many automakers are investing in what is known as L2+, a refined form of partial automation L2 mode in which the vehicle can operate in larger sets of scenarios, defined as extended Operational Design Domain (ODD), and thus perform automated driving for longer periods of time, always under the supervision of the driver. The current trend is to maintain the driver in the loop for commercial vehicles until 2050, as outlined in the ERTRAC roadmap for Cooperative, Connected, and Automated Mobility (CCAM) under the working group "Connectivity and Automated Driving" [49]. This roadmap envisions L4-like systems for buses and shuttles, supported by control centers, while passenger cars will primarily offer L1/L2 functionalities that cover most of the possible scenarios. Although many questions remain unanswered, it is evident that intelligent vehicles will rely on the collaboration between two intelligent agents, one artificial and another human.

Under this collaboration, human beings will remain an integral part of the DDT, hence, the need for intelligent vehicles to automatically observe, understand, model, infer and anticipate driver's constantly changing state and behavior is necessary now more than ever [3]. Driver sensing refers to multi-modal estimation of the overall physical and functional characteristics of the driver including the level of distraction, fatigue, attentional allocation and capacity, cognitive load, emotional state, and activity [62]. These features are expected to be part of future Driver Monitoring Systems (DMS) of SAE Level 3 vehicles.

There are different types of sensors and input signals to characterize driver state and behavior. The first trend collects signals directly from the vehicle such as steering angle, pedals activity, acceleration, velocity, lane position, etc., and builds systems to correlate these signals with impaired driver state [3], [138]. However, as autonomous features will become available in a greater number of cars, driving action on the vehicle will be less available. Hence, there will not be driver inputs to measure from. Nevertheless, it is still necessary to continuously monitor the driver's state during autonomous operation, as the vehicle could require the driver

to get back control. Hence, only vehicle measures are not valid inputs, in all circumstances, and require other types of sensors and signals to assess the driver’s state.

The second trend investigates the relation between measurable physiological cues and driver state [3], [193]. Within these physiological measures, two types of parameters can be identified: internal physiological signals such as electroencephalogram (EEG), electrocardiogram (ECG), electrooculogram (EOG), skin conductivity, heart rate, etc.; and external visible physiology signals such as eye and eyelid movements, head position, mouth activity, body pose, etc. The former requires intrusive instrumentation attached to the body of the driver while the latter requires the use of non-attached sensors such as video cameras installed inside the vehicle. In general terms, body-attached sensors are not desirable in the automotive industry as they reduce the comfort of the user and the price of such equipment is much higher.

Vision-based driver monitoring methods, first measure the visible physiological cues to obtain important instantaneous features such as head pose, eyelid aperture, gaze direction, body and hand pose, etc. Then, based on these physiological measures of the driver it is possible to infer and predict the driver state and driving behavior [65]. These two steps in driving monitoring theory have been studied from different disciplines’ perspectives for some time. From a computer vision (CV) and machine learning (ML) perspective, the goal is to assess visual-based metrics that describe various characteristics of the physiological state of the human and process them in a temporal fashion to generate a driver state and driving behavior model [155].

1.2 Challenges of driver state monitoring

Modeling the driver’s state and behavior using vision-based methods face several challenges due to external and internal factors. For instance, external factors usually include variations in lightning conditions due to the dynamic driving environment (i.e., different times of the day, weather conditions, traffic density, etc.). Besides, the camera perspective and occlusions cause information loss. Occlusion can be caused by an opaque object located between the camera and the monitored user, or part of the monitored user is occluded by itself (self-occlusion) due to large movements and camera perspective limitations. On the other hand, internal factors are those that involve the interpersonal variability along different drivers such as user ethnicity, age, gender, driving experience, regional driving practices, etc.

Both external and internal factors require different methodologies to overcome the challenges. On the one hand, external factors require using better training databases and improved ML and image processing methods. For some cases, such as low illumination and occlusions, specific hardware is needed, i.e., cameras with IR illumination [33], [146] and multi-camera setups [4], [136]. On the other hand, internal factors (i.e., inter-personal variability) usually imply the application of personalized models that incrementally learn the particular features and context of the driver [102]. There have been a lot of efforts to improve the accuracy and performance of CV algorithms to model driver behavior in a global fashion. However, there is still a lack of methods that identify and learn the specific characteristics and dynamics of diverse drivers.

Moreover, the analysis of human behavior while driving a vehicle is not an easy task due to the high number of modalities of human attention and the presence of complex interrelationships between driver state constructs and their indicators [139]. A consensus of the factors that cause the different driver states has not been reached. However, several works have taken steps towards automatic monitoring of driver state and behavior [41], [134], [155], [204]. It is important to keep track of as many physiological phenomena as possible to promote valid and reliable interpretations of the human state.

In addition to the mentioned challenges, traditional computer vision algorithms for driver monitoring usually rely on the availability of annotated databases to build supervised models for human-centric visual cues extraction. However, the annotation of video sequences in a naturalistic driving environment is very time-consuming and for some attention mechanisms (e.g., fatigue and cognitive load) require the validation with intrusive sensors (EEG, ECG, EOG, etc.) [16], [107].

Furthermore, most of the state-of-the-art classifiers and regressors are based on Deep Learning (DL) methods which require a very large amount of training data. Hence, there is a need to explore semi-supervised or unsupervised learning methods to extract visual patterns of driver state and behavior. Moreover, it is difficult to collect a significant number of samples of interesting “in-the-wild” events for driving monitoring such as fatigue or distraction since it will compromise the safety of volunteer subjects. To overcome this issue, simulation environments could help reproduce the same driving conditions and collect the required data for developing and testing the desired methods.

1.3 Objectives

Vision-based driver monitoring is a broad field in the automobile sector that still needs further development of tools to build exploitable DMS. This thesis aims to analyze the requirements of driver monitoring which allows the development of non-intrusive applications based on cameras that enable the construction of a multi-modal and multi-sensor image dataset; a real-time scalable software framework for DMS integration and their application to build CV analysis methods such as eyelid aperture estimation, gaze estimation and driver action recognition.

The challenges for developing vision-based DMS lead to the following research questions:

- RQ1 What are the necessary facial and body features to determine the driver’s attentional state?
- RQ2 Should DMS methods be designed to adapt to the specific physiological aspects and characteristics (such as eyelid shape) of individuals?
- RQ3 What characteristics should a dataset used for driver monitoring have to cover the most relevant aspects of DMS?
- RQ4 Can a structured methodology be proposed to design, plan, and perform a full collection of driver monitoring data?

RQ5 How should a software framework be devised to support the effective execution of the multiple components required for DMS applications?

RQ6 Could new datasets and structured software architecture benefit new methods for face and body analysis?

To answer the research questions of the thesis, three main lines of work are defined:

1. The proposal of a new methodology to collect and annotate a multi-sensor and multi-modal Driver Monitoring Database (DMD)
2. Definition of a novel software framework for the development of Driver Monitoring Systems (DMS)
3. Analysis and application of state-of-the-art vision-based algorithms for driver state and behavior modeling

A list of specific objectives is derived which guide the development of this thesis:

- O1 Review the human physiology signs of fatigue and distraction as a manifestation of driver inattention behavior.
- O2 Develop a novel eyelid aperture estimation method based on the analysis of the eyelid shape which adapts to the particular eye shape characteristics of each individual.
- O3 Propose a metadata taxonomy to describe and characterize a dataset for driver monitoring based on the reviewed signs of driver inattention.
- O4 Design a methodology to plan, collect and process a complex driver monitoring dataset based on multi-camera setup in real driving and simulator scenarios.
- O5 Develop a generic software framework to enable the implementation of real-time vision-based driver monitoring systems.
- O6 Demonstrate the application of the proposed dataset and real-time framework to build vision systems components to monitor driver eyelid aperture, gaze estimation and action recognition and its application in real DMS.

During the development of this thesis, the research questions are answered with the final goal of accomplishing the stated objectives. The outcomes of these answers are presented in scientific journals and international conferences. Along with the development of this thesis, the findings when answering the research questions are summarized.

1.4 Outline and contribution of the thesis

The proposed thesis contributes to the state-of-the-art in driver monitoring by providing and demonstrating different tools to build vision-based DMS. The contributions of this thesis analyze the requirements to build complex modular DMS in modern computational platforms taking into consideration the industry performance and practical requirements.

The overall concept of the thesis is presented in Figure 1.2.

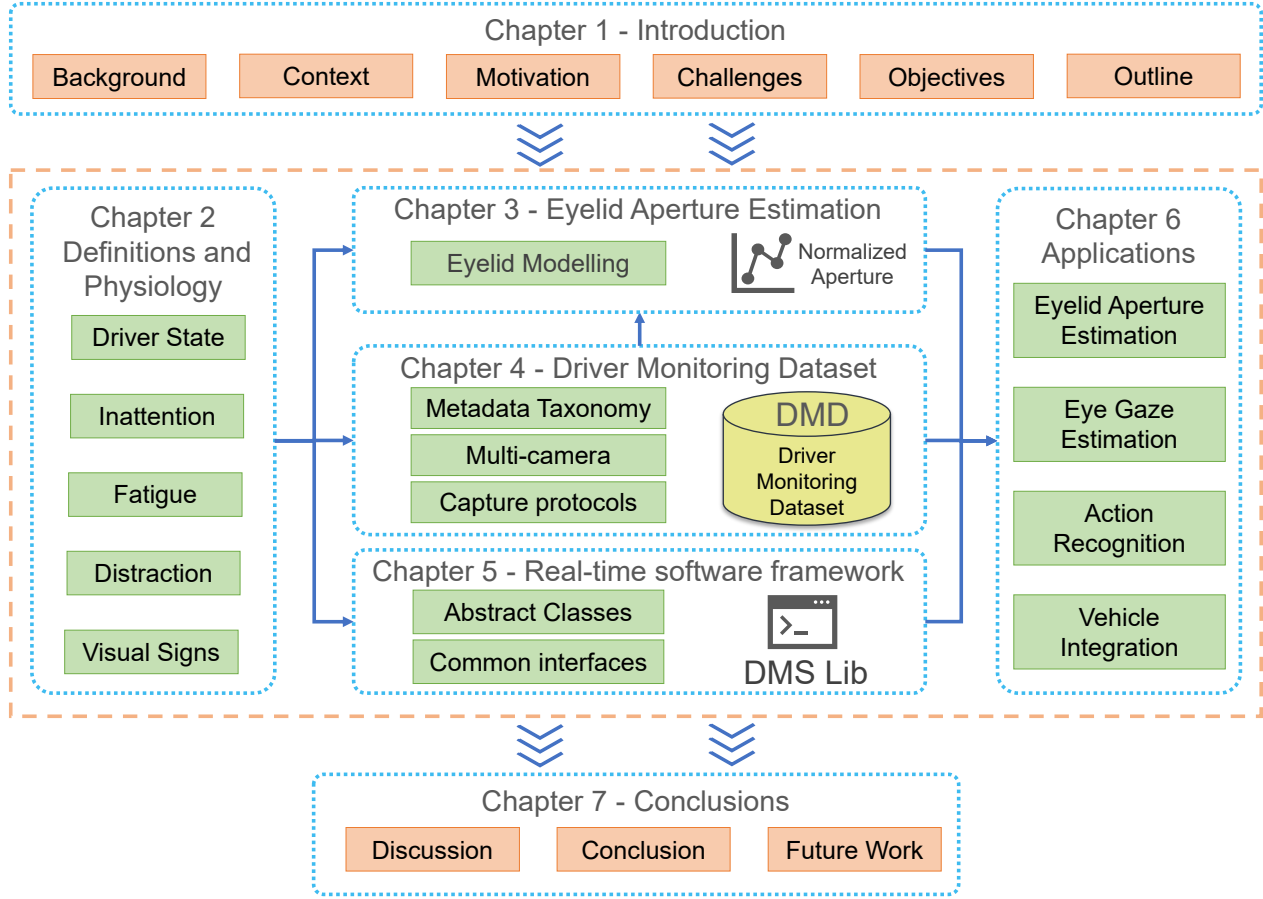


Figure 1.2: Overall thesis structure

In Chapter 2 the analysis of driver state physiognomy is presented. This analysis is performed based on the technology perspective of vision-based Advanced Driver Assistance Systems (ADAS). Part of this work is presented in the international journal (IET Intelligent Transport Systems, 2015) [148]. The main contributions can be identified:

- Classification of driver state physiological behaviors based on visual characteristics.
- Definition of visual-based system requirements to sense fatigue and distraction.

Based on the analysis of the different characteristics of driver inattention, especially fatigue and drowsiness; Chapter 3 offers a thorough explanation of the methodology used in driver eye blink analysis. This chapter focuses on the development of a novel eyelid aperture estimation method that examines the distinctive features of individual eyelid shapes. The method is designed to adapt and accommodate the diverse eye shape variations observed in different individuals. This chapter follows the review of the state of the art in driver state monitoring, as it represents the first technical contribution within the scope of this thesis. The chapter is presented in a natural chronological order, although the work covered in subsequent chapters was later used to validate the proposed method for eyelid aperture estimation. This work is supported by the presentation at an international conference (VEHITS, 2020) [164]. The main contribution of this work is:

- A novel user-adaptive eyelid aperture estimation method was developed which adapts the eyelid basal state to the person’s physiognomy and changes over time. This method leverages the availability of data in the DMD (Chapter 4) and the integration process of the DMS library (Chapter 5).

In most of the AI problems nowadays, there is a clear need to have a large amount of labeled data to train and test the algorithms. In the field of driver monitoring this is not an exception. Available ground truth of driver behavior with reliable complex annotations of inattention events is not easy to find. In Chapter 4, the process of building a multi-modal, multi-camera DM database is presented. The DMD dataset was first introduced in an ECCV workshop publication in 2020 [162]. Later, the description of the methodology and process of creating the dataset was published in an international journal (Sensors, 2022) [161]. The main contributions are:

- Definition of a taxonomy of minimal dataset metadata required to build vision-based DMS.
- Design of requirements for large-scale driver monitoring datasets, including the definition of relevant events to be captured, logistics and planning of recording sessions, and definition of sensor setup and characteristics.
- Development of an annotated driver monitoring database (the DMD) that includes several modes of driver inattention (e.g., distraction and fatigue) for different drivers and under various driving conditions. This dataset allows the development of supervised ML models for driving monitoring.

To implement DM algorithms in industrial applications some requirements in terms of software implementations were identified. Then, in Chapter 5 the real-time modular framework DMS is described. Some of the benefits of using the DMS framework are the reuse of components in complex pipelines, integration with automotive middleware, and the ability to assess latency and synchronization. This contribution is supported by the presentation of the work at an international conference (Conference on Real-Time Image Processing and Deep Learning, 2021) [163]. The main contributions of this work are:

- Classification of DMS feature analysis based on complexity level: low-level, physiological, and driver status features.
- Development of a modular software architecture that is flexible to incorporate CV algorithms to build DMS in industrial-grade applications.
- Definition of abstract interfaces to make it easy to achieve interoperability between the DMS modules.

Finally, Chapter 6 presents a novel contribution to the field of DMS by applying different state-of-the-art analysis methods based on CV and ML to validate the proposed DMS framework and the application of the DMD dataset. These have proved to be successful use cases in relevant demonstrations in international events and research projects [160], [210]. The main contributions of Chapter 6 can be summarized:

- Description of the application of modular pipelines, including interfaces and param-

ters definition, of different relevant DMS functions: eyelid aperture estimation, gaze estimation, and driver action recognition.

- Application of the DMS framework for an eyelid aperture estimation was presented at an international conference (CCIS, 2021) [160]. For this use case, the method explained in Chapter 3 was applied.
- Application of a preliminary version of the DMS framework for a gaze estimation system was presented at an international conference (DDI, 2018) [77]
- Application of DMS framework and DMD dataset to develop a driver action recognition system, presented in an international conference (VISIGRAPP, 2021) [25]
- Application of DMS systems in the validation of autonomous functionalities (CCIS, 2021) [160] and Cloud IoT systems (IEEE Intelligent Vehicles Symposium, 2019) [157].

Chapter 2

Definitions of driver state and physiology

2.1 Introduction

Human errors are one of the main causes of traffic accidents. According to NHTSA [195], up to 94% of all observed road-related accidents are due to these reasons. These errors are complex to analyze because there is not a single important factor that can describe them. For instance, one error could be produced because of a lack of attention or incorrect control. However, there is generally a major critical reason in every human error-related traffic accident, which is sometimes influenced by pre-crash events and associated factors.

It is possible to classify driver-related critical reasons into the following types: (i) recognition error, e.g., distraction and inattention, (ii) decision error, e.g., aggressive driving behavior, (iii) performance error, e.g., overcompensation, and (iv) non-performance error, e.g., mental fatigue [195]. To reduce accident fatalities and enhance driving safety, it is necessary to design information systems that track this multitude of driver-attributed critical reasons to pre-determine a high risk of an accident and try to eliminate it.

Since the early 90s, the interest to understand driver state and their influence on driving performance has increased considerably and many research works have been published. For instance in [150] the authors describe the symptoms of fatigue during simulated driving and concluded that the subjective level of fatigue was similar at the end of the driving session, regardless of how long the subjects drove before wanting to quit. Hence the subjective measures are fundamentally not suitable to determine the driver's state. An impaired mental and physical state makes a driver prone to errors and could lead to dangerous driving [84], [90].

Modern ADAS, like lane departure warning, parking assist and adaptive cruise control, became very robust in interpreting vehicle state and context; this has significantly enhanced the vehicle's awareness of a driving situation. However, the majority of these systems do not address common causes of abnormal driving behavior, namely constantly changing drivers' states.

Analysis and online tracking of the driver’s state could help to estimate and reduce the risk of incorrect vehicle control before its actual occurrence, which subsequently has the potential to reduce the number of human error-related road accidents. Such a feature is often referred to as Driver State Monitoring (DSM). It is also recommended to combine driver state with driving context and performance to boost even more the control of driving risk. Advances in sensor technology and an increase in computational capability have helped the research community to develop innovative DSM systems, actually, the current regulations and certification authorities in Europe [51], [53] already require some form of DSM systems and will gradually require improved versions for 2030.

This chapter aims to analyze the physiological aspects of the driver’s state and use this information to define requirements for DSM. The purpose of this study is to identify the main characteristics of driver inattention with the final objective of extracting the visual features necessary to be considered when designing computer systems that can automatically sense, assess, and estimate the driver’s state. The contributed analysis presented in this chapter is based on various taxonomies of driving inattention. The review is followed by identifying the phenomena of driver inattention, which will help in extracting visible signs that a DMS must prioritize in terms of data and vision-based methods.

2.2 Driver state definitions

Driving is a complex process that requires situational awareness of the environment, decision-making, and performance of actions. In this process, the most complicated stage is situation awareness. In [46], a three-level situation awareness model is defined as “the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future.”

The process of perceiving information involves deploying attention, which can limit a person’s ability to accurately perceive multiple items simultaneously, and ultimately affect their situation awareness. Direct attention is crucial not just in perceiving and processing available cues but also in the later stages of decision-making and executing a response. In a complex and dynamic driving environment, there are several demands on attention due to information overload, complex decision-making, and multitasking. Hence, monitoring the driver’s attention status is vital to ensure safe driving [41].

2.2.1 Driver state

Correctly defining driver state is inconsistent, which has several implications. First, it can make it difficult or even impossible to interpret and compare research findings across studies for a given form of state. Second, similar works may be measuring slightly different constructs and measuring different outcomes.

According to [139], driver state can be decomposed into the following four constructs: (a) cognitive distraction, (b) mental workload, (c) mental fatigue and (d) emotions. The definitions of these constructs are presented in table 2.1.

They state that most of the seen accidents were caused by a combination of these driver

Driver State construct	Definition
Cognitive distraction	"any activity that diverts a driver's attention away from the task of driving" [124]
Mental workload	"the specification of the amount of information processing capacity that is used for task performance" [215]
Mental fatigue	"a functional state, which grades in one direction into sleep, and in the other direction into a relaxed, restful condition, both of which are likely to reduce attention and alertness"[121]
Emotions	"a complex pattern of changes, including physiological arousal, feelings, cognitive processes, and behavioral reactions, made in response to a situation perceived to be personally significant."[74]

Table 2.1: Driver state physiological constructs

state constructs. As a result, they conclude that the state of the driver is a multidimensional construct, and the evaluation of this construct using physiological indicators may be challenging due to the intra-personal nature of such indicators.

All of these driver constructs have a direct relation with driver attention. Hence, each of the four constructs affects driver performance on the main driving task both positively and negatively [13]. According to [168], distraction is a major factor in up to 25% of all observed highway crashes. Additionally, high level of mental fatigue was found to be responsible for up to 20-30% of road fatalities [91]. Moreover, elevated levels of mental workload could lead to negative stress and may cause impairments and distraction in completing the primary task of driving [122]. Nevertheless, a low level of mental workload could lead to impaired vigilance and cause mental fatigue. Driver mental workload should be optimal, i.e. not too high, not too low, to ensure adequate driving performance [168]. Emotions, and anger, in particular, were found to degrade driving performance. While happiness could be beneficial to driving [176].

2.2.2 Driver inattention

There is not a common agreement in the definition of driver inattention or distraction. Usually, these two terms are used indistinctly. In [172], the authors introduce a complete taxonomy of driver inattention in terms of the different mechanisms by which inattention may arise. In the taxonomy, driver distraction is conceptualized as just one of several processes that give rise to driver inattention. According to their statement, while distracted driving can lead to inattentiveness, it's not the only cause of inattention, indicating that inattention can occur without the presence of a distraction. The taxonomy was intended to be used in the classification and categorization of accident databases. In this study the following definition of inattention was given: *"Insufficient, or no attention, to activities critical for safe driving"* [172]

This taxonomy is useful to measure the statistics of accidents caused by driver errors. However, the taxonomy is not very useful from an engineering point of view, in which a taxonomy of the driver state should be proposed in terms of the measurable physiological parameters that determine the driver state.

Besides, other studies define driver state as the level of driver inattention. Likewise, DMSs have studied driver inattention following two main categories: **(a) distraction** and **(b) fatigue** [41]. Both of them are grouped into the driver's inattention mechanism described before. Hence, taking into account the different definitions described before, table 2.2 summarizes the different classifications of driver state in terms of inattention relevant to vision-based DMS.

		Driver State Constructs
Driver Inattention	Distraction	Cognitive distraction
		Mental workload
		Emotions
	Fatigue	Mental fatigue

Table 2.2: Driver state classification in terms of inattention

Driver distraction

Distraction usually involves a "diversion of attention away from activities critical for safe driving". Different types of distraction have been categorized. In [41] the authors added two more types to the NHTSA classification of types of distraction [168]:

1. visual distraction (e.g., looking away from the roadway);
2. cognitive distraction (e.g., being lost in thought);
3. auditory distraction (e.g., responding to a ringing cell phone);
4. biomechanical distraction (e.g., manually adjusting the radio volume);
5. olfactory distraction (e.g. bad smells in the road);
6. gustatory distraction (e.g. eating while driving);

However, many distraction situations could be produced by a combination of these types, increasing the risk of having an accident. Particularly, two types of distraction are of interest: visual and cognitive. Visual distraction occurs when a driver's visual attention is diverted away from the road scene to something else, such as the car stereo. Drivers are more vulnerable to distractions that originate from within the vehicle compared to those from outside, which highlights the two basic categories of visual focus: the road scene and the interior. On the other hand, cognitive distraction happens when the driver's visual focus is on the road but their ability to correctly assess the driving task is compromised. This can be described as "seeing but not seeing," where drivers may appear to be staring at a point in space, resulting in limited eye movements that make it harder to estimate distances. The challenge with this type of distraction is that it cannot be defined solely in terms of visual focus.

Driver fatigue

The phenomenon of fatigue is different and its definition could be more difficult. The term fatigue refers to a combination of symptoms such as impaired performance and a subjective feeling of drowsiness. Nevertheless, there is no consensus in the literature on which methods for extracting driver state should be prioritized in an effective Driver Monitoring System (DMS). According to the European Transport Safety Council (ETSC), fatigue "concerns the inability or disinclination to continue an activity, generally because the activity has been going on for too long" [50]. Taking into account organ functionality, different types of fatigue can be identified [41]:

1. local physical fatigue (e.g., in a skeletal or ocular muscle);
2. general physical fatigue (following heavy manual labor);
3. central nervous fatigue (drowsiness);
4. mental fatigue (not having the energy to do anything).

2.3 Driver state physiological patterns

To assess the driver's state and determine the presence of different forms of inattention, such as distraction and drowsiness, it is crucial to identify both physiological signs exhibited by the driver and changes in driving performance. These mechanisms are outlined in table 2.2. In this section, those signs and effects are presented from a physiological and behavioral perspective.

2.3.1 Signs of distraction

Performing a cognitively demanding task while driving influences both the driver's visual behavior and driving performance, like the braking behavior.

Driver behavior patterns of distraction

In some driving situations, there is an increase in the cognitive demand. As this demand increases, many drivers change their inspection patterns on the forward view. In [5], the eye glance pattern could be used to discriminate driving while performing a secondary task from driving alone. It could also be utilized to differentiate secondary tasks with high workloads. Further insights into the effects of cognitive distraction while driving were explored in [169] and [16]. Their findings suggest that distracted drivers exhibit reduced external scanning and spend more time looking straight ahead, resulting in decreased monitoring of instruments, mirrors, traffic signals, and the surrounding area near intersections. In addition, they spend less time looking at the road ahead and more time looking at the road directly in front of the vehicle. These findings are consistent with the notion that drivers are less likely to scan the environment when they are distracted.

In [85], the authors showed that drivers reduced their inspection of the external view and spent more time looking directly ahead. They reduced their inspection of the instruments

and mirrors and reduced their glances at traffic signals and the area around an intersection. All these studies show consistently that the amount of workload affects directly driver's distraction. The study revealed that the visual field diminished by 7.8% during a moderate workload counting task, and by 13.6% during a more demanding counting task that required greater cognitive effort [169]. Drivers had fewer saccades per unit time, which was consistent with a reduction in glance frequency and less exploration of the driving environment, and in some cases, drivers completely discard these tasks and did not inspect these areas at all [85].

In [86], the authors showed links between eye movements (fixation, saccade, and smooth pursuit), cognitive workload, and distraction.

- Fixations happen when the eyes of an observer are nearly stationary.
- Saccades are very fast movements that occur when visual attention shifts from one place to another.
- Smooth pursuits occur when the observer tracks moving objects such as a passing pedestrian.

Saccade distance decreases as task complexity increases, which indicates that saccades may be a valuable index of mental workload. Conversely, the amount of head movement increases when cognitive loads are imposed [86]. This behavior may be due to a compensatory action by which a driver attempts to obtain a wider field of view [142]. In [142] a study of the correlation between eye and head movement was done, determining that the standard deviation of these two measurements could be suitable for detecting the state of cognitive distraction.

On the other hand, in [127] found that visual distraction resulted in frequent long off-road glances. Moreover, both cognitive and visual distractions caused gaze concentration and slow saccades when drivers looked at the roadway, and cognitive distraction increased blink frequency [126]. The study performed over the literature determined good measures of visual distraction are: eyes-off-road glance duration, head-off-road glance number, and standard deviation of lane position (SDLP).

Other physiological responses

In addition to eye-related responses observed in distracted drivers, research has revealed other patterns associated with distraction. In [143] was found that pupil dilation occurred when cognitive loads (such as conversation or where asked arithmetic questions) were imposed on the subjects. The phenomenon occurs due to the acceleration of the sympathetic nerve. They state that the heart rate also increased by approximately 8 beats per minute. However, the average value of the heart rate decreased under the same situation.

Other studies such as [100] demonstrate and reproduce the phenomenon in which performing a cognitively distracting secondary task (e.g. talking or thinking about something) while driving produced a decrease in the driver's temperature at the tip of the nose. Also, it was reported that considerable and consistent skin temperature increases in the supraorbital region could be observed during cognitive and visual distractions [219]. Other works [16] found that the electroencephalography (EEG) signal also contained information about the task engagement level and mental workload. In [165] conclude that the most sensitive measure

to mental workload seems to be the electrical heart activity, i.e. Electrocardiogram (ECG).

Effects of distraction in driving performance

Performing additional cognitive tasks while driving has been observed to lead to significant changes in vehicle control. Several studies have found that distraction may result in lapses in vehicle control, such as unintended changes in speed or drifting outside the lane boundaries.

Additionally, other effects of performing a secondary task that influence the lane-changing behavior include a reduction in the frequency of the checking behavior (check a side mirror or dashboard), a delay in the checking behavior, and a longer time to perform the checking behavior [129]. Some reports have found that the effects of cognitive distraction on driving performance considerably differed from the effects of visual distraction. On the one hand, visual distraction affects a driver's steering ability and lateral vehicle control, whereas, on the other, cognitive distraction affects longitudinal vehicle control, particularly car following. Moreover, cognitive distraction made steering less smooth but improved lane maintenance [126].

Visual distraction has been linked to steering neglect and overcompensation, while cognitive distraction has been associated with undercompensation [126]. Overall, visual distraction seems to interfere with driving performance more than cognitive distraction. Another intriguing and unexpected discovery is that when secondary task cognitive demands increased, a driver's ability to maintain lateral control improved [41]. It was also found by [85] an increased incidence of hard braking associated with cognitive distraction driving.

2.3.2 Signs of fatigue

One of the most common signs of fatigue is when drivers try to fight against sleep in an attempt to avoid having an accident. These signs, usually, involve different duration and sequences of the physiological events that precede the beginning of sleep [90]. When a driver becomes fatigued and begins to fall asleep, the following symptoms may be observed [121]:

- repeated yawning;
- confusion and thinking seems foggy;
- feeling depressed and irritable;
- slower reaction and responses;
- daydreaming;
- difficulty keeping eyes open and burning sensation in the eyes;
- lazy steering;
- difficulty maintaining concentration;
- swaying of head or body from nodding off;
- vehicle wandering from the road or into another lane;
- nodding off at the wheel;

- breathing becoming shallow;
- heart races.

Different individuals show different symptoms to varying degrees. Thus, there is no concrete method of measuring the level of fatigue. The European Transport Safety Council study [50] on diver fatigue established four levels of fatigue:

1. completely awake;
2. moderate sleepiness (drowsiness);
3. severe sleepiness;
4. sleep.

Most of the research found in the literature focuses attention on the second point above since it corresponds to the transition from awake to asleep (i.e. the drowsiness phase).

Furthermore, this study found that the degree of fatigue or sleepiness, as exhibited by an individual, is determined by the amount of activity relative to the brain's physiological capacity to stay awake. Several factors can influence this physiological waking capacity and, hence, lower the fatigue threshold such as sleep disorders, the low point in the circadian rhythm, and hard work before driving [50], [91]. These factors are independent of the activity being undertaken but result in the fatigue effect of that activity appearing more quickly. Therefore, fatigue cannot be seen simply as a function of the duration of time engaged in driving.

Driver behavior patterns of fatigue

When a driver is fatigued, certain physical and physiological phenomena can be observed, including changes in brain waves or EEG, eye activity, facial expressions, head nodding, body sagging posture, heart rate, pulse, skin electric potential, gripping force on the steering wheel, and other changes in body activities.

The most important signs of fatigue seen in drivers while performing their tasks have been studied in [175]:

- Drivers show a reflexive head nod after checking the side mirrors.
- The head motions were significantly less frequent.
- The number of times that the drivers touched or scratched their chin, face, head, ears, eyes, and legs significantly increased.
- Drivers were inclined to turn their heads to the left to relieve muscular tension in the neck.
- Eye-blinking activity radically increases.
- Episodes of yawning were more frequent.
- Drivers tend to adopt more relaxed hand positions on the steering wheel.

Many of these symptoms of fatigue come correlated with other context characteristics. In [195] have identified some of these characteristics:

- The time of the day (accidents due to fatigue driver reasons occur late at night [0:00 am-7:00 am] or during mid-afternoon [2:00 pm-4:00 pm])
- Presence of other vehicles (accidents due to drowsiness usually involve a single vehicle running off the road)
- Type of road (accidents occur on high-speed roadways)
- Presence of company (driver is often alone)
- Gender and Age of the driver (driver is often a young male, 16 to 25 years old)

Above all of these identified signs, for eye-blinking patterns, PERCLOS [117], which is the percentage of time that the eye is more than 80% closed, is one of the most widely accepted measures in the scientific literature for drowsiness detection. It has been validated using both EEG data and subjective evaluation.

Other physiological responses

Various physiological parameters have been studied to investigate their correlation with fatigue. In [121] the authors conducted a comprehensive investigation of the psychophysiology of driver fatigue, where they measured changes in brain activity, heart rate, and eye movements using electroencephalogram (EEG), electrocardiogram (ECG), and electrooculogram (EOG), respectively. Their results showed that low-frequency EEG activity ranging from 0 to 20 HZ is significantly associated with sleepiness. When drivers become drowsy, they often exhibit bursts of α activity in the central regions of the brain, specifically C3 and C4.

However, some people do not show any α activity. As the driver gets drowsier, the α activity is replaced by θ activity. When δ activity occurs in the EEG, the driver is no longer awake, which is an indicator of deep sleep [121]. Regarding heart rate, it was found a significant and large decrease in heart rate during driver fatigue. However, there are no firm conclusions on this matter. Additional non-visual features such as galvanic skin response (GSR) and the conductivity, steer gripping force have some kind of relation to the psychological state of the person [175].

Effects of fatigue in driving performance

Various studies have observed unusual driving behaviors that occur during drowsy periods. These behaviors can be identified using a variety of techniques and can be correlated to determine whether the driver is fatigued. These are some of the main signs identified [41]:

- Reduced frequency of steering reversals (each time the steering angle crosses zero degrees).
- Decline in steering performance.
- More frequent steering maneuvers during wakeful periods, possibly as compensatory behavior.

- Prolonged periods without steering correction, followed by abrupt or jerky motions during drowsy periods.
- Decreased steering velocity.
- Increased amplitude of steering-wheel movements.
- Higher standard deviations in the steering-wheel angle.

Works such as [228] found that the steering-wheel angle and vehicle tracking became irregular, and the range of deviation greatly increased during fatigued driving. Additionally, it was found that lane-tracking ability decrease as the time on the task increased. Variables such as square deviation of lane position (SDLP) and maximum lane deviation were found to be highly correlated with eye closures. The mean square of lane deviation, mean square of high-pass lateral position, and SDLP showed good potential as drowsiness indicators [41]. While there is little correlation between fatigue and braking or acceleration, vehicle speed variability has not been strongly correlated with drowsiness.

2.4 Conclusions

The scope of this thesis is to provide a comprehensive analysis of the physiological signs of driver fatigue and distraction, which can be monitored through cameras. To achieve this goal, extensive research was conducted on the relevant literature in psychology, human factors, and ergonomics, to identify the relevant patterns of human behavior that can help the design of effective estimation models and methods.

The research presented in this chapter has provided a thorough understanding of the different taxonomies and the underlying mechanisms of driver inattention. By focusing on the characteristics and aspects of fatigue and distraction, this study has identified the most relevant signs of these phenomena. Specifically, the study focuses on two key aspects of driver inattention that can be effectively sensed through cameras: the level of fatigue and the level of visual-manual distraction. These findings are crucial for the design of effective DMSs that can accurately detect and alert drivers of potentially dangerous situations.

The review of the physiological aspects of fatigue and distraction has contributed to the identification of the key data and methodological needs of DMS. The results of this analysis have been used as a foundational reference for the design of complex driver monitoring datasets and driver monitoring analysis methods, which will enable the development of advanced DMS capable of detecting different levels of driver inattention and providing timely alerts to prevent accidents. The insights gained from this chapter will be instrumental in the development of future research in this area and could be used to design more effective interventions to reduce the risk of accidents due to driver inattention.

Chapter 3

Eyelid aperture estimation method

3.1 Introduction

The shape of the human eye naturally differs among individuals, influenced by factors such as ethnicity, age, and gender. This variability poses a significant challenge when developing methods to assess driver states. Many DMS depend on the condition of the eyes to gauge the level of driver fatigue. Unfortunately, variations in eye shape can result in imprecise estimations of the eye state. These variations can arise from two main sources. Firstly, there is interpersonal variability, meaning variations in eye shape between different individuals. Secondly, there are intrapersonal variations caused by changes in the eye shape of an individual due to factors like lighting conditions or their state of arousal. Therefore, a method characterizing the palpebral fissure should determine the current eye aperture degree based on the maximum and minimum eyelid aperture levels for opened and closed eye states, respectively.

It is important to differentiate between *eyelid amplitude* and *normalized eyelid aperture*. In this thesis, *eyelid amplitude* is defined as a value obtained from the eye dimension ratio. These dimensions correspond to the width and height of the enclosing bounding box of the eye, measured in pixels. On the other hand, the term *normalized eyelid aperture* or simply *eyelid aperture* refers to the degree of eye openness. The eyelid aperture is expressed as a value ranging from 0 (closed-eye) to 1 (open-eye).

With a normalized eyelid aperture metric, it is possible to easily compare the eye state of different individuals. This metric is also useful for comparing the eye state of the same individual under different conditions. For example, it is possible to compare the eye state of a person under different lighting conditions or when the person is experiencing different levels of fatigue. In this chapter, a novel user-agnostic method for automatically extracting dynamic eyelid information is presented; enabling normalization of the eyelid aperture level.

3.2 Related work

3.2.1 Eye blink analysis methods

When it comes to detecting driver fatigue [113], [114], blink detection is the most widely utilized source of information. In fact, eye blink detection has become a standard method for assessing the driver's fatigue state in modern Driver Monitoring Systems (DMS) [79]. However, it is essential to acknowledge that environmental factors significantly influence the accurate definition of the eye state for blink detection. Various challenges and external elements, including lighting, light angle, head posture, and image quality, can have a substantial impact on the appearance and shape of the eyes. Consequently, accurately quantifying the eye state becomes difficult in such situations [198]. Scientific works on blink detection tackle different techniques which could be grouped into three categories: (i) appearance-based methods, (ii) motion-based methods and (iii) shape-based methods. Figure 3.1 shows a classification of the different methods.

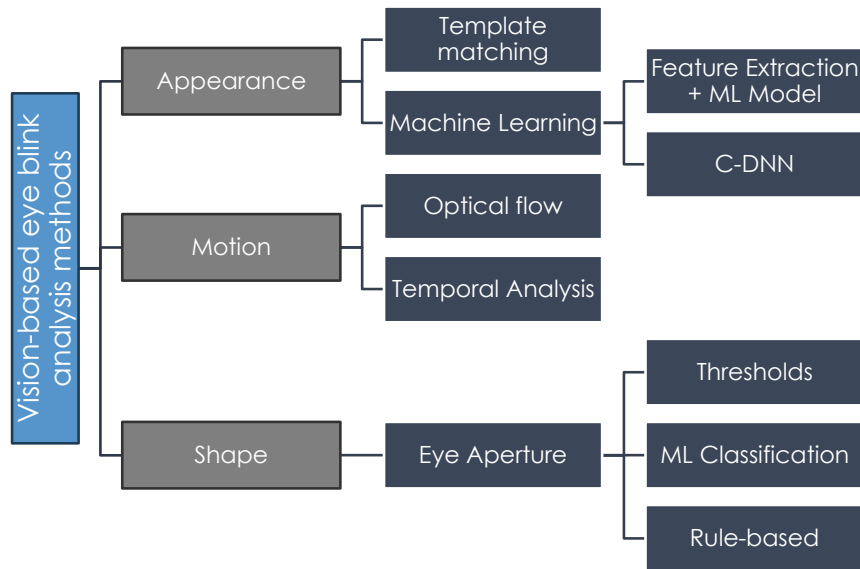


Figure 3.1: Classification of Vision-based Eye blink Analysis methods

Appearance-based methods determine the eye state by either using templates [78] or trained classifiers using machine learning [83], [133] techniques. Methods based on templates are simple and fast, but they are sensitive to illumination changes and head pose variations. On the other hand, machine learning methods are more robust to these variations, but they require a large amount of data to train the classifiers.

Previous publications on eye state recognition have proposed machine learning algorithms such as AdaBoost [61] and support vector machine (SVM) [87] to enhance the efficiency of recognition systems in unpredictable contexts. However, in addition to these machine learning methods, manual feature extraction techniques are typically employed to extract relevant features. It is worth noting that hand-crafted feature extraction approaches often incur significant computational overhead, resulting in slower systems and requiring a considerable amount of expertise and experience [110].

Moreover, CNNs have been also applied to the task of eye state recognition in images. A key advantage of CNN is its ability to integrate low-level features and automatically learn high-level abstract representations. This eliminates the need for manual extraction of representative characteristics from raw data. However, the limitation of appearance-based methods is that they typically provide a binary classification of the eye state, distinguishing between open and closed. While useful for basic eye state detection, this binary output may not be sufficient for accurately identifying incomplete eye blinks or determining the precise degree of eye openness. The ability to measure the level of eye openness is crucial for extracting driver drowsiness indicators, such as PERCLOS [108].

On the contrary, *motion-based* methods typically require to detect the face and eye regions within the image using statistical classifiers. Then, by applying optical flow strategies, the motion of the eyelids can be obtained [60]. Finally, by analyzing the eyelid movement, blinks can be detected. However, these methods also suffer from the same limitations as appearance-based methods, as they are also not able to determine the intermediate states of the eye aperture. Hence, limiting the applicability of these methods for fatigue detection.

Then, to overcome the limitations of the previous methods, *shape-based* methods have been proposed. These methods are based on the analysis of the shape of the eyelid region. Since the level of eyelid aperture is considered a reliable physiological indicator for determining the driver's state [36], leveraging this feature has yielded promising results in various studies [9], [164]. By accurately assessing the eyelid aperture level, more complex and discriminative indicators such as blink duration, blink frequency, and PERCLOS can be derived. Notably, PERCLOS is widely adopted in the literature [108] as a metric to determine the fatigue state of the driver.

Traditionally, shape-based blink detectors rely on thresholds for the eyelid aperture [186] to determine whether the eyes are closed or open. In addition, other approaches such as classification algorithms [199] or rule-based methods [9] have been utilized for blink detection in different studies.

For example, the method introduced in [9] relies on the eye geometrical ratio as input. The procedure for defining blink features involves analyzing the characteristics of blinks. A standardized set of steps is provided for regular blinks, with the consideration of some edge cases. However, the computation of certain design thresholds involves utilizing temporal windows spanning several minutes, which limits its applicability in continuous driving monitoring scenarios.

3.2.2 Eyelid aperture estimation methods

To determine the level of eyelid aperture, conventional methods typically involve extracting the contour of the eyelid borders and calculating an indicator that quantifies the extent of eyelid movement. Various techniques can be employed to derive the palpebral fissure contour, including image processing algorithms such as edge detection methods [35], [132], [203], curve fitting approaches [67], [222], the utilization of an Active Shape Model (ASM) for adjustment [151], [216], or a Regression Landmark Model [79]. Figure 3.2 shows a classification of the different methods.

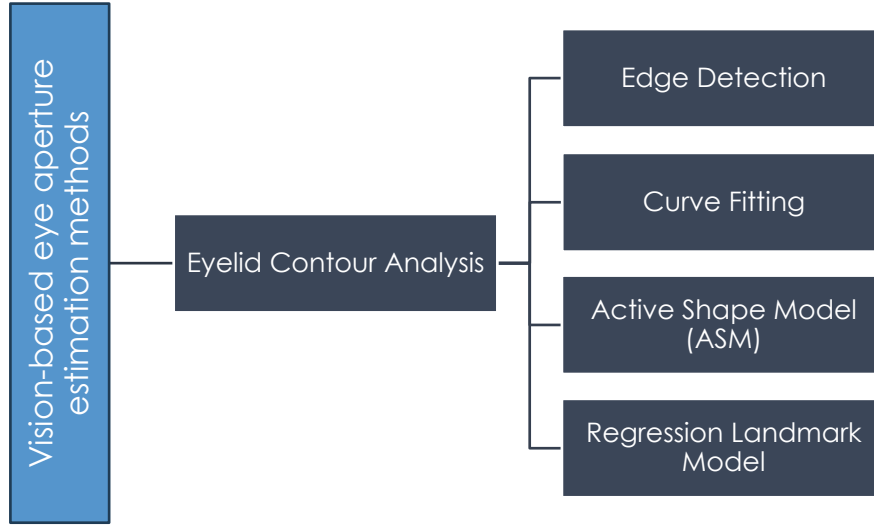


Figure 3.2: Classification of Vision-based Eyelid Aperture Estimation methods

Edge detection methods have been used to extract the eyelid contour. For instance, [203] and [132] proposed similar methods for detecting the position of the eyelid. Once the eye region is extracted, an accumulated signal is computed for each row of the eye image. Subsequently, the derivative of this signal is calculated. By identifying the local maximum and minimum points of this derivative, the locations of the eye edges are determined. The aperture level of the eye is then directly obtained by calculating the pixel difference between these two levels. Moreover, in [35], the eyelid edges are detected by applying two kernel filters for the upper and lower eyelids. Then, the eyelid aperture is estimated by calculating the distance between eyelids for both eyes.

In the case of curve-fitting approaches, the eyelid contour is approximated by a curve. For instance, [67] approximate the upper and bottom eyelid contours by fitting two intersected parabolic functions, one per eyelid border. The eyelid aperture is estimated by using the distance from the upper and lower eyelid curves. Similarly, in [222], a face tracker based on ASM is computed to obtain a first position of eye landmarks. Then the eye contour is refined by fitting a deformable template of two intersected parabolic sections to a distance map based on the distance of each pixel to the distribution of the skin colors. The final eye closure score is evaluated from the converged eye shape.

Methods that enable the extraction of facial landmarks are highly suitable and practical for real DMS solutions. These methods not only fulfill the requirement of detecting facial landmarks but also facilitate other functionalities, including blink analysis, head pose estimation, and refinement of gaze estimation [63], [77].

In [216] an 8-point eye deformable model is proposed. The eyelid aperture level degree is obtained by computing the ratio of the maximum vertical distance and the intra-ocular distance (IOD) for each eye. Eye blink detection is determined by applying a heuristic threshold determined by a set of evaluation face data. In [151], a 20-point eye deformable

model is fitted to the eye contour by using an optimization process. Then the eye aperture level is obtained and classified according to different templated blink categories using Hidden Markov Models (HMM).

In [79], a cascade regression framework is applied to simultaneously estimate image features from the eye center as well as contextual image features from eyelid and eye corners are jointly used to estimate the eye position and openness probability. Using the eye openness probability, the most likely eye state can be estimated.

All these methods usually extract a geometrical metric of the eyelid aperture. However, the eyelid aperture is a geometrical metric that is highly dependable on the individual physiological characteristics of the person. Then, these methods usually compute the *eyelid amplitude* value, but not the eyelid aperture level. In contrast, the method proposed in this work is based on a novel analysis of the normalized eyelid aperture level, which is a more robust metric for invariant blink detection analysis.

3.3 Definition of eyelid amplitude

Facial landmarks models are chosen to extract eye dimensions. Robust real-time facial landmark detectors found in the literature [8], [111] and open-source libraries such as DLib ERT [115], OpenFace [10] or MediaPipe [80] enable obtaining eye dimensions. Moreover, the facial landmarks information can be employed by other driver monitoring methods, such as head pose estimation and gaze estimation, reducing the computational overhead of algorithms in complex systems and achieving real-time integrated DMS applications.

Face alignment techniques determine the eye shape through a connected set of feature points. Consequently, measuring the eyelid amplitude is essential to obtain the final eye aperture level. Various methods for calculating eyelid amplitude from landmarks have been proposed in the literature.

In [201] the amplitude is measured as the mean distance between vertically corresponding landmarks. Similarly, in [69] the eye amplitude is defined as the height between eyelids. However, these methods will not tolerate changes in image scale. In contrast, other authors [133], [199] suggest using scale-independent metrics where the measure involves using a ratio of vertical and horizontal distance.

Moreover, in [9], eye closure is determined by calculating the ratio between the vertical distance between the eyelids and the fixed diameter of the iris. While this method can be effective, it requires a calibrated camera to obtain accurate measurements of the eye. Unfortunately, in many DMSs, camera calibration is not possible due to various practical constraints.

In this work, the eye amplitude A_t is proposed to be defined as the eye aspect ratio (EAR) between height and width. The eye contour landmarks provided by the facial landmark model are used, and the EAR is computed based on the maximum height H_t and width W_t of the contour formed by the facial points, as illustrated in Figure 3.3.

The shape of the eye can be approximated as either rectangular or elliptical, with the width

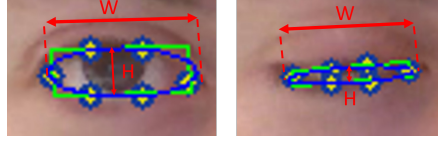


Figure 3.3: Eye landmark fitting and estimation of the height (H) and width (W) of the eye

being greater than the height. Therefore, to obtain values closer to one when the EAR is at its maximum, we propose using double the EAR value as the input for the proposed method (as shown in equation 3.1). This approach enables better capturing of the variability in eyelid fissure height between individuals of different ethnicities.

$$A_t = \min \left(1, \frac{2H_t}{W_t} \right) \quad (3.1)$$

The eye amplitude A_t reaches saturation at 1 when the eye height is half its width, a situation that may arise for very round eyes. Depending on an individual's physiological state and facial characteristics, the nominal amplitude levels for opened and closed eyes can vary significantly. Figure 3.4 demonstrates this variability, showing that different individuals exhibit distinct maximum and minimum eye amplitude A_t values.

3.4 Normalized aperture estimation.

In this work, a method is introduced to obtain a person-invariant metric of the eyelid aperture. The objective is to normalize the eyelid amplitude A_t value (eq. 3.1) to derive an aperture level that is robust to variations in subject facial characteristics. The method adapts as new values of eye amplitude are obtained. Different signals and metrics are updated on each iteration. This section will explain the proposed methods step by step in the following paragraphs. A summary of the proposed method is presented in Figure 3.5.

The computation of the normalized eyelid aperture $A_{n,t}$, for each time frame t is achieved using an online probabilistic approach, which computes the posterior probability of the event where the eye is open $E_{o,t}$ and closed $E_{c,t}$. Then the normalized eyelid aperture is approximated as $A_{n,t} = P(E_{o,t}|A_t)$, as the probability of the eye being open given the eye amplitude A_t .

Using the Bayesian formulation, the following expressions can be derived:

$$P(E_{o,t}|A_t) = \frac{p(A_t|E_{o,t})P(E_{o,t})}{P(A_t)}; \quad (3.2)$$

$$P(E_{c,t}|A_t) = \frac{p(A_t|E_{c,t})P(E_{c,t})}{P(A_t)} \quad (3.3)$$

where $p(A_t|E_{o,t})$ and $p(A_t|E_{c,t})$ are the probability density functions that represent the likelihood of observing the eye in open and closed states, respectively.

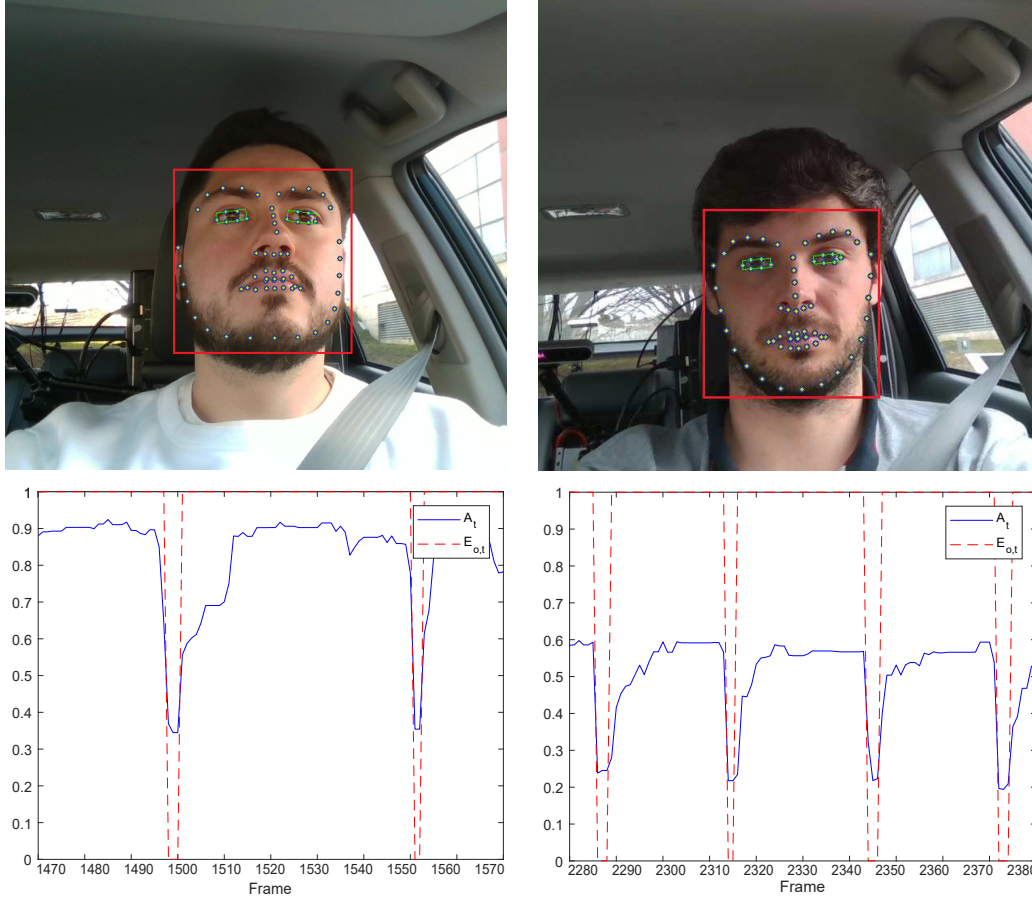


Figure 3.4: Eyelid amplitude variations for two individuals blinking normally. In the top row, facial landmark extraction for each individual is presented. The bottom row displays the graphs corresponding to their respective eye amplitude A_t , calculated as in eq. 3.1. The signal $E_{o,t}$ denotes the event when the eye is open.

$P(E_{o,t})$ and $P(E_{c,t})$ are the prior probability of each event for the eyes to be open and closed respectively. The prior probability plays an important role in shaping the outcome of the analysis. In this work, the prior probabilities are set to represent equal prior probability of the eyes to be open and closes as the method is designed to be user-agnostic and does not have any prior information about the user's eye state. This selection of this prior probability is based on the assumption that the initial state of the eye could be either open or closed. Future analysis could focus on adapting these probabilities to the case where the assumption could be that the eyes are more likely to be open than closed during the driving task.

$P(A_t)$ is the evidence, a normalization factor to ensure the sum of probabilities of open and closed eyes is 1, $\sum_{s \in \{o,c\}} P(E_{s,t}|A_t) = 1$, as we only have these two events.

Then by applying this principle to Eq. 3.2 and Eq. 3.3, this factor can be obtained as:

$$P(A_t) = \sum_{s \in \{o,c\}} p(A_t|E_{s,t})P(E_{s,t}) \quad (3.4)$$

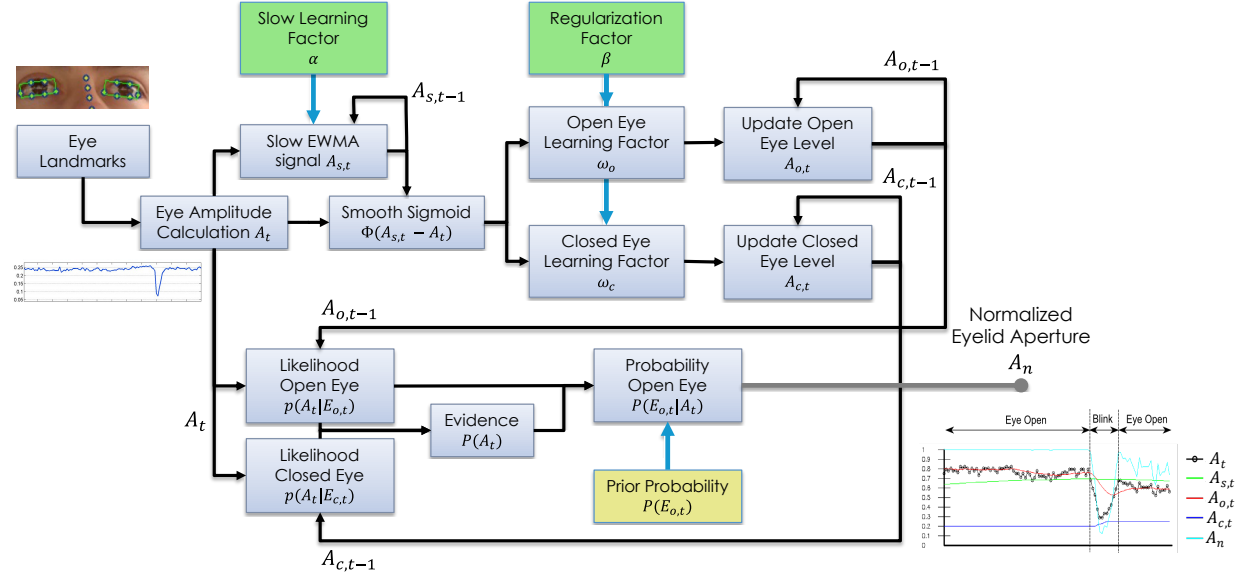


Figure 3.5: Eyelid aperture normalization method. The input of the method is the eyelid amplitude ratio A_t . In the process, different accumulated mean signals are updated. Finally the normalized eyelid aperture $A_{n,t}$ is computed using a probabilistic approach based likelihood of the input signal to be an open eye. The parameters α and β are input parameters of the proposed method.

The likelihood models for observing the eye in the open and closed state are derived from two balanced distributions, truncated at their extremes:

$$p(A_t|E_{o,t}) = \omega_g(A_t)g(A_t|A_{o,t-1}; Var(A_{o,t-1})) + \omega_u(A_t)u(A_t|A_{o,t-1}, 1) \quad (3.5)$$

where $g(A_t|A_{o,t-1}; Var(A_{o,t-1}))$ is the normal distribution with mean equal to $A_{o,t-1}$ and variance equal to the variance of $A_{o,t-1}$; and $u(A_t|A_{o,t-1}, 1)$ is a uniform distribution in the interval $(A_{o,t-1}, 1)$, scaled to $g(A_{o,t-1})$. The factors ω_g and ω_u are step functions which determine the application of functions g and u , respectively: $\omega_g(A_t) = 1$ for $A_t \leq A_{o,t-1}$ and $\omega_u(A_t) = 1$ for $A_t > A_{o,t-1}$. The likelihood of A_t of event $E_{c,t}$, $p(A_t|E_{c,t})$ can be expressed analogously.

The rationale behind this approach is that when a new observation is made, the likelihood to be open would not be penalized (i.e. decrease) when A_t is larger than $A_{o,t-1}$. Figure 3.6 illustrates the shape of the likelihood of the eye being open.

To obtain the likelihood models, estimation of $A_{o,t}$ and $A_{c,t}$ values is essential. Here, $A_{o,t}$ represents the eyelid amplitude of the driver when the eye is fully open, while $A_{c,t}$ denotes the eyelid amplitude when the eye is completely closed. These values are user dependent then they need to be adapted when specific eyelid information of the user is available. Initially, these values are set to standard defaults and subsequently refined by incorporating new A_t values specific to each user.

The process of updating the values of $A_{o,t}$ and $A_{c,t}$ introduces recursion. To address this, it is

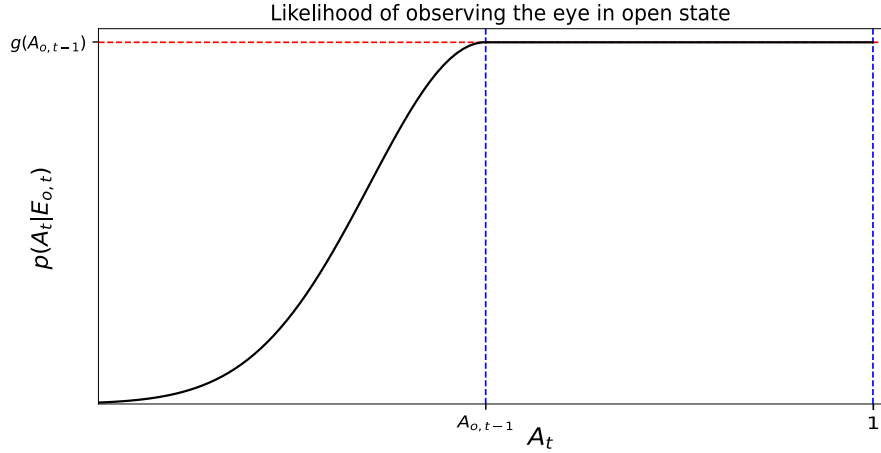


Figure 3.6: Representation of likelihood of the eye to be open by a combination of a normal distribution $g(A_{o,t-1})$ and a uniform distribution $u(A_t|A_{o,t-1})$

proposed to estimate these values using Exponential Weighted Moving Averages (EWMA) [66]. This approach allows the incorporation of new data while maintaining a degree of influence from previous samples. By employing this method, the update of signals $A_{o,t}$ and $A_{c,t}$ becomes smooth and gradual. Then the update of $A_{o,t}$ and $A_{c,t}$ is defined as:

$$A_{o,t} = \omega_o A_{o,t-1} + (1 - \omega_o)A_t \quad (3.6)$$

$$A_{c,t} = \omega_c A_{c,t-1} + (1 - \omega_c)A_t \quad (3.7)$$

The adjustment of the EWMA learning factors, ω_o and ω_c , takes place at each frame based on a function that assesses the local variability of the signal within the temporal window. These learning factors are not static but dynamically defined to enhance the influence of a new measurement A_t based on its proximity to $A_{o,t-1}$ and $A_{c,t-1}$. Specifically, when A_t is close to $A_{o,t-1}$, the impact on the update of $A_{o,t}$ becomes more significant by decreasing ω_o . Figure 3.7(c) show this variability of the learning factor when A_t varies in a time interval. According to the proposed formulation, when the learning factors are close to 0, the influence of the previous value is higher.

To find a method to update these learning factors a new signal $A_{s,t}$ is proposed, which is the EWMA of measurement A_t .

$$A_{s,t} = \alpha A_{s,t-1} + (1 - \alpha)A_t \quad (3.8)$$

where α is the averaging factor of $A_{s,t}$.

Under the hypothesis that the time eyes are open is higher than the time eyes are closed, then $A_{s,t}$ is always closer to $A_{o,t}$ than to $A_{c,t}$ (see Figure 3.7(a)).

Therefore, $A_{s,t}$ can be used to define the value of ω_o . A way to implement this idea, and also provide a mechanism to define ω_c is to create a sigmoid function (which returns a value between 0 and 1) on the difference between A_t and $A_{s,t-1}$ (higher values of this sigmoid

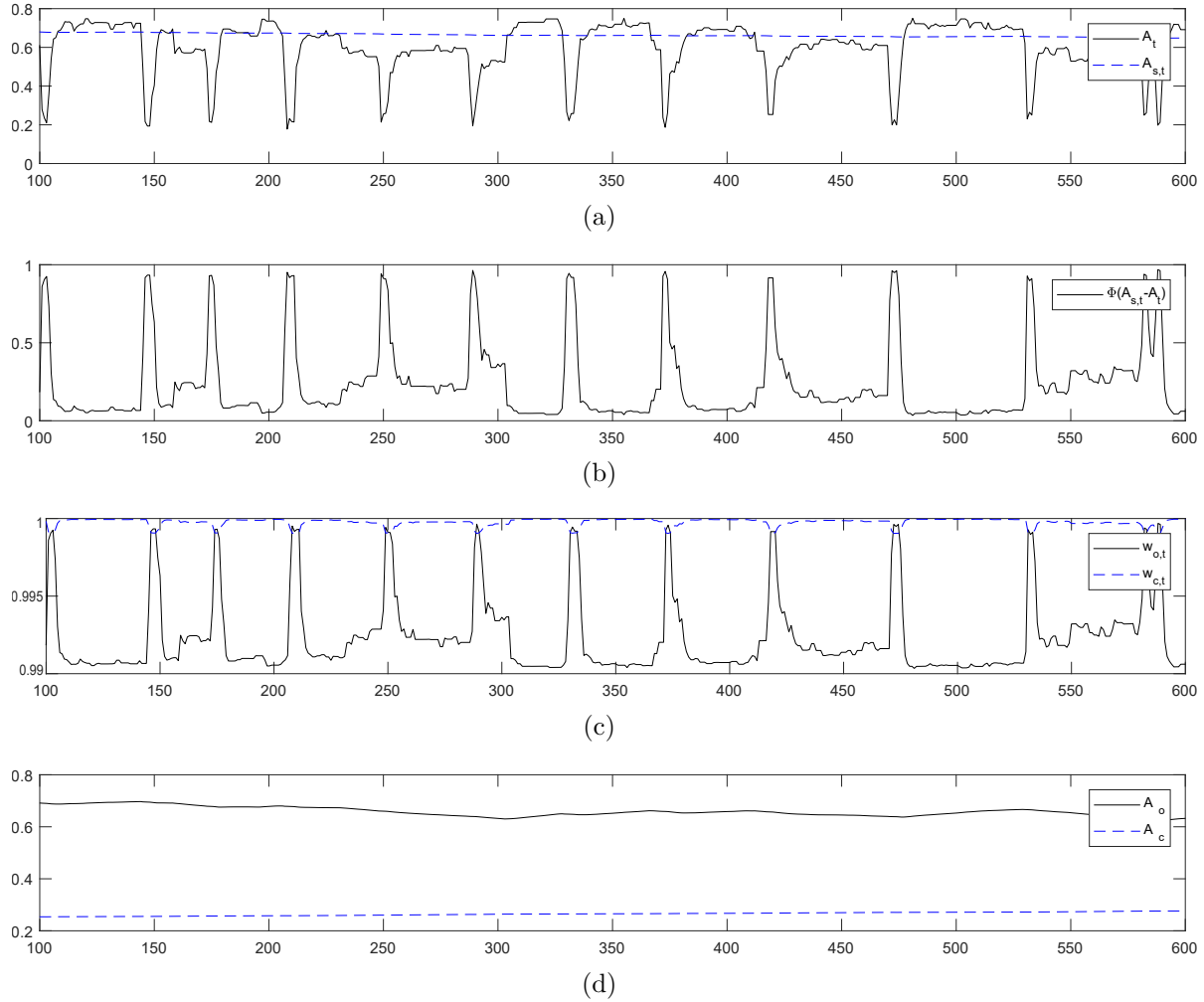


Figure 3.7: Values of the parameters involved in the computation of the normalized aperture for a sample sequence. (a) Input signal A_t and EWMA $A_{s,t}$. (b) Sigmoid function $\Phi(A_{s,t} - A_t)$. (c) Learning factors w_o and w_c . (d) EWMA of the open and closed eye signal $A_{o,t}$ and $A_{c,t}$

correspond to situations the eye is more likely closed, and lower values correspond to open eye measurements). The sigmoid function is defined as:

$$\Phi(A_{s,t} - A_t) = \frac{1}{1 + e^{-a(A_{s,t} - A_t - c)}} \quad (3.9)$$

where variables a and c can be selected to make the sigmoid function be centered at $c = (A_{o,t} - A_{c,t})/2$ (i.e. the expected mid-way between the eye amplitudes at open and closed states), and to reach a significant value at the maximum possible difference, e.g. $\Phi(A_{o,t} - A_{c,t}) = 0.95$ (note the sigmoid function asymptotically approaches to 1 but without never reaching it):

$$a = -\frac{\log\left(\frac{1}{\Phi(A_{o,t}-A_{c,t})} - 1\right)}{A_{o,t} - A_{c,t} - c} \quad (3.10)$$

To illustrate the behavior of the sigmoid function, Figure 3.8 depicts the sigmoid curve for a specific set of parameters. It should be noted that the sigmoid function is centered at $c = (A_{o,t} - A_{c,t})/2$ and attains a value of 0.95 at $A_{o,t} - A_{c,t}$.

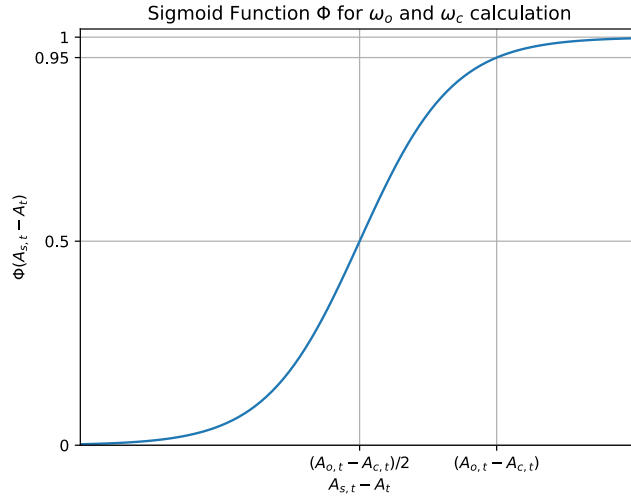


Figure 3.8: Representation of the sigmoid function $\Phi(A_{s,t} - A_t)$, used to calculate the EWMA learning factors ω_o and ω_c .

Figure 3.7 illustrates the progression of the functions involved in computing the normalized aperture. The first row (Figure 3.7(a)) showcases the comparison between the input signal A_t and the calculated $A_{s,t}$. It is important to note that $A_{s,t}$ remains stable throughout with slight variations over time. Additionally, it consistently remains close to the upper level of the signal, indicating the situation when the eyes are open.

Moreover, in the second row (Figure 3.7(b)) the values of the sigmoid $\Phi(A_{s,t} - A_t)$ range from 0 to 1, following the variability of A_t . In practice, this variability is counterproductive for an EWMA learning factor (i.e. it makes the EWMA not smooth). Therefore, the learning factor update equation needs to be regularized as follows:

$$\omega_o = \beta + (1 - \beta)\Phi(A_{s,t} - A_t) \quad (3.11)$$

$$\omega_c = \beta + (1 - \beta)(1 - \Phi(A_{s,t} - A_t)) \quad (3.12)$$

On Figure 3.7(c) the learning factors ω_o and ω_c are depicted. It can be observed that when the learning factor for the open eye signal ω_o is close to 0, the signal is close to A_o , and close to 1 when the signal is close to A_c . This shows the adaptability of the learning factors when the situation is most likely to be an open eye or a closed eye. The opposite behavior is observed for ω_c .

In addition, it is important to highlight that the learning factor ω_o changes rapidly to avoid introducing small eye aperture values in the open eyes EWMA signal. This is due to the

fact that the EWMA of the open-eye signal is used to compute the normalized aperture. Therefore, the learning factor ω_o is used to avoid the EWMA of the open eye signal being affected by small eye aperture values. These learning factors lead to smoother evolution of $A_{o,t}$ and $A_{c,t}$. Parameter β is a user-defined parameter that balances the impact of Φ .

Furthermore, the last row of Figure 3.7 shows the evolution of the EWMA of the open eye signal $A_{o,t}$ (Figure 3.7(d)). It can be observed that the EWMA of the open eye signal adapts to the input signal A_t and remains close to the upper level of the signal, indicating the situation when the eyes are open. Signal $A_{o,t}$ is always higher than $A_{c,t}$, which is the expected behavior of the EWMA of the open-eye signal.

Finally, Figure 3.9 illustrates the global evolution of the computed signals on a sample 500 frames sequence. As it can be observed, the EWMA $A_{s,t}$ is slowly learning the average of A_t , while $A_{o,t}$ and $A_{c,t}$ adapt to the observed open and closed-eye amplitudes. For this example, the following constants were used: $\alpha = 0.999$ and $\beta = 0.99$. The normalized eyelid aperture is obtained by applying $A_{n,t} = P(E_{o,t}|A_t)$ as described in equation 3.2

By observing Figure 3.7 and Figure 3.9 it is possible to see that Φ determines how likely the measurement belongs to the open and closed states, and the learning factors ω_o and ω_c are updated according to Φ . In other words, the average closed-eye amplitude $A_{c,t}$ is updated with significant weight, assigned to the current measurement A_t proportionally to ω_c , which corresponds to the situations where the eye is likely closed.

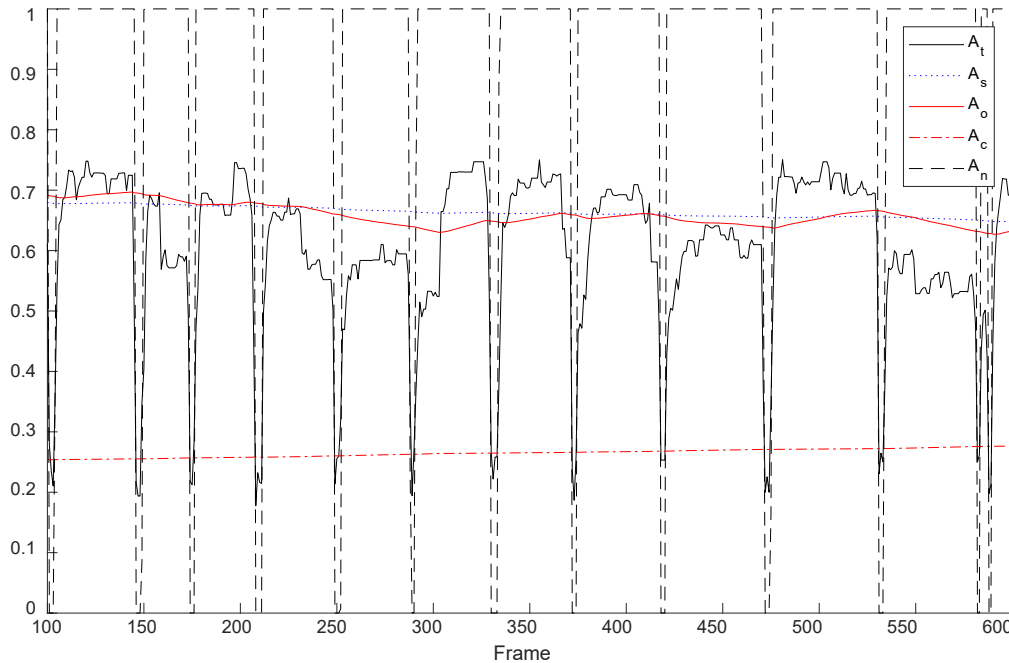


Figure 3.9: Sample evolution of eyelid closure A_t , EWMA $A_{s,t}$, open and closed-eye estimated amplitudes $A_{o,t}$ and $A_{c,t}$, and normalized aperture $A_{n,t}$ in a 500 frame sequence. The value of $A_{n,t}$ is obtained by normalizing A_t with $A_{o,t}$ and $A_{c,t}$.

3.5 Parameter learning

3.5.1 Manual parameter selection

The two design parameters of the proposed method are α and β . On the one hand, α determines how fast the EWMA $A_{s,t}$ learns from observed measurements, while β determines the amount of impact function Φ can have on the estimation of the amplitude values for open and closed-eye states, $A_{o,t}$ and $A_{c,t}$.

It is noteworthy to mention that the learning parameters of the EWMA expressions are inversely proportional to the learning speed of the function, i.e. values closer to 1 (e.g. 0.9999) express slower learning rates than smaller values. Therefore, their selection is critical to get the expected behavior.

We can select values for these parameters by defining what is the expected learning period for the estimated magnitudes. For that purpose, the EWMA equation (eq. 3.8) can be rewritten as time series:

$$A_{s,t} = \alpha^t A_{s,0} + (1 - \alpha) \sum_{i=1}^t \alpha^{t-i} A_i \quad (3.13)$$

From this expression, and considering an extreme case where EWMA is initialized to 1.0, and then all subsequent measurements are 0.0, the equivalent time period to decrease $x\%$ can be expressed as:

$$T_x = \frac{\log(1 - x)}{\log(\alpha)} \quad (3.14)$$

and reversely,

$$\alpha = \exp\left(\frac{\log(1 - x)}{T_x}\right) \quad (3.15)$$

This equation can be used to obtain a guess on the required value of the learning parameter α for a certain period, e.g. T_{95} . For instance, α should be at least 0.999 to get a period of about 1000 frames, which corresponds to 40 seconds at 25 fps, because the average value of the eyelid amplitude $A_{s,t}$ should not change faster than that (physiologically, average eyelid amplitude changes slowly due to fatigue factors [193]).

A similar procedure can be done over eq. 3.11 to obtain an estimation of β . Therefore, β should be around 0.99 to get faster adaption ($T_{95}(0.99) = 50$) to the expected value of open and closed-eye amplitudes, $A_{o,t}$, $A_{c,t}$, which can dynamically change due to face gestures, gaze patterns (e.g. looking to the dashboard), etc.

3.5.2 Parameter training

However, to improve the adaptability of the method to the data of each user, the selection of the values for α and β should be done automatically. To achieve this, it is proposed to define a cost function and using a training set that covers a variety of subjects and blinking situations, determine the parameter values automatically.

Let's consider $E_{o,t}$, a binary signal whose value is 1 and 0 when the eye is open and closed, respectively, and $\mathcal{O}$ the set of time indexes for which $A_{n,t} > 0.5$ (i.e. the method is classifying the eye as open) and $\mathcal{C}$ the set of tie indexes for which $A_{n,t} \leq 0.5$ (i.e. classified as closed). Then, the cost function is defined, which penalizes the errors between the predicted normalized eyelid aperture and the ground truth labels. The objective is to obtain the values for α and β that minimize the following cost function:

$$J = \frac{1}{|\mathcal{O}|} \sum_{t \in \mathcal{O}} \rho(A_{n,t}, E_{o,t}) + \frac{1}{|\mathcal{C}|} \sum_{t \in \mathcal{C}} \rho(A_{n,t}, E_{o,t}) + \frac{1}{|\mathcal{O} \cup \mathcal{C}|} \sum_t \gamma(A_{n,t}, E_{o,t}, \tau) \quad (3.16)$$

where $|\mathcal{O}|$ and $|\mathcal{C}|$ are the cardinalities of sets $\mathcal{O}$ and $\mathcal{C}$ respectively, and $\rho()$ is a M-estimator of the squared difference:

$$\rho(A_{n,t}, E_{o,t}) = \begin{cases} (A_{n,t} - E_{o,t})^2 & , \text{if } |A_{n,t} - E_{o,t}| < \epsilon \\ \epsilon^2 & , \text{otherwise} \end{cases} \quad (3.17)$$

where ϵ can be selected as a suitable maximum expected error of a classifier (e.g. $\epsilon = 0.3$) so that larger errors are considered outliers by the M-estimator.

The factor $\gamma()$ provides temporal smoothness to the measurement, by applying a running window factor with length τ (i.e. $\tau = 5$ frames):

$$\gamma(A_{n,t}, E_{o,t}, \tau) = \frac{1}{2\tau + 1} \sum_{i=t-\tau}^{i=t+\tau} \rho(A_{n,i}, E_{o,i}) \quad (3.18)$$

For the training process, a labeling process was performed on 12 video sequences 5000 frames long each (60000 labeled frames), of 3 subjects with different physiological facial attributes (and thus different eyelid amplitude values), each of them in 4 different blinking states: (awake blinking, no-blinking, long blinks, drowsy). The sequences were captured using a real vehicle (see Figure 3.4 for examples). The amplitude value A_t is obtained using the face alignment method in the literature [111].

To capture the distinctions between users, the cost maps for each subject were calculated, as depicted in Figure 3.10. These maps were generated by summing the costs of all sequences for each subject. It can be observed that while the shapes of the cost maps vary slightly, they exhibit similar minimum values, except for subject 3. These differences are due to the different eye shapes and possible eyelid asymmetries of each subject. However, using the proposed method it is possible to obtain a personalized suitable value for α and β for each subject.

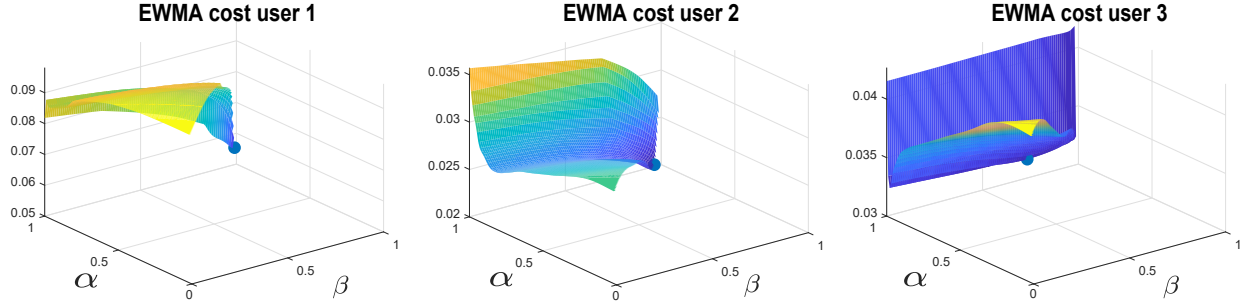


Figure 3.10: Map of cost function J for each user and for α and β values spanning from 0.0 to 1.0 in steps of 0.01

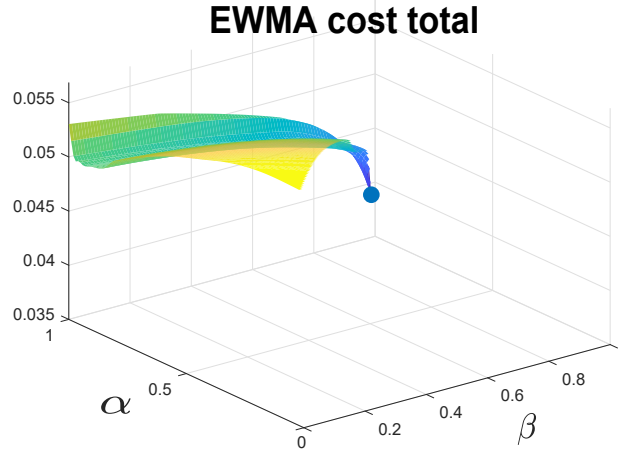


Figure 3.11: Map of the cost function J for α and β values spanning from 0.0 to 1.0 in steps of 0.01

For an online version of the algorithm, it is advantageous, to begin with global parameters. As more data becomes available, the algorithm can progressively refine the parameters for each user using the proposed method. This approach enables personalized adjustments to accommodate the specific characteristics of each user, thereby enhancing the algorithm's performance.

Moreover, the cost values for α and β from 0 to 1 are collected, and as a result, the cost map summed to all sequences is illustrated in Figure 3.11. The minimum cost is obtained when α gets closer to 1.0, and β is kept behind at around 0.98. This is well aligned with the approximate values reasoned in section 3.5.1. However, this automatic process allows a better fine-tuning adjustment of the parameters to real driving sequences.

3.6 Experiments

3.6.1 DMD annotation for eyelid aperture estimation

One of the contributions of this thesis is the creation of a comprehensive dataset for driving monitoring, the DMD. The full description of the DMD creation process is presented in

Chapter 4.

To evaluate the eye aperture estimation method, a subset of the DMD recordings was selected for testing purposes. As one of the key objectives of this work is to contribute to the annotation and validation of videos from the DMD dataset, the experiments involved two main steps. Firstly, the recordings of drivers were manually annotated. Following that, the proposed method was applied to evaluate blink detection on these annotated recordings. This sequential approach allowed for the assessment of the method's performance and its effectiveness in accurately detecting blinks in driver videos.

The DMD is intended to include spatial and temporal annotations; that is, from landmarks of the face to frame intervals that contain an action. To use the material related to fatigue, a list of classes with annotation criteria for these video sequences was used, contributing to the continuous development of the DMD.

To enhance scene description, the proposed approach involves the application of multi-label annotation. This approach enables the comprehensive description of various aspects within the same annotation file of a video. Multiple elements are targeted for annotation, including different cases of "Yawning" (with hand and without hand, as defined in the DMD dataset), the eye state of each driver, and the annotation of camera occlusions. All of these aspects are annotated through frame intervals, allowing for a detailed understanding of the scene dynamics.

In Table 4.3 of Chapter 4 the defined categories or levels of annotation along with their corresponding labels are presented. Notably, the annotation of eye states is crucial for conducting experiments to evaluate the proposed eyelid aperture estimation method. By incorporating multi-label annotation, a comprehensive understanding of the scene and accurate eye-state information can be captured, facilitating the evaluation of the proposed method.

To reach an agreement between annotators and reduce ambiguity in the annotations, an annotation criterion is proposed. That is the explanation of each of the labels and the conditions in which a frame should be labeled as such. The definitions of the eye states adopted are described below:

- **"Open"**: The driver has his/her eye open and this does not change in time. This means that this state is a stationary state in which the eye aperture is constant or stable. Comes after an "Opening" state.
- **"Close"**: The driver has his/her eye closed, meaning that, from the frontal camera, there is not a pixel belonging to the interior of the eye that is visible. This is also a stationary state in which the eye remains closed. Comes after a "Closing" state.
- **"Opening"**: This is a transitory state in which the eyelid moves upwards. After a "Close" state, this one starts when the interior of the eye is visible. After a "Closing" state, it starts when the eyelid stops going down to go up. Finishes when the upper eyelid has completed its upwards trajectory, passing to the next state of "Open", or when the eyelid starts going down, leaving to the next state of "Closing".
- **"Closing"**: This is a transitory state in which the eyelid moves downwards. After an

"Open" state, this one starts when the eyelid starts to go down. After an "Opening" state, starts when the eyelid stops going up to go down. Finishes when the upper eyelid has completed its downward trajectory, passing to the next state of "Close", or when the eyelid starts going up, leaving to the next state of "Opening".

- **"Undefined":** This occurs when the eye cannot be captured correctly from the front camera and one of the previously defined states can't be assigned. It usually occurs when the driver turns his head, taking the eye off the camera view. Also, when occlusion of the eye occurs; either due to the driver's hand or an object interfering between the eye and the camera or when lighting conditions (e.g. sun-ray) affect the camera and it is unable to capture a recognizable image.

Within the dataset video sequences, people do not blink in the same way, as it is natural. There are complete blinks in which the flow of states is normally performed: "open"- "closing"- "close"- "opening"; however, people's blink pattern is not perfect and sometimes they do not close their eyes completely, so the "close" state is not included in the blinking cycle. These differences in blinking patterns make annotating eye states a complex task, hence the relevance of defining annotation criteria.

As an extension of the DMD, a tool for temporal annotations called TaTo (Temporal Annotation Tool) was developed. This software allows making multi-label temporal annotations, providing the option of annotating time intervals by blocks or frame-to-frame. It offers a simple and intuitive interface (see Figure 3.12), showing the annotations on a timeline through colors, and offers easy keyboard navigation. TaTo is public to the scientific community and to anyone who wants to use the DMD, create and/or change its annotations. Besides, it is possible to adapt the tool to perform temporal annotations in other domains outside of the DMD. To evaluate the eyelid aperture estimation method the tool was adapted and used to create the ground truth necessary for this evaluation, making temporal annotations of the DMD's drowsiness-related material. The DMD annotations are in VCD (Video Content Description) format ¹. This is an open description language that supports spatial and temporal annotations for describing objects, actions, events, contexts and relations, of scenes or data sequences. This is the first annotation toolset compliant with the definitions of the ASAM OpenLABEL [6] standard.

Recordings for both male and female drivers were selected to have a balanced evaluation set. Recordings from a total of 4 individuals were selected, being 2 female and 2 male. The recording of the DMD contains sequences of drowsy driving in which the driver shows patterns of increased periods of closed eyes, yawning and microsleeps. In addition, in other sequences, the driver is performing common actions (using cellphone, combing hair, drinking, etc), some of which could imply some form of distraction. In these samples, the drivers are fully awake and rested. To validate the eyelid aperture estimation method, recordings from both drowsy and actions data were selected. A total of 15000 frames were used from the drowsy sequences and 12000 frames from driver actions.

¹<https://vcd.vicomtech.org/>

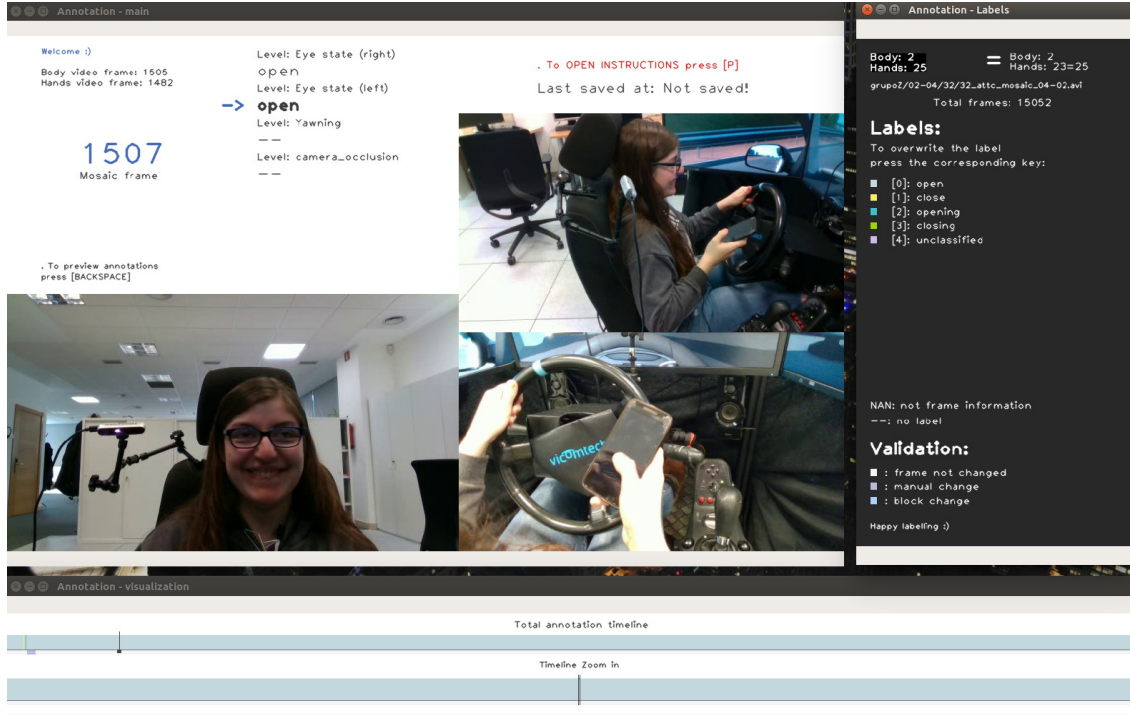


Figure 3.12: Temporal Annotation Tool (TaTo) for annotating eye states of the DMD videos

3.6.2 Results and discussion

To validate the benefits of normalizing the eyelid amplitude, the normalized eyelid aperture $A_{n,t}$ is computed and a threshold of 0.8 is defined to determine binary blink events. Since the signal is normalized for each user, the threshold applies to all the tested sequences and will not suffer accuracy loss when detecting blinks [186]. This value was set to correctly include the closing and opening phases of the blinks (i.e. eye states with eyelid aperture lower than 80% are considered as blinks).

Ground truth annotations on blink patterns in the driving context are not easily available. Therefore, an initial experiment was performed before the DMD dataset was available. Sequences of three users and two arousal states with different blinking patterns (awake and drowsy) were collected. Test sequences were obtained in simulated driving conditions at different times of day, with volunteers inside a real car. A total of 10000 frames were captured for each user. Manual labeling of the open and blinking states was performed on the sequences.

The first evaluation was done based on a binary classification approach. The aim is to compare the proposed method's ability to correctly classify the eyes as open or closed. Therefore, two eye states (classes) were defined: open and closed (blink). Accuracy values were computed for both classes (see Table 3.1). Other related work results are included for comparison. These methods use different evaluation datasets which were not available for evaluation. However, our initial testing set shares similar characteristics with their data which makes the experimentation representative and comparable.

The results show lower accuracy for the awake sequences, especially when classifying closed

Accuracy (%)	Open	Closed	All
Sukno et. al. [201]	99.5	80.5	97.1
Qin et. al. [167]	97.0	88.7	91.6
Gou et. al. [79]	-	-	91.4
Ji et. al. [106]	96.8	96.2	97.6
Proposed Method (Awake)	97.3	92.1	97.8
Proposed Method (Drowsy)	99.1	95.9	98.9
Proposed Method (Total)	98.5	95.3	98.1

Table 3.1: Comparison of frame classification accuracy between open and close states using the proposed method: selection of α and β with the lowest average cost

(blink) eyes. These errors are mainly due to the fast transitions between open and closed eyes in normal blinks and the capturing rate of the camera ($\approx 30fps$). In these situations, one or two-frame error has a greater impact on the accuracy values compared to the drowsy sequences, where blink intervals are longer. Nevertheless, the overall accuracy is higher than other results reported in related state-of-the-art methods

Once the DMD dataset was available a second evaluation was performed which expands the first evaluation to a new dataset which is more challenging as the variability of illumination conditions and driver actions is richer. Although the dataset used in the first evaluation was collected in a driving scenario, the driver’s head movements were much more controlled and the illumination conditions were clearly stable. Therefore, using the DMD dataset results in a more challenging scenario.

This second evaluation aims to assess correctly the classification of open and closed states of the driver’s eyes. The defined method computes a normalized eyelid aperture which is a value between 0 and 1, being 0 totally closed and 1 totally open. The benefit of this method is the adaptability to the eye shapes of different drivers. Therefore, it is possible to apply a simple closing threshold (e.g. eyelids below 80% are considered closed) to extract open or closed eye states. In addition, a recent method for eye blink detection was implemented which was presented in [9] to compare with the proposed method in this thesis and the robustness against the DMD dataset. The method proposed by Baccour et al. [9] generates blink segments that were treated as closed eye intervals for the purpose of this evaluation. This approach enables the application of the same evaluation methodology as used for the proposed method. By considering the blink segments as closed eye intervals, a consistent evaluation framework can be established for comparing the performance of both methods.

The classification results of open and closed eyes are presented in Table 3.2. As it can be seen the proposed algorithm gets better accuracy results than other state-of-the-art methods. The overall results beat the 90% accuracy reference even for a challenging dataset such as DMD, confirming the proposed method can be applied to real scenarios. The F1-score values show that there is still room for performance, however, the results are valid for implementation in

combination with other DMS modules. Actually, this method was successfully integrated into a real DMS system to feed advanced decision support methods in autonomous driving (see Section 6.5).

Moreover, the influence of different types of data was also analyzed to identify possible weaknesses. In general, better results are obtained from the sequences of drowsy drivers. In these videos the time with closed eyes is larger and generally, the persons tend to reduce head movements. Therefore the detection of the face and eyes is better. On the contrary, in the sequences of actions while driving, since the drivers are performing several activities, some form of noise is introduced in the input signal, therefore the lower results are consistent. An analysis according to driver gender was also performed. From the results, it is not possible to conclude there is any correlation between gender and open/closed eye classification.

Method	Data	Accuracy	Acc. Open	Acc. Close	F1-Score
Baccour et.al. [9]	Drowsy	72.9%	76.1%	61.8%	0.519
	Actions	71.3%	72.8%	43.4%	0.334
	Male	68.0%	72.1%	44.3%	0.342
	Female	76.5%	76.8%	60.1%	0.455
	All	72.1%	74.4%	57.2%	0.387
Proposed Method	Drowsy	91.2%	91.5%	89.4%	0.763
	Actions	80.6%	80.8%	72.6%	0.586
	Male	86.2%	86.5%	79.5%	0.758
	Female	85.5%	85.9%	80.4%	0.635
	All	90.1%	90.5%	89.4%	0.755

Table 3.2: Classification results of open and closed states of the proposed method and Baccour et.al. [9]. Values of accuracy for open a closed eye states are presented. Separation for different typology video sequences is also analyzed.

3.7 Conclusions

In this chapter, a novel method to obtain the normalized eye aperture level of a person is introduced. The method is based on the calculation of different EWMA signals that try to capture, adaptively, the open and closed levels of each individual. The computation of the normalized eyelid aperture serves as the first step for the detection of blinks. The analysis of blinks is relevant for estimating the fatigue and drowsiness level of the driver.

This method has been shown to be robust against different types of data and drivers. The results obtained from the DMD dataset show that the proposed method can be applied to real scenarios with a high level of accuracy. The method was also compared with a recent method for blink detection and the results show that the proposed method outperforms the state-of-the-art method. Future work could include the analysis of the impact of selecting

different prior probabilities for the open and closed eye states, as well as the impact of the threshold value for the normalized eyelid aperture.

The proposed method does not make any assumption of the driver state. The estimation of the driver fatigue state requires a posterior analysis of the evolution of the driver eyelid aperture level. In order to proceed with this analysis, a personalized method is required. This means that the eyelid aperture levels should be stored in a historic matter in order to determine the situations where the driver is more likely to be awake or fatigued.

The availability of the DMD dataset has opened up new possibilities for research in the field of driver monitoring systems, particularly in the analysis of blink detection. The eyelid aperture estimation can feed other complex algorithms for driver monitoring so higher knowledge of the driver behavior could be extracted. In this sense, the results of the different components of a DMS (such as in Figure 6.1) could help improve the accuracy of the algorithms. For instance, the proposed method for blink detection can benefit from head pose estimation methods. The availability of head pose could help identify large movements and filter noisy eyelid aperture signals improving the overall accuracy.

Finally, the method for eyelid aperture estimation could serve as a tool for pre-annotating the remainder of the DMD dataset. This could reduce considerably the manual annotation efforts since the annotators will only need to correct the small divergences in the boundaries between the defined eye states. With these eye state annotations, more complex features can be extracted such as microsleep events, PERCLOS, time between blinks, blinking frequency and speed. Such type of features are valuable for the development of robust fatigue monitoring systems.

Chapter 4

Driver monitoring dataset generation

4.1 Introduction

Data has become the most important asset in the development of ADAS and AD functions. Deep learning (DL) and other artificial intelligence (AI) methods are being employed in the development and testing phases of many ADAS/AD functions present in modern vehicles [217]. Their non-deterministic nature obliges the industry to shift closed-loop validation approaches into long-term Big Data procedures, where sensor data fuels training and testing processes [205].

Large volumes of carefully crafted sensor data are needed for two main purposes: (i) to create datasets as input for DL training mechanisms and (ii) to create ground-truth metadata for validation of ADAS/AD functions in SiL (software-in-the-Loop) or HiL (hardware-in-the-Loop) setups.

The more complex the information the AI or function produces, the larger and richer the datasets needed. The trend in the automotive sector is to have open datasets that are multi-purpose and incorporate multiple sensors (e.g., LIDAR and cameras) datasets, which can be used both for training and validation of functions. The production of such datasets is no longer feasible for small companies or university groups. Compared with small-scale datasets, a large-scale dataset implies a tremendous technical and economical effort for designing, setting up, recording and post-processing data according to the quality standards required by the AI frameworks and the automotive industry.

Most of the largest public datasets in the ADAS/AD domain focus on exterior perception systems, ideal for building automated braking systems, lane departure warning methods, etc. However, there is a clear lack of datasets centered on monitoring the interior of the vehicle, i.e., the driver. Excluding SAE-L5 autonomous driving (see Figure 1.1), which is not yet a reality, the driver is still a key figure in the driving task. Though, as automated levels increase, there is a continuous shift in humans' role from driver to passenger, which can occur during driving situations, especially when automated functions get out of their defined ODD. In such situations (e.g., exiting from the highway), a transition between automated and manual driving modes must happen to respect safety regulations and principles. In that sense, DMS have gained focus in the industry.

Commercial vehicles have incorporated some sort of driver attention systems since 2006 [180], [207], [213], analyzing driving patterns and behavior. However, camera-based DMS has emerged as the most unobtrusive technology that enables gathering information about the physiological state of the driver: fatigue and distraction, without interfering with the driving task itself or implying a decrease in comfort.

While sharing similarities with exterior perception systems, DMS focus on analyzing the behavior of humans inside a small cabin as they perform driving tasks and engage in other activities, such as talking to passengers or operating car controls and other devices. DMS employs camera sensors to gather data that is used to evaluate the driver's ability to perform tasks and to monitor their state, which includes factors such as fatigue and distraction. As a consequence, DMS data is complex (fatigue and distraction-related), heterogeneous (facial features and actions), human-centered (individuals may show different behavior patterns) and mostly centered in camera sensors.

Datasets for DMS need to satisfy several requirements:

- Large amounts of data for training and validation of DL methods.
- Realistic data of relevant situations (e.g., safety related such as drowsy drivers, talking on the phone, etc.).
- Spatio-temporal labels, including visual features (e.g., face landmarks) but also actions as frame intervals with a semantic load.
- Captured data need to represent physiological states for fatigue, behavior and distraction, and thus several cameras might be needed to monitor the face, hands and body of the driver.

The preparation of a DMS dataset that meets these requirements imposes several important challenges, most of them from a technical perspective but also economic and organizational:

- Preparation of multiple environments (real car and simulator for simulation of non-safe driving behaviors or physiological states).
- Creation of complex annotation or metadata schemes to host heterogeneous labeling data.
- Organization of recording sessions and management of volunteers.
- Data preparation: recording, transferring and compressing large volumes of raw data.
- Data processing: synchronization, alignment, calibration, labeling, etc.
- Privacy and ethical aspects (GDPR compliance).
- Dissemination aspects, including website preparation, management of updates, GitHub repositories, user manuals, samples, etc.

The physiological signs of human inattention while driving are required to be taken into consideration to define a methodology to capture camera-based data and generate a complex annotated dataset that serves as material for training AI/DL methods for driving monitoring and as ground truth to validate DMS. This chapter aims to collect the characteristics of driver

inattention analyzed in Chapter 2 to propose a methodology to build a complex, multi-sensor dataset for driving monitoring.

To pursue this objective, an overview of the challenges faced to build a multi-camera and multi-modal large-scale dataset for DMS is presented. Then, a description of the approach to solving the technical and organizational issues that appear during the design and construction of DMS is described. The main contributions of this work are:

- Definition of a multi-sensor set-up architecture (multi-modal and multi-camera) for capturing large-scale video footage in the context of driving monitoring.
- Organizational approach to managing human volunteers and recording sessions: environments, scripting, recording, privacy and ethical aspects, etc.
- Orchestration of data preparation and data processing stages considering the DMS requirements: storage, containers, transmission, compression, synchronization, alignment, calibration and labeling.
- Taxonomy of driver monitoring labels for multi-level dataset annotation.

The proposed methodology can serve as a reference approach to create other multi-camera and multi-modal large datasets for interior or exterior monitoring applications in the automotive domain. This methodology was applied to build the recently published DMD (Driver Monitoring Dataset) in the context of the activities initiated during the execution of the European project VI-DAS. The main objective of the DMD is to gather a diverse collection of image data of the interior of the vehicle based on the most relevant features to observe inside the cabin. The target is to provide driver monitoring practitioners with a semantically rich dataset that allows the design and generation of DL-based algorithms for driving monitoring. The results of the work include the creation of recording and labeling tools, the implementation of new methods to synchronize and align streams and the construction and publication of the DMD, one of the largest open datasets designed to develop DMS.

4.2 Related work

4.2.1 Datasets in the automotive sector

The revolution of DL methodologies [123] has made indispensable the availability of large volumes of high-quality data to train network models. To extract all the potential of DL techniques for computer vision tasks, the available data need to have a diverse collection of annotations aimed at tackling different tasks. By using these annotated datasets and a variety of state-of-the-art DL methods [93], [120], [131], [171], [174], a DL neural network can then be used and fine-tuned for the particular application in the driving context. Many generic datasets have been proposed: ImageNet [37], MS Coco [130] and PASCAL [54], [55]. All of these datasets consist mainly of a collection of images containing instances of different objects with their corresponding label to address the object detection task in scene understanding. The application of these datasets is limited to generic object detection and semantic segmentation tasks, where the object can appear in a wide range of scenarios.

Although these datasets can be the starting point for training object detection models, the adaptation to the automotive domain results in limited and not robust DL models. Hence, additional effort is required to adapt and fine-tune the models by using domain-specific data. In addition, such types of datasets only provide information from one type of sensor: an image camera. Having only one modality of input data may limit the robustness of autonomous systems in complex scenarios.

To adapt the DL models and provide multi-modal data in the context of automated driving, there has been an intense development and effort to design, build and make available large-scale datasets to sense the exterior of the vehicle. Both industry and academia have published diverse collections of data that combine RGB images, IR images, LiDAR point clouds and high-precision GPU/IMU inertial navigation data. Some of the first to appear and most relevant datasets for exterior perception include KITTI [72], [73], WAYMO [202], Cityscapes [31], nuScenes [23] and Lyft Level5 [92].

The experience gathered to produce these massive datasets for exterior sensing has resulted in some common practices in terms of the type of data and sensors used for recording the sessions. Based on some shared processing tasks such as object detection (pedestrians, vehicles, obstacles, etc.), segmentation (free space and instances), 3D sensing or localization and positioning; these datasets have defined the path to producing a large-scale dataset for analyzing the exterior of the car. However, the level of maturity in the generation of databases focused on driver monitoring has not been reached yet, even though there have been some efforts to produce datasets aimed to cover specific tasks of driving monitoring.

4.2.2 Driver monitoring systems and data requirements

The driver’s behavior involves different body and face activities. Ensuring normal driver behavior is critical to avoid any human error that could eventually produce traffic accidents with the consequent harm to drivers and passengers. Therefore, to prevent vehicle accidents, systems are built to monitor the drivers and assess their arousal and distraction levels. Among driver behavior conditions, fatigue and distraction are the two most relevant subjects of study when designing DMS [173]. According to the type of input data provided by the measurement sensor, systems designed to analyze and assess driver behavior can be broadly divided into two categories: visual-based features and non-visual-based features.

Techniques based on non-visual features obtain the input data from sensors, which can either be intrusively attached to the driver’s body [29] or require the extraction of car parameters [193]. Within this group, techniques are also divided into two categories: *driver physiological features analysis* and *vehicle parameter analysis*.

Physiological features are often a good indicator of early signs of fatigue or distraction. Sensors to measure the physiological features of the driver extract signals from the person’s heart, brain, eyes or skin. Changes in physiological signals such as electroencephalography (EEG) [103], electrocardiogram (ECG) [183], electrooculography (EOG) [43] and surface electromyography (sEMG) [193] can be an accurate method to detect driver state. Physiological features are a direct measure of fatigue, however, their application in real driver fatigue and distraction detection systems is limited due to intrusive characteristics of the sensors used and critical

issues to eliminate noise and artifacts, which are inevitable in real-world driving conditions.

Moreover, it is well known that fatigue and distraction reduce the driver’s ability to perform. Therefore, other researchers have studied parameters obtained from vehicle sensors to analyze the driver’s state. The deviation in features such as lane crossing [45] and steering wheel angle [190] are indicators of deteriorating driving ability. Other strong indicators of abnormal driver behavior also include unusual activities such as pressure changes on brake and accelerator [128], load distribution on the driver’s seat [143] and vehicle speed [108]. In general, there is limited research available exploiting this technique. Vehicle signals for driver inattention assessment are easily affected by driving habit, driving skill, vehicle speed, vehicle type and road conditions. Therefore, a robust DMS solution based only on vehicular signals is not yet feasible.

Techniques using visual features usually rely on the use of camera sensors pointing to the driver to obtain input data. Image processing techniques enable monitoring of these activities through a camera capturing the most relevant parts of the driver. Image-based DMS should address two primary tasks: analyzing the sequence of images for extracting important indicators and evaluating the temporal evolution of the driver state using such indicators.

Recent DMS have focused on evaluating different driver states using visual-based camera sensors. RGB or visible spectrum cameras, depth cameras or infrared cameras are the three main types of sensors used in vision-based DMS.

Among the different methods and systems found in the literature, RGB cameras are the most applied in research DMS [82]. The reason is that visible spectrum cameras are affordable and their integration in modern systems is easy to implement. However, due to illumination limitations in real driving conditions, NIR external illumination and IR camera sensors are preferred since the use of DMS can be extended to low-light conditions [38], [144].

Other works have integrated depth sensors to provide additional data to the DMS algorithms and better predict driver state [17]. Besides visual data, some efforts have been conducted to incorporate other modalities such as audio using microphones attached to the person to estimate stress and emotions in drivers [71]. The outcomes of many of the currently available DMS methods are pushing forward to fuse several modalities and data streams to develop a robust method to infer driver state [155], [179].

4.2.3 Datasets for application in driver monitoring systems

The majority of efforts in generating visual datasets have been focused on sensing and capturing a vehicle’s exterior. While numerous initiatives offer vision-based datasets for DMS, the absence of comprehensive, consolidated datasets has inspired this thesis to contribute to the development of the DMD; in Figure 4.1, there are example images of some activities included in this dataset. Driver monitoring requires the interpretation of the driver’s features regarding the attention and arousal state, the direction of gaze [214], head pose [63], the position of the hands [226], blink dynamics [160], facial expressions [101], body posture [134] and drowsiness state [75]. Presently, the available datasets address these DMS aspects separately and fail to offer a thorough overview of the various driver attention states.

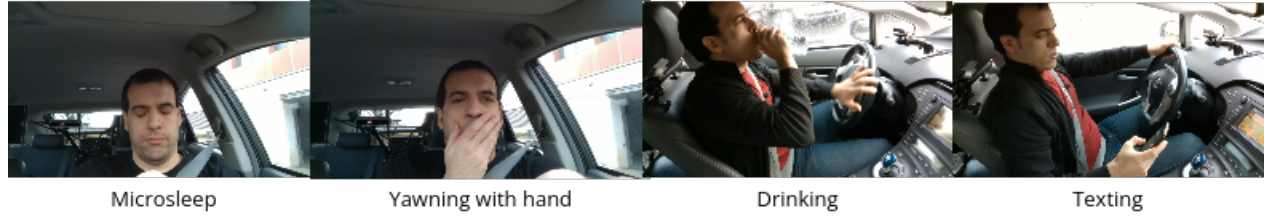


Figure 4.1: Examples of activities performed in the DMD

In Table 4.1, a list and description of the most relevant open datasets found at the moment are presented. The analyzed datasets present data of the driver focused on specific parts of the body such as the face [39], hands [154], [226], or upper body [18]. They primarily employ RGB cameras and depth sensors. Metadata labels include geometrical features like bounding boxes, landmarks, masks, and temporal actions. Recording scenarios differ between real vehicles and in-lab simulators. Most datasets utilize only one camera for recordings, except for a recent dataset [135] that provides visual data from different viewpoints using multiple cameras. However, its focus is on analyzing driver actions during autonomous car functions.

Dataset	Year	Drivers ^a	Views ^b	Size ^c	GT ^d	Streams	Scenarios	Usage
CVRR-Hands [154]	2013	8 (1/7)	1	7 k	Hands, Actions	RGB Depth	Car	Normal driving, Distraction
DrivFace [39]	2016	4 (2/2)	1	0.6 k	Face/Head	RGB	Car	Normal driving, Head pose
DROZY [137]	2016	14 (11/3)	1	7 h	Face/Head Physiological	IR	Laboratory	Drowsiness
NTHU-DDD [218]	2017	36 (18/18)	1	210 k	Actions	RGB IR	Simulator	Normal driving, Drowsiness
Pandora [18]	2017	22 (10/12)	1	250 k	Face/Head, Body	RGB Depth	Simulator	Head/Body pose
DriveAHead [187]	2017	20 (4/16)	1	10.5 h	Face/Head, Objects	Depth IR	Car	Normal driving, Head/Body pose
UTA-RLDD [75]	2019	60 (9/51)	1	30 h	Subjective KSS labels	RGB	Laboratory	Drowsiness
DD-Pose [179]	2019	24 (6/21)	2	6 h	Face/Head, Objects	RGB ^e Depth ^f IR ^f	Car	Normal driving, Head/Body pose
AUC-DD [47]	2019	44 (15/29)	1	144 k	Actions	RGB	Car	Normal driving, Distraction
Drive&Act [135]	2019	15 (4/11)	6	12 h	Hands/Body, Actions, Objects	RGB ^e Depth ^e IR	Car	Autonomous driving, Distraction
DMD	2021	37 (10/27)	3	41 h	Face/Head, Eyes/Gaze, Hands/Body, Actions, Objects	RGB Depth IR	Car, Simulator	Normal driving, Distraction, Drowsiness

^a Number of drivers (female/male)

^c h: hours of video, k: image number

^e Only for side view

^b Simultaneous views of scene

^d Ground-truth data

^f Only for face view

Table 4.1: Comparison of public vision-based driver monitoring datasets

Based on the limitations of the currently available in-cabin driving monitoring datasets, the limited size of these datasets and the increasingly new challenges and requirements of DMS, the DMD project has built a preparation methodology to collect different sources of drivers' behavior data and deploy the required metadata annotations to be used by different driving monitoring sensing systems.

4.3 Dataset creation strategy

The preparation of a large-scale dataset for DMS involves different organizational and implementation stages, which require careful planning and execution. During the preparation of the DMD, three stages were defined. The first stage involved the preparation of the resources needed and planning the execution of the recordings. During the second stage, the actual recordings with the volunteers were performed. The third stage consisted of post-processing the recorded material, data extraction in compressed and manageable formats, solving syncing and alignment issues between video streams and annotation of the defined classes and labels.

In Figure 4.2 the complete process for the creation of the DMD is depicted, showing the three main stages for creating the dataset and the sub-actions for each stage. The relation between the tasks and the timeline required to complete all the required actions is also represented. Some of these steps can be performed in parallel to reduce the time spent in the creation of the dataset.

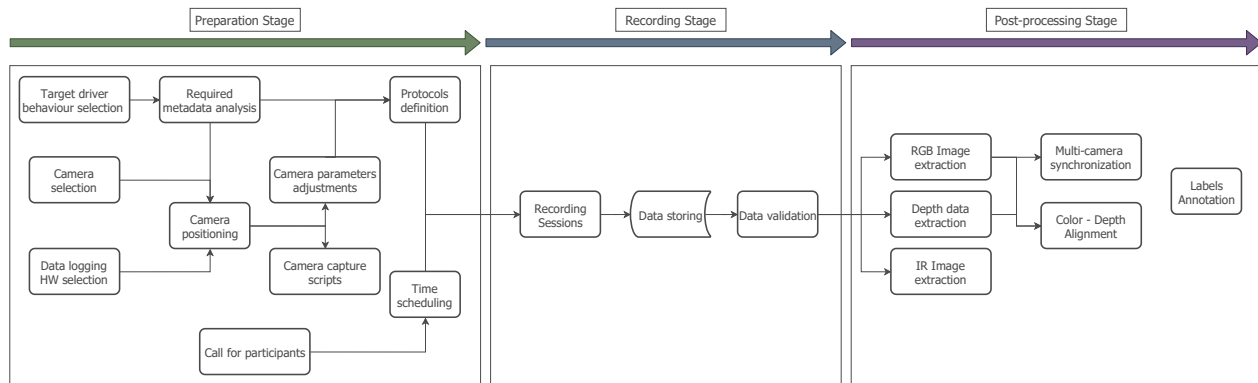


Figure 4.2: General overview of the DMD creation process

4.3.1 Metadata taxonomy definition

Planning the creation of a computer vision dataset requires having in mind the target type of annotations to be exported. A rich and varied set of labels makes the dataset more usable for different scenarios and problems, especially when the task to be solved can be decomposed as a set of small jobs. In particular, the implementation of DMS usually involves the combination of task-specific algorithms with the final objective of exporting a set of features to describe the actual state of the driver [163].

The required target metadata for the DMD was defined based on the analysis of the most relevant characteristics found in the state-of-the-art methods for DMS. Specifically, the DMD

wants to cover three main pillars of DMS: drivers' distraction detection, fatigue estimation and gaze estimation. The recordings and the annotation planning are oriented to support these methods; however, due to the DMD's richness and variety of data, other uses of the dataset may appear in future studies.

Previous works have analyzed driver distraction from still images, with techniques such as body pose estimation, which requires body landmarks and body position annotations [82]. Besides geometrical metadata, the goal was to perform the analysis including the temporal dimension; this involves recording moving sequences and providing temporal annotations. The head/face-centered analysis, for driver fatigue estimation for instance, typically focuses on quantitative measures such as PERCLOS, eyes aperture, etc., which are calculated based on face landmarks. This approach requires annotating these spatial landmarks to facilitate a temporal analysis.

To better distribute the dataset annotation tasks, three metadata groups are defined: scenario, geometrical and temporal data. These groups allowed us to design the recording protocols and annotation guidelines, reducing ambiguity and boosting the value of the recorded material. The proposed taxonomy of features in the DMD is presented in Figure 4.3. This taxonomy is flexible and allows the inclusion of other labels within each category. Moreover, it is possible to increase the level of detail of the presented label tree.

Scenario Metadata

The process of creating a vision-based dataset requires checking not only the availability of a diverse set of visual labels but also other information that describes the capturing conditions of the data. For the creation of the DMD, this context information was considered to complement the other visual labels, which are valuable for the development of robust DMS. This group includes the information related to the driver (age, gender, use of glasses, driving experience, etc), capture context (weather, timestamp-ISO8601 and capturing set-up), recording metadata (total number of frames, duration of session) and camera parameters (model, calibration and synchronization parameters). Moreover, scenario information is required to further analyze the correlation of some driver behavior performance based on personal characteristics.

Geometrical Features

For any computer vision dataset, geometrical labels are the core part of the dataset. Therefore, the DMD was designed to produce a diverse set of geometrical characteristics of the driver and the car interiors. The geometrical annotations considered within the creation of the DMD depict the visible characteristics of the scene related to the driver's behavior while driving.

The regions of interest where DMS typically focuses are the face and body of the driver, mainly because it is possible to infer the actual state of the driver in terms of fatigue and distraction. For instance, fatigue is generally observed from eye activity, such as analyzing the blinking cadence and frequency and computing some statistics that highly correlate with drowsiness and fatigue. A multi-sensor camera set-up for capturing the DMD and collecting information on the driver from different perspectives was used to collect these types of features.

Geometrical features are usually represented as a set of values that enclose some spatial region.

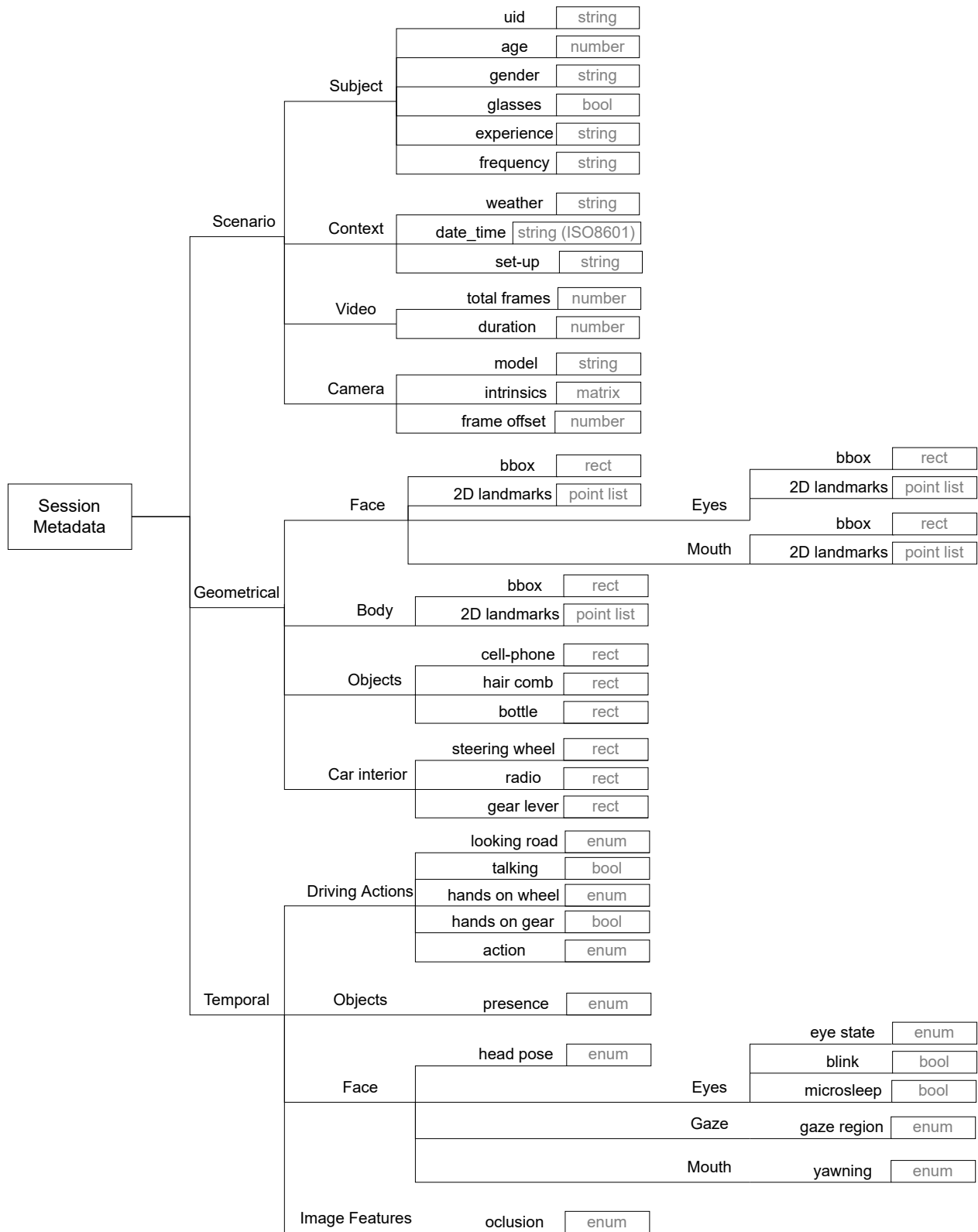


Figure 4.3: Metadata label taxonomy for the DMD

These features can be: (i) points such as image landmarks; (ii) bounding boxes to delimit a rectangular area such as the face or body of the driver; (iii) pixel masks to delimit a region of the images taking into account the object's contour.

Temporal Features

A temporal label describes a sequence or action through time. It assigns a semantic value, e.g., a name or type of action, to a time interval of a recorded video, indicating when the action begins and ends. A temporal annotation job starts with the definition of the semantic values or labels and the level of detail to be handled in the final annotations. For example, a general annotation should have one label that describes the entire sequence, such as 'texting'; at a higher level of detail, this label can be derived into two, 'texting left' and 'texting right', differentiating the hand with which the person is texting.

Labels can be defined and arranged hierarchically (based on the level of detail) and logically connected. For instance, when an individual is engaged in the 'texting left' action, which entails sending a text message with their left hand, they cannot simultaneously perform the 'texting right' action or send a message with their right hand. This analysis is crucial for understanding the context of the annotation task and establishing these logical relationships.

There are not only mutually exclusive labels that can not coexist in the same time interval. A sequence can be described by several labels simultaneously. To make this possible, they can be organized by levels. A level of annotation is composed of one label or a group of mutually exclusive labels.

Suppose the case of a person sending a text message and discriminating the hand with which he/she is holding the mobile phone; furthermore, you want to describe where the driver is directing his/her gaze while performing this action, either at the mobile phone or somewhere else. For this case, there should be two levels of annotation, the first level identifies how the individual is texting, while the second level determines whether or not the person is looking at their mobile phone. Those two levels of annotations have mutually exclusive labels since the person is either using the right or left hand to text and because his/her gaze is either directed at the phone or somewhere else, none can occur at the same time.

Additionally, when annotating temporal features, camera occlusion plays a crucial role in video sequences. It enhances the accuracy of the driver action detection algorithm by excluding these obstructed images from the machine learning model training. Regarding the creation of DMD, it was decided to annotate occlusions in any of the three cameras for all recording sequences. An occlusion, in this context, is defined as an event where more than 50%-60% of the camera's view is obscured by the driver's body or any other object, rendering the scene unrecognizable.

Using this logical and hierarchical organization by levels, the annotations of the DMD were created. Based on this labelling taxonomy a comprehensive annotation manual can be created and provided to the annotators of the dataset.

The following paragraphs provide a detailed explanation of the annotation criteria used in the DMD annotation process. First, in Table 4.2, an example of 7 levels of annotation for

distraction-related recordings is shown. All levels are mutually exclusive or encompass only one label.

Level	Labels	Description
Gaze on Road	- Looking road - Not looking road	This category is used to know whether the driver is putting all his/her visual attention on driving
Talking	- Talking	This category is used to know whether the driver talking
Hands Using Wheel	- Both - Only right - Only left - None	This category is used to know the driver's involvement of his/her hands in the driving task
Hand on Gear	- Hand on gear	This category is used to know whether the driver has his/her hand on the gear
Objects in Scene	- Cellphone - Hair comb - Bottle	This category is used to know if there is a visible object in the scene (captured from any of the cameras)
Driver Actions	- Safe drive - Texting right - Texting left - Phone call right - Phone call left - Radio - Drinking - Reach side - Hair and Makeup - Talking to passenger - Reach backseat - Change gear - Standstill/Waiting - Unclassified	This category is used to know which of the listed activities the driver is doing
Camera Occlusion	- Face camera - Body camera - Hands camera	This category denotes whether there is an obstruction in a camera's view

Table 4.2: Subset of annotation levels for distraction-related material in the DMD

It is crucial to be as descriptive as possible to avoid misinterpretations by the annotators. To begin with, the label text or name should be clear and specific: it must represent the sequence correctly and be different from other labels. It is also recommended to write a complete definition of each label, indicating when the sequence it represents should begin and end.

For the **drowsiness-related** material, the participants were asked to do activities like yawning with and without covering their mouths with their hands, have micro-sleeps and, in general, perform sleepy driving. In Table 4.3 an example of the different annotation levels selected for the drowsiness sequence in the DMD is presented. By annotating this material, it is possible to identify the different stages of drowsiness and the actions that the driver performs while being drowsy. This information is useful for the development and validation of drowsiness detection systems.

Level	Labels	Description
Eye State (Right)	- Open - Close - Opening - Closing - Unclassified	This category aims to indicate the state of the driver's right eye
Eye State (Left)	- Open - Close - Opening - Closing - Unclassified	This category aims to indicate the state of the driver's left eye
Blink	- Eyes blinking	This category identifies occurrences of eye blinks
Yawning	- With hand - Without hand	This category identifies occurrences when the driver yawns and distinguishes whether they cover their mouth with their hands or not
Camera Occlusion	- Face Camera - Body Camera - Hands Camera	This category denotes whether there is an obstruction in a camera's view

Table 4.3: Subset of annotation levels for drowsiness-related material in the DMD

Labels within this category correspond to actions that occur in an instant or a sequence in time. Then to annotate these features, the annotators must indicate the time interval in which the action occurs. For example, if the driver yawns, the annotator must indicate the time interval in which the driver yawns. If the driver yawns and then covers his mouth with his hand, the annotator must indicate the time interval in which the driver yawns and then the time interval in which the driver covers his mouth with his hand. Hence it is important to observe and analyze the full image sequence to identify the different actions that the driver performs.

In addition, it was noted that driver behavior encompasses distinct temporal actions, particularly those involving gaze activity. Consequently, specific annotation criteria for sequences related to gaze were incorporated into the design of the DMD. Table 4.4 outlines the label categories established within the DMD dataset design for **gaze-related** sequences.

Level	Labels	Description
Gaze Zone	- Open - Left mirror - Left front - Center mirror - Front right - Right mirror - Right - Infotainment - Steering wheel - Not valid	This category aims to identify the gaze zone at which the driver is looking
Blinking	- Eyes blinking	This category identifies occurrences of eye blinks
Camera Occlusion	- Face Camera - Body Camera - Hands Camera	This category denotes whether there is an obstruction in a camera's view

Table 4.4: Subset of annotation levels for gaze-related material in the DMD

4.3.2 Scripted protocols

Various methods exist for conducting driver monitoring tasks. The created dataset specifically emphasizes three aspects: distraction, drowsiness, gaze direction and hand behavior. Consequently, the development of the DMD involved capturing specialized footage of drivers in distracting circumstances, exhibiting signs of drowsiness, and directing their gaze toward distinct areas within the vehicle.

Driver behavior is very complex and is conditioned to certain circumstances. To intentionally capture those behaviors, it is ideal to recreate the trigger situations to have data as realistic as possible. However, to collect realistic situations of the interesting driver behavior, it is required to collect many hours of video and filter the relevant parts of the scene. In addition, sometimes some situations are very difficult to collect from naturalistic recordings, such as sleepy drivers, since these edge situations will put the driver's safety at risk. Therefore, for situations where it is hard to collect data, the behavior should be acted in the most natural way possible.

To circumvent these limitations and optimize recording sessions, a series of protocols have been proposed to guide the process of capturing the desired driver behavior. While spontaneity in driving is important, the activities, timings, conditions, and driver states need to be strategically planned to fulfill the intended content capture objectives.

These recording protocols were scheduled across various days and times to ensure diverse lighting and weather conditions. Furthermore, volunteer drivers were recorded on multiple occasions, allowing for different attire during the sessions. A total of 150 recording sessions

were planned and executed during this phase. Of these, 87 took place in the morning, while 63 were conducted in the afternoon time slot.

Each protocol comprises a comprehensive set of instructions on how to conduct each experiment (recording session), ensuring uniformity among experiments and the completeness and quality of the content. As a result, distinct protocols exist for each aspect of driver analysis and recording location (car or simulator). A protocol's general structure consists of the following stages:

- **Participant welcoming:** Check which participant is programmed, call him/her and welcome him/her. A description of the context of the experiment must be explained to the participants, along with the legal privacy terms and a brief description of their function: perform the indicated activities most naturally while driving (real or acting) and if required with a certain state (drowsy or attentive).
- **Rehearsal and technology check:** A quick practice of the activities to perform is necessary, both to verify that the participant has understood the activities to perform and that the recording equipment (cameras, microphones, etc.) is working correctly.
- **Recording activities to perform:** The actual recording is carried out following a predefined list of actions to be performed or acted by the driver, specifying the time (seconds) that each activity should last, approximately. The individual responsible for conducting the experiment has authority over the timing and instructs the participant on when to perform a specific action.
- **Final check:** It is proposed to conduct a quick check of the recorded material to verify that the recording session went well and prepare everything for the next participant.

Distraction Protocol

The objective of this scripted protocol is to capture a person performing several activities related to the concept of distraction while driving. The list of proposed activities is presented in Table 4.5. When the driver is not distracted and is focused on the road or when the subject is not performing any of the activities related to distraction, it is considered 'safe driving'.

Texting, phone call, hair and makeup and reaching behind are activities that are considered dangerous and illegal to perform while driving. This limitation motivates the use of a driving simulator that allows recording those activities while driving without the mentioned risks. For keeping the environment of an actual car, these dangerous activities were also captured with the car stopped. However, some activities such as operating the radio, talking to the passenger and reaching behind are not possible to record in the simulator because of its limitations of space and lack of adjacent real car elements.

The distraction protocol was then adapted into two parts: one exclusively for the activities included in the simulator recordings and another for those performed in the car. This last one was also divided into two sections: one for the activities executed with the car moving and one with the car stopped.

The estimated time to carry out one complete recording session of the distraction protocol

Car Stopped	Car Driving	Simulator Driving
Safe driving	Safe driving	Safe driving
Reach object backseat	Operating the radio	Brush the hair
Reach object side	Drinking	Phone call – right hand
Brush the hair	Talk to passenger	Phone call – left hand
Phone call – right hand		Texting – right hand
Phone call – left hand		Texting – left hand
Texting – right hand		Drinking
Texting – left hand		

Table 4.5: Actions performed in the distraction protocol

in the car is 28 min per person, considering that first the protocol with the car moving is performed, followed by the one with the car stopped. For the simulator, a separate session is scheduled and it takes 20 min to complete its distraction protocol.

The footage’s approximate mean length in the simulator is 9 min per recording, 4 minutes for the stopped-car sessions and 8 min with the car moving.

Drowsiness Protocol

In contrast with the distraction activities, drowsiness sessions, as shown in Table 4.6, are more complicated to achieve naturally, mainly because a state of drowsiness cannot be induced immediately; it has to be acted. Subjects were asked to show signs of drowsiness such as yawning, nodding and having micro-sleeps; although they are left free to perform the activities as they prefer to not affect the naturalistic behavior of the action, it highly depends on the imitation abilities of the driver for these to seem natural.

Car Stopped and Simulator Driving
Drive normally
Sleepy driving
Yawn without hand
Yawn with hand
Micro-sleep

Table 4.6: Actions performed in the drowsiness protocol

At the beginning, the participant is asked to drive normally, this can be manifested by an alert disposition for driving, checking the rear mirrors, firmly holding the steering wheel, etc. Then, the participant imitates sleepy driving, which is characterized by a tired position of the head and body, a certain lack of attention to the surroundings, short eyelid aperture and

occasional nodding. It was indicated to the driver when to perform the signs of drowsiness. He/she maintained this sleepy driving behavior throughout the recording until its completion. It took approximately 10 min for every execution of the drowsiness protocol, and the length of the resulting recordings is about 3 min per session and setup.

Gaze and Hands Protocol

In this protocol, 9 regions of gaze direction were established. These are zones inside the car where the driver frequently looks while driving and which are wide enough to be considered a region. These same regions were also calculated in the simulator. The positioning and rotation of the simulator's multiple screens are adjusted to provide a realistic car cabin perspective. This produces the feeling of being in a vehicle and allows maintaining the same regions to detect gaze direction. The participant was asked to stare at each of the regions for 5 seconds when indicated.

In addition, an analysis of the driver's hand position was also considered. The objective of defining a set of actions is to gather a diverse collection of hand positions while using the steering wheel. During the sessions performed using this protocol, the driver adopts one position for a few seconds and then changes to the next when indicated. To add variation, the hands were captured in two situations: standing still and moving as if the participant were driving. A representation of the gaze regions and the list of hand actions is depicted in Table 4.7. It took about 7 min for every session, and the length of the resulting recordings is around 3 min.

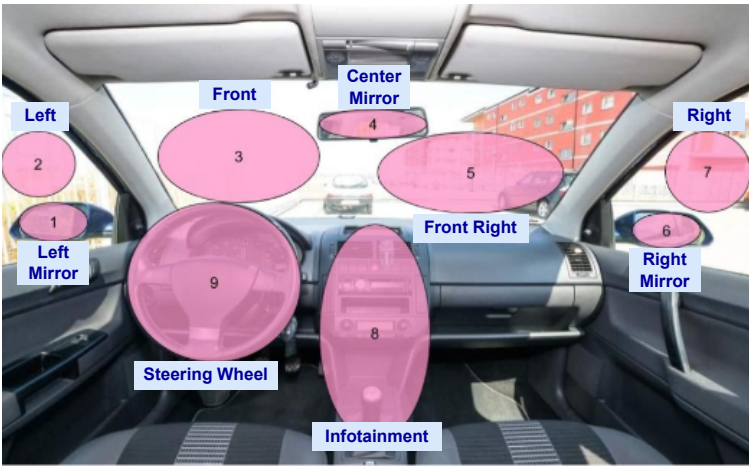
Car-Simulator-Driving	
Gaze Zones	Hand Actions
	Both hands on (not moving)
	Right hand on (not moving)
	Left hand on (not moving)
	Both hands off (not moving)
	Both hands on (moving)
	Right hand on (moving)
	Left hand on (moving)

Table 4.7: Actions performed in the gaze-hands protocol

4.3.3 Participants selection

The success of a feature-rich visual dataset highly depends on the variety of characters that appear in the recordings, especially when human participants are the subjects of the analysis. To have a diverse collection of driving monitoring videos, the participants should meet different requirements: diverse ethnicities, ages, driving experience and visual characteristics. As shown in Figure 4.4, for the recording of the first stage of the DMD, 37 driver volunteers participated. This group offers variability in gender: 73% men and 27% women. The age distribution of the participants was kept homogeneous in the range of 21 to 50 years of age. In addition, 10 participants regularly use glasses when driving. These participants were selected to assure novice and expert drivers were included in the recordings. Each participant signed a GDPR informed consent, which allows the dataset to be publicly available for research purposes.

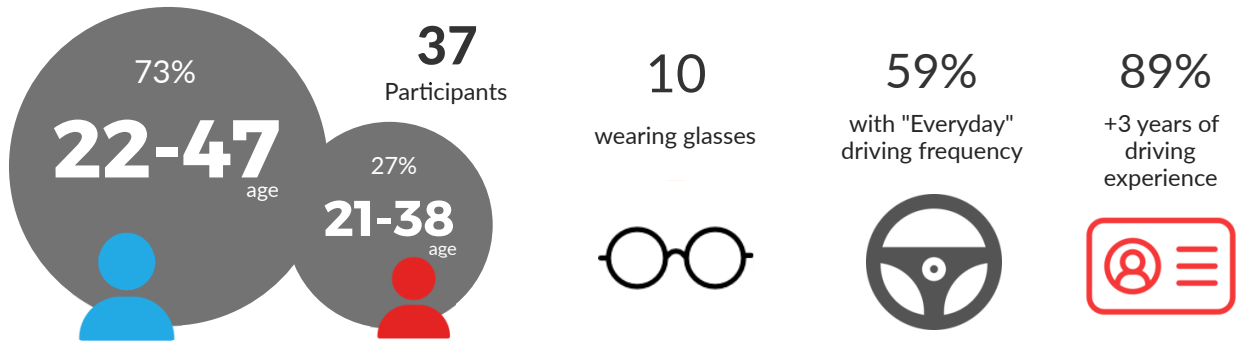


Figure 4.4: DMD Participants information

No dataset is perfect (a look at any major dataset may throw a non-50-50% balance in many dimensions, e.g., Waymo, Apollo, Mapillary-Vistas, and any of the driver monitoring datasets referenced in Table 4.1). It is very difficult to reach equilibrium in all aspects, especially when datasets are as complex as the DMD. Context, illumination, variety of actions and subjects may easily scale up the combinations exponentially. Presently, manufacturers and AI practitioners are encountering this issue as a common industrial challenge. The DMD has tried to achieve the best balance in different aspects. For the case of the gender balance, the main limitation was the possibility to find volunteers to participate in the recording campaigns. However, the balance obtained, despite not being perfect, reaches sufficient representativeness to research and build applications with the DMD.

Participants were grouped into 6 groups of 5 individuals and 1 group of 7. Each participant was asked what time of the day they could participate in the recordings: in the morning, in the afternoon, or both. Knowing the availability of the participants, they were organized by groups depending on the recording sessions they could attend. Then, a double participation schedule was assigned to 3 groups of volunteers whose availability was “all day”. All the protocols were recorded twice for these participants. Meaning that the same person did the same activity with morning and afternoon lighting, not on the same day, contributing to variations in clothing. In the DMD, the material of 15 people meets this condition.

Taking into account all these considerations and the estimated times of each protocol, the daily planning of the recordings program was sent to each group with specific dates and times.

This is how the participants were organized, making the logistics of the recordings easier in general.

4.4 Recording of visual data for Driver Monitoring

4.4.1 Multi-sensor setup architecture

The selection of the sensor architecture is one of the main pillars for a successful generation of a CV dataset. These sensors should provide enough data quality to collect the desired information in the form of images. For the case of the DMD, a set of requirements in terms of sensor capabilities were identified that fulfill the overall objectives of creating a dataset for DMS applications. The following requirements were identified:

- **Camera-based sensors:** Data extracted from the sensors primarily relies on a visible feature inside the vehicle and driver.
- **Multi-modal camera sensors:** For a diverse and richer collection of data, the different modalities include RGB images, IR images and depth data.
- **Multi-camera arrangement setup:** To capture several characteristics of the driver, multiple views of the inside of the vehicle are required.
- **Synchronized images:** The data obtained from various cameras and multi-modal streams are timestamped, synchronized, and appropriately aligned.

Depth sensing has become one additional source of data to improve the accuracy of computer vision tasks. Three depth-sensing modalities are the most relevant: stereo sensors (both active and passive), structured light, and time of flight (ToF). Attending to the defined requirements, three main characteristics of sensor performance are analyzed to select the camera used for recording the DMD: range, resolution and reliability.

Range is perhaps the most important parameter. In driving monitoring, the cameras shall have a capture range where body parts such as faces, hands and limbs are accurately sensed. Since the person will be inside the car cabin, this limits the range to less than 2 m. Depth sensors have ranges that start at around 0.5–1 m from the sensor and extend to 3–6 m at best. Here, stereo sensors can provide a good trade-off between range and camera baseline.

Regarding **resolution**, it depends on the range as well as the nature of the optics used by the sensor. Structured light sensors and active stereo sensors' resolution are limited by the density of the pattern that they project and the resolution of the IR camera. In the DMD, the resolution of the camera sensors is selected to be high enough to capture the smaller parts of the driver's body (i.e., the eyes and mouth).

In third place, sensors' **reliability** depends on the inherent limitations of each sensor's modality. Stereo sensors, for instance, need sufficient ambient light in the visible spectrum to work properly. Conversely, infrared-based sensors such as structured light and ToF suffer when exposed to too much ambient light, making them not appropriate for exterior applications. Active stereo sensors with infrared pattern projectors offer the *best of both worlds*, improving sensor reliability across diverse environments and lighting conditions.

To build the DMD, the Intel® RealSense™ cameras were chosen as they provide a good trade-off between hardware cost, power consumption and form factor. Models D415 and D435 matched the active stereo sensors requirements for the three modalities (RGB, IR and Depth).

We placed three cameras to record the driver’s face, body and hands. The location of the cameras is shown in Figure 4.5. The selection of the camera model was conducted based on the field of view (FOV) of the available cameras, without compromising the minimum range for depth sensing.

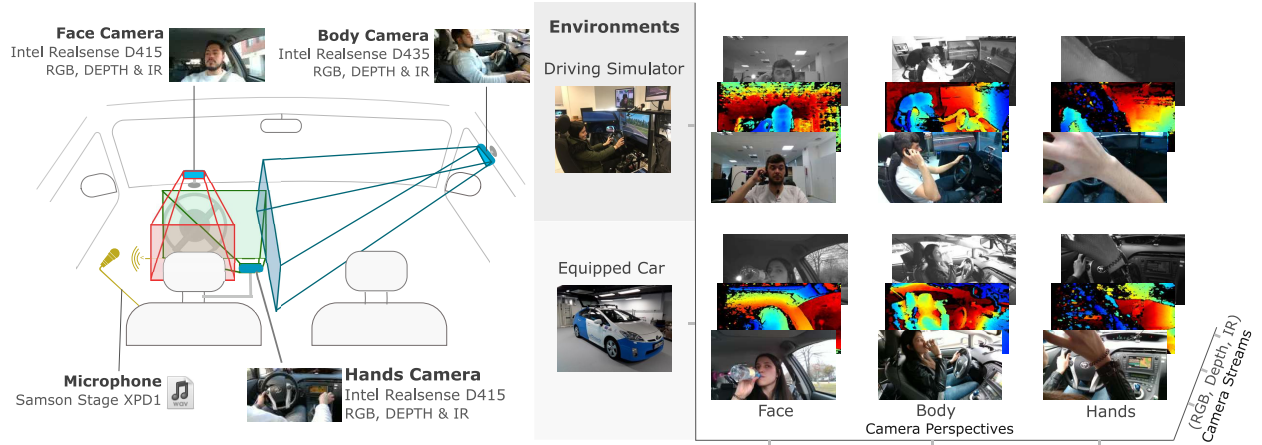


Figure 4.5: DMD camera setup and recording environments

Other in-cabin multi-sensor monitoring architectures published have either focused on single-camera set-ups to monitor the driver or the passengers [208], while other works have increased the number of cameras for inside sensing to include body sensing cameras [62] or depth information from one view [135]. The **DMD** is the first dataset to use three synchronized cameras, each equipped with three modes (RGB, IR, and Depth), to monitor distinct aspects of the driver such as their body, face, and hands.

In Table 4.8, the most relevant characteristics of cameras D415 and D435 are shown. Model D435 was used to capture the body of the driver due to its larger FOV and D415 was used for face and hands sensing due to its lower price tag.

To extend the completeness of the DMD, a microphone *Samson Stage XPD1* was attached to the driver to record usual sounds during the performed activities. The audio information of the microphone was recorded for future utilization, as a complementary data stream that can help to distinguish actions that may be difficult to label or detect using only image information. For instance, “talking” or “talking to the passenger” are actions labeled in the DMD whose start and end time instants could not be easily determined by only looking at the images. The audio information has not been labeled for the research conducted in this dissertation due to resource constraints and to prioritize the analysis of visual aspects of driver behavior. Extending vision-based datasets with other modalities is beneficial for the multi-modal study of the detection tasks, allowing practitioners from different areas of expertise, besides computer vision, to explore the relationship between visual features and other data modalities.

	D415	D435
Use Environment	Indoor/Outdoor	Indoor/Outdoor
Depth FOV ($H \times V$)	$65^\circ \times 40^\circ$	$87^\circ \times 58^\circ$
Depth Resolution	Up to 1280×720	Up to 1280×720
Depth Frame Rate	Up to 90 FPS	Up to 90 FPS
RGB FOV ($H \times V$)	$69^\circ \times 42^\circ$	$69^\circ \times 42^\circ$
RGB Resolution	Up to 1920×1080	Up to 1920×1080
RGB Frame Rate	30 FPS	30 FPS
Min. Depth Distance at Max Resolution	~ 45 cm	~ 28 cm
Ideal Range	0.5 m to 3 m	0.3 m to 3 m
Dimensions	99 mm $\times$ 23 mm $\times$ 20 mm	90 mm $\times$ 25 mm $\times$ 25 mm

Table 4.8: Specifications of Intel[®] RealSense[™] cameras used to generate the DMD

In addition, two twin recording scenarios were considered: a real car and a driving simulator, as shown in Figure 4.5. A driver monitoring dataset should be collected in real driving conditions with a real car. Nevertheless, certain activities pose a risk to the driver’s safety and cannot be recorded while driving on the road, such as using a cell phone or driving while experiencing significant drowsiness. For those cases, the methodology presented in this work defines alternatives that mimic real conditions. Those cases were captured using a stopped car and the driving simulator. The purpose of the driving simulator is to immerse the driver in a close-to-real driving experience. The cameras were positioned at an equivalent distance from the driver’s seat in both the actual car and the driving simulator.

4.4.2 Recording sessions

Once the preparation stage is completed, the recording sessions can take place. One common problem faced in real cars is the reduced bandwidth support for data transmission due to the limitations in the onboard PC hardware. To overcome this common case in automotive applications, a dedicated recording computer was used which is capable of supporting intense work and movements in the car. This computer was equipped with 3 dedicated USB 3.0 ports to support the connection interfaces of Intel RealSense cameras. The PC Hardware specifications for the recordings are:

- CPU: Intel i9 7940X series X 4.30 GHz
- RAM: 64 GB
- HD: 2 $\times$ SSD 1 TB M2
- GPU: 2 $\times$ NVIDIA GeForce[™] RTX 2080Ti 11 GB GDDR6 PCIe 3.0

The selected cameras allow the user to configure different parameters to improve the capture

quality of the data. For any dataset acquisition task, the dataset designers should pay special attention to the parameters which affect the image quality. When building datasets where multiple cameras are present, different factors affect the performance and the allowed configuration. Intel RealSense cameras can operate in multiple modes for frame resolution and FPS, but these parameters are limited by the bandwidth supported by the connection interface and recording PC. In general terms, the camera’s parameter selection should try to maximize the frame resolution and FPS for all the cameras. Having larger images helps to capture smaller body part regions with more detail, and having a larger FPS enables better video quality in fast-moving scenes. For the DMD, more importance was given to the frame resolution, since the driving actions do not imply fast movements (except for blinking).

The frame resolution and FPS were selected attending the recommended allowed data bandwidth of USB 3.0 connections. The USB 3.0 “SuperSpeed” interface of the Intel RealSense cameras supports 5 Gbps. However, considering the encoding overhead, the raw data throughput is 4 Gbps. The USB specification considers it reasonable to achieve 3.2 Gbps. This could be an optimistic upper limit. As the USB protocol can support several devices connected to the same port; another analysis [70] showed that a device that requires over 30% of the bus bandwidth should be considered to be used separately from other devices in a single USB port. So, in general, Intel RealSense manufacturers recommend not to go further than $0.3 \times 4 \text{ Gbps} = 1200 \text{ Mbps}$ to ensure robust continuous streaming.

Then, as shown in Table 4.9 the camera configuration that is closest to this recommended bandwidth and maximizes the frame resolution is for images of 1280×720 pixels and 30 FPS in all three modalities. The three selected cameras were plugged into three independent USB controllers to have the maximum bandwidth possible.

RGB W×H×FPS (24 bits)	Depth W×H×FPS (16 bits)	IR W×H×FPS (8 bits)	Bandwidth 1 Camera (Mbps)	Bandwidth 3 Cameras (Mbps)
1920×1080×30	1280×720×30	1280×720×30	2157	6470
1280×720×30	1280×720×30	1280×720×30	1327	3981
848×480×30	848×480×30	848×480×30	586	1758
848×480×60	848×480×60	848×480×60	1172	3517
640×480×30	640×480×30	640×480×30	442	1327
640×480×60	640×480×60	640×480×60	885	2654
640×360×90	640×480×90	640×480×90	1161	3484

Table 4.9: Camera recording parameters and required bandwidths. The selected specifications for the DMD are highlighted in bold.

Moreover, image adjustments were made to improve the image quality of the recorded images. The selected cameras allow the definition of dedicated regions of interest (ROI) to perform auto exposure adjustment. This step was required to obtain well-lit images of the body of the driver. Then, for each camera, the ROI around the driver’s body parts was configured. The

ROI corresponds to the face, body and hands for each camera.

In addition, an important characteristic of multi-modal datasets is the requirement of having synchronized data streams. During the recording sessions, the three-camera streams were recorded using ROS bag format [177]. One advantage of the rosbag format is its ability to store all three camera modalities in a single file, along with accompanying metadata such as camera properties and timestamps. By utilizing these timestamps, it becomes possible to obtain perfectly synchronized multi-modal streams, since Intel Real Sense cameras have hardware synchronization capabilities for depth, color, and infrared modalities. However, rosbag files also add some overhead to the output, which increases the final size of the file and potentially impacts the recording bandwidth. The selected camera stream’s configuration has a bandwidth of 1327 Mbps, which is almost 10 GB per minute of recording per camera. To reduce the final size of the rosbag per camera, the lossless compression data algorithm LZ4 was applied, which reduces the final size of the videos to approximately 6 GB per minute.

During the recording sessions, after each group session (morning or afternoon) was completed, the recorded files were transferred to centralized network-attached storage (NAS). Although the recording PC was equipped with 2 TB SSD hard drives, the generated data volume of the cameras (ref. Table 4.9) easily fills the HD. This process was conducted during the break times between recording sessions to avoid processing overhead on the PC.

The raw material obtained has a total weight of approximately 26 TB. Figure 4.6 illustrates the distribution of the material based on the recording method used, as presented in Section 4.3.2.

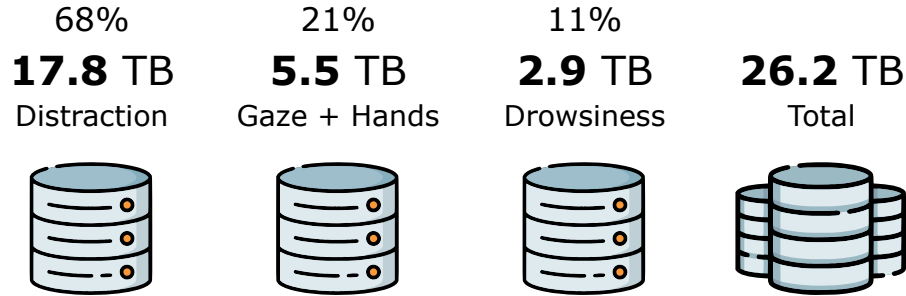


Figure 4.6: Total raw data volume in terabytes (TB) of the DMD dataset

The dataset comprises a total of 1062 rosbag files, each capturing the video feed from three cameras within the recording setup, containing the three video channels (RGB, IR and Depth). Consequently, it consists of 354 recordings, each featuring three distinct video streams. Figure 4.7 illustrates that among these recordings, 203 recordings depict drivers engaged in activities indicative of distraction, while 53 exhibit signs of drowsiness. Furthermore, there are 98 recordings available for analysis, showcasing gaze direction and hand positioning.

4.5 Dataset post-processing

After completing the recording session, all the captured material was centralized in a NAS and kept as raw data backup for future post-processing tasks. As the purpose of the DMD

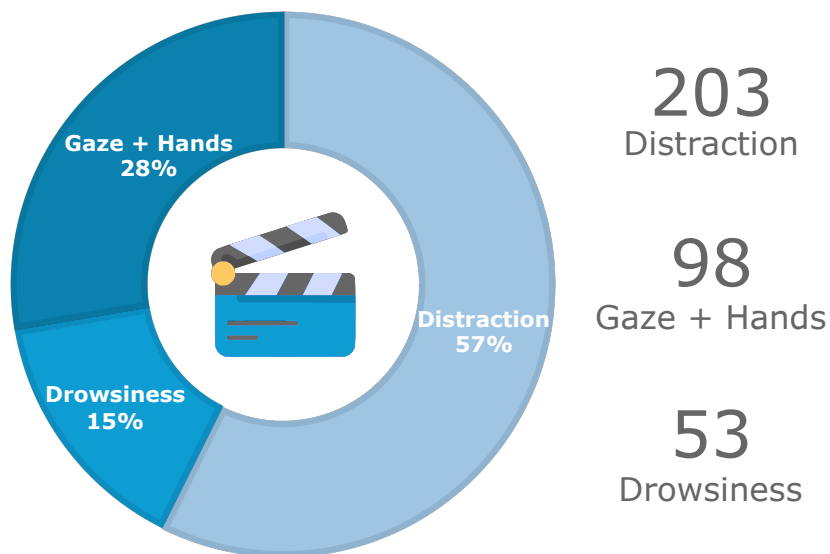


Figure 4.7: Distribution of recordings categorized by the exhibited type of driver behavior. Each recording consists of three sets of videos (RGB, IR, and Depth), corresponding to the three cameras utilized in the setup.

was to address the lack of data for various driver monitoring tasks, it was crucial to store the material in the highest quality possible, since different tasks may require varying compression qualities. However, to facilitate the use of visual material by annotation tools and deep learning training algorithms, a post-processing stage was implemented to export, calibrate, and synchronize the multi-camera streams. The visual content, including images and videos, was created using the FFMPEG library [59].

4.5.1 Stream Compression

Raw data is stored in the rosbag format (.bag), which is a useful format for recording and storing large amounts of data, such as multi-modal streams. However, to annotate, work with, and visualize the recorded material, it's often necessary to extract the data from the rosbag and convert it into more accessible formats. For this purpose, a custom Python script using the ROS library was developed to efficiently extract and convert the data from the rosbag into a more usable format, such as video or image files, while preserving the synchronized timestamps and other metadata.

- **RGB and IR:** The images are extracted and saved in PNG format with a size of 1280×720 pixels. After having the total number of images per stream extracted, they are encoded in a video with libx264 encoder and saved in MP4 format with a bit rate of 15,000 kbps and a frame rate of 29.98 FPS for hands and face camera videos and 29.76 FPS for body camera video. These frame rates were the ones stored by the cameras during the recordings.
- **Depth:** Depth data captures information about the distance between the camera and objects in the scene, measured in millimeters. This distance information is stored as an image, where each pixel represents the distance between the object and the camera. As

a result, depth data can be visualized as an image where each pixel contains a distance value instead of a color value, providing important insights into the 3D structure of the scene. This image is in 16UC1 format, which allows each pixel to have a 16-bit value in one channel. This pixel format is equivalent to gray16le.

To save as an image and preserve this pixel format, depth frames were saved in TIFF format (.tif). After exporting all images, the videos were encoded with FFV1 codec and saved in AVI format. The FFV1 codec is lossless and supports gray16le pixel format (images in 16UC1).

The resulting video is not intended for visual playback; instead, it contains distance information that is not recognizable by conventional video players. This decision was made to simplify the distribution of data since both the infrared (IR) and RGB modalities are also distributed in video format. By including the depth data in the same video format as the other modalities, it is possible to maintain synchronized data across all modalities while minimizing the amount of data that needs to be distributed and stored.

- **Metadata:** Inside the rosbag, there was also information about the recordings that were extracted and saved in the annotation file. This data is considered in the annotations list as scenario metadata (subject, context, video and camera information), shown in Figure 4.3.

4.5.2 Multi-sensor stream synchronization

When hardware synchronization is not possible, capturing with multiple cameras produces a small time shift (a few frames) in different streams. In this case, it happened that the 3 cameras operated asynchronously, hence, they did not start recording at the same instant; most of the time, the first camera to switch on and start recording was the face camera, then the body camera and finally the hands camera. Therefore, there is a frameshift between videos in every recording. Manually correcting this shift in every video per recording would require much effort and time. As a solution for this, it is proposed to use an automatic method to identify those shifts by extracting the “movement” signals of each camera, and calculating the absolute difference between consecutive frames from regions of interest in the image. The shifts are given when analyzing the correlation of the signals of the corresponding videos.

The intuition behind this is that the movements of the driver have the same pattern independent of the capture perspective. By understanding the normalized absolute difference of the images of two consecutive frames as “movement”, with the analysis of all the frames, the signals of “movement” per video can be extracted. These signals must correlate since they belong to the same moving person.

To be able to make these comparisons between the 3 cameras (face, body and hands perspectives), some regions of interest (ROIs) must be established, as shown in Figure 4.8a. The body camera and face camera are related by the driver’s face since this is the part of the driver visible from these two perspectives; the hand camera and body camera are related by the driver’s hands. To find the shifts, these ROIs were taken into account to produce signals

that could better correlate with each other. There is a signal per ROI and they are correlated depending on their relationship with each other.

It was found that better results were achieved in the correlation result if the signals were passed through a bandpass filter to reduce the signal amplitude. The goal is to get the most overlapping possible between the 2 signals. Some peaks or densities at the bottom of the signal could harm this calculation for the most overlapping rate. To filter the signals, the standard deviation σ was calculated and the band-pass filter was applied from $\max(x, \sigma)$ to $\min(x, 3\sigma)$ of the signals' amplitude. Moreover, the signal was cut 1/6 from the starting and ending sides. This process is shown in Figure 4.8b.

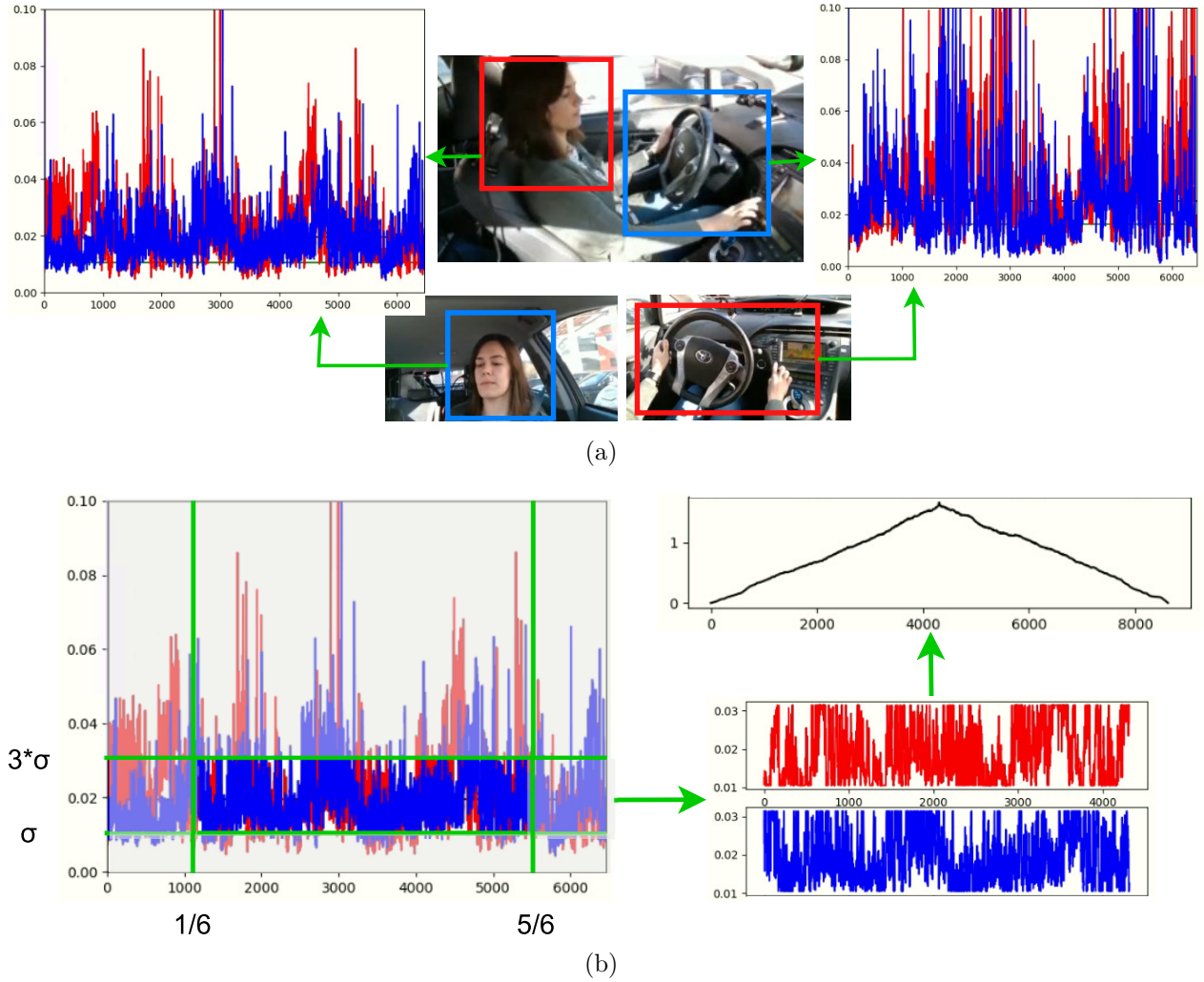


Figure 4.8: Process for multi-sensor stream synchronization: (a) Region of interest and signal extraction, (b) Processing of temporal signal and correlation calculation. The blue signal is computed from face and hands ROI in the face and body camera, respectively. The red signal is computed from face and hands ROI in the body and hands camera, respectively.

Then, the frameshift number is defined by the minimum length of the videos minus the point where the highest correlation was reached minus 1. Once the shifts between videos

were identified, an additional mosaic video with the 3 streams synchronized was created per recording session. This mosaic video is used in the next phase of the dataset preparation, for semi-automatic label annotation.

To demonstrate the effectiveness of applying the bandpass filter to the signals, the synchronization shift between the body and face cameras and between the hands and body camera was obtained using manual observation of each pair of streams. Then the automatic shift calculation was performed without and with the bandpass filter. It can be seen that the automatic calculation with the bandpass filter is closer to the manual calculation than the automatic calculation without the bandpass filter. This is because the bandpass filter reduces the amplitude of the signals and eliminates peak noise in the signals, which allows a better correlation between two stream signals.

As illustrated in Figure 4.9, a consistent improvement in the mean error between manual and automatic calculations is observed across all recording groups. These groups primarily differ in the participating volunteers and the time of day during which the recordings were conducted. Consequently, the extent of improvement may vary among groups. Nevertheless, applying the vertical and horizontal cuts as previously outlined yields a reduction in error.

In addition, the synchronization error analysis can also be done by observing the ability of the automatic method to infer correctly the frame shift for each of the recording sessions of the planned protocols. In Figure 4.10 the results of this analysis are presented. As can be seen, the method that does not include the signal cuts produces an overall average error of 2.8 frames for the body-face stream synchronization and 2.4 frames for the hands-body stream synchronization. The worse error is obtained for the recordings of the "reach behind" protocol. This is because the driver performs large body and face movements which may introduce higher peaks in the signal. Consequently, implementing the bandpass filter significantly reduces the average error.

4.5.3 Semi-automatic labeling

The generation of a visual-based dataset implies careful planning of a concatenation of several steps to finally obtain a collection of visual data and metadata annotations. The recording process, especially when a large amount of data is collected such as in the DMD, has to be performed correctly to make the process of annotating the results easier. That is why the focus is on the definition of the whole framework process of creating a large set of data. In this process, an important part is the generation of metadata annotations. Annotating a large dataset is not an easy task, especially when the projected annotations include multiple dimensions of a computer vision problem, such as in DMS. To annotate the DMD, *semi-automatic labeling* [184], [209] methodology aimed to reduce annotation times is applied. The process consists of the generation of the first set of labels based on weak models. Then, a human-in-the-loop approach is included to provide manual annotations and recursively improve the label's quality.

For the temporal annotations of the DMD, a tool was developed to make annotation tasks easier and faster to complete (TaTo, Temporal Annotation Tool, version V1 [25]); it is Python-based and allows the visualization of frame-by-frame annotations by colors on a

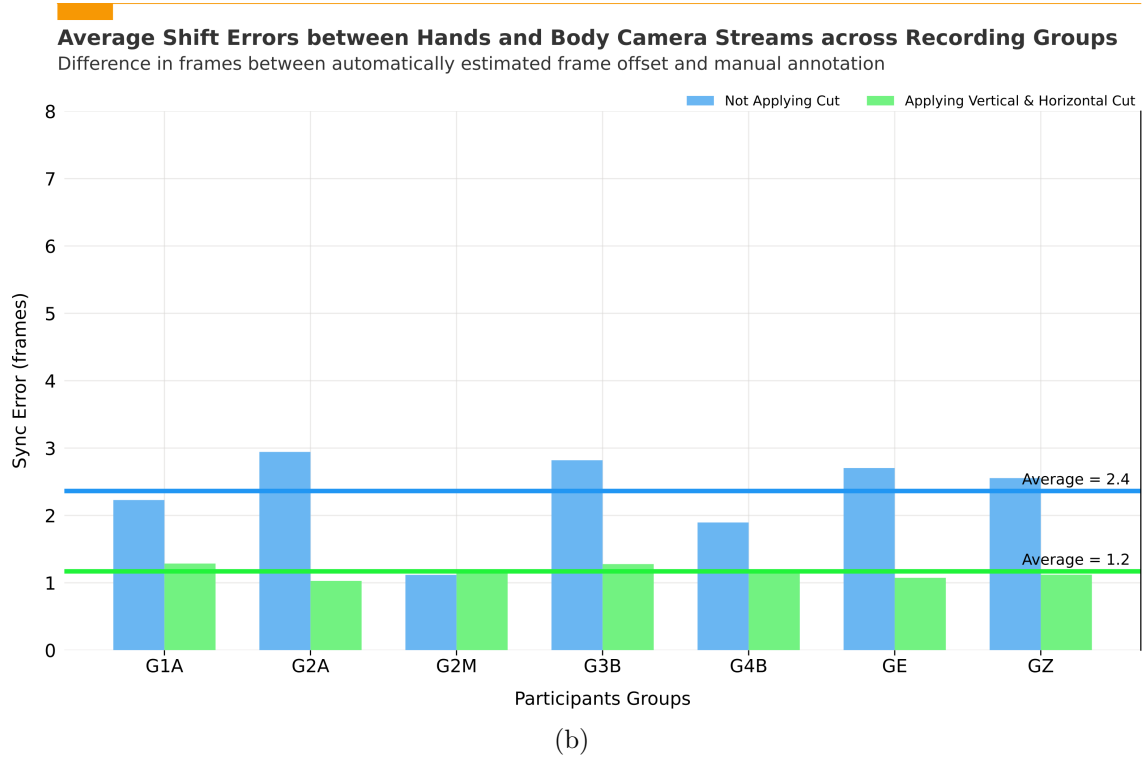
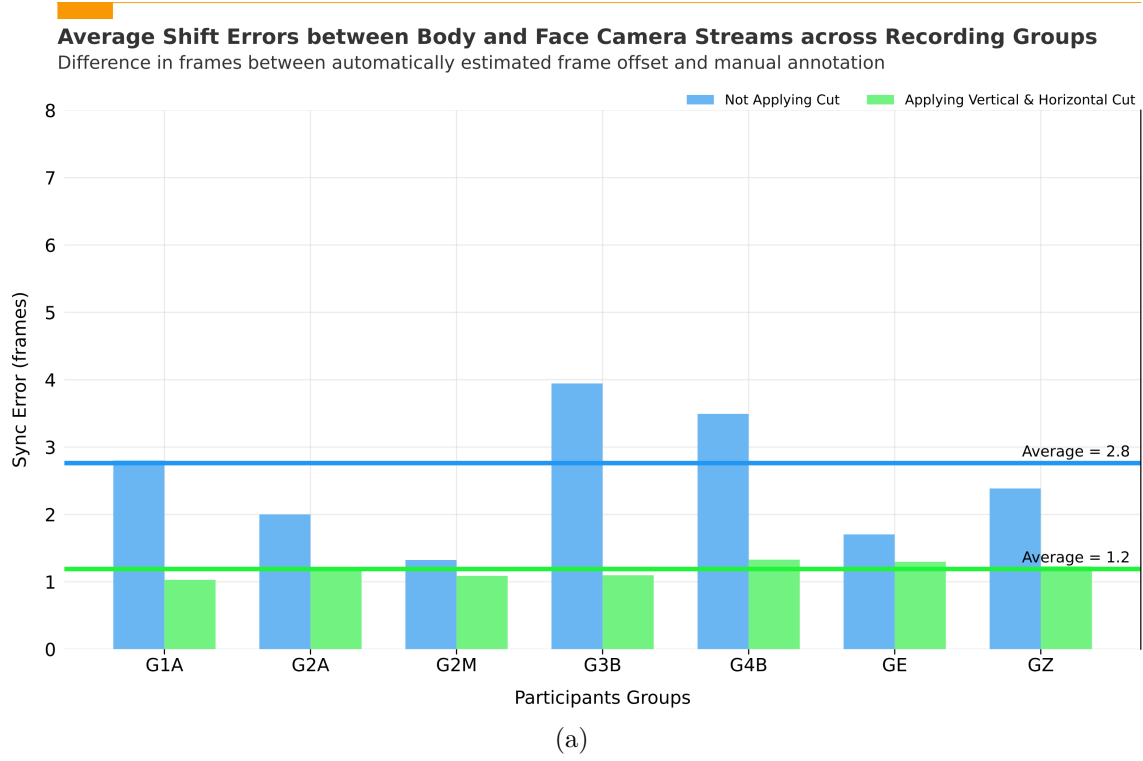


Figure 4.9: Comparative analysis of the average synchronization frame shift error for participant groups upon implementing vertical and horizontal signal cuts to the movement signal: (a) Synchronization of Body and Face camera streams, (b) Synchronization of Hands and Body camera streams.

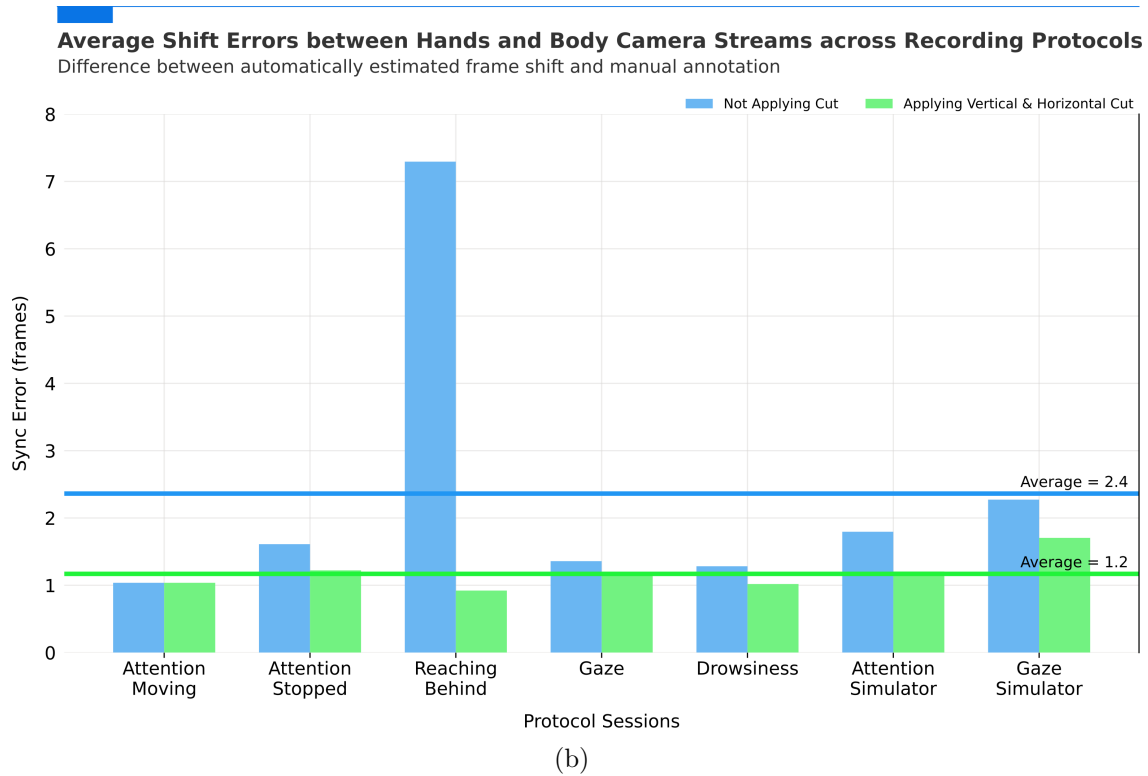
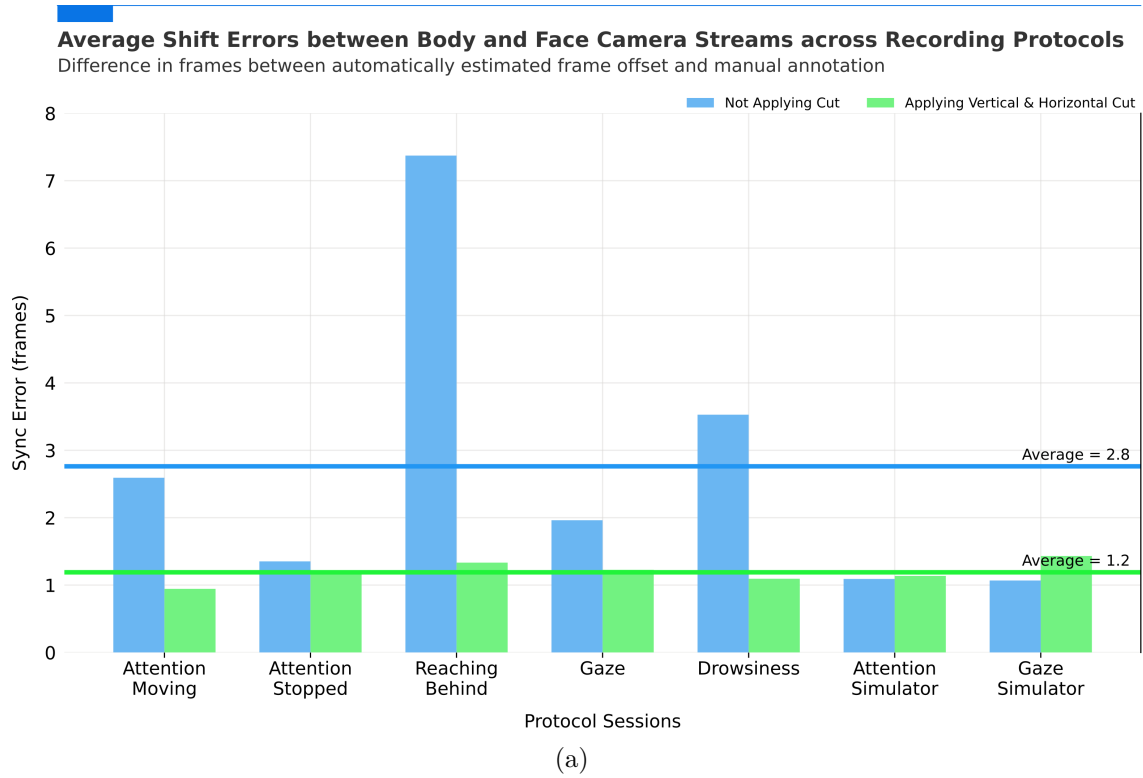


Figure 4.10: Comparative analysis of the average synchronization frame shift error for recording protocols upon implementing vertical and horizontal signal cuts to the movement signal: (a) Synchronization of Body and Face camera streams, (b) Synchronization of Hands and Body camera streams.

timeline, which makes this process more intuitive. The output metadata was structured using OpenLABEL annotation format [6], [149].

A second version of the annotation tool (TaTo V2) was extended to add some *semi-automatic labeling* principles with new strategies to create some annotations automatically and improve labeling time:

- **Pre-annotations predicted by model:** Only few samples of the DMD were manually annotated for the driver actions with TaTo. These were then prepared to become training material. Then, a model is trained using transfer learning from a Mobilenet [93] originally trained with ImageNet [37]. The predictions were taken as pre-annotations and the annotator’s job was to correct and continue annotating instead of starting from zero.
- **Logical relations among levels of annotations:** There exist some logical relations between levels of annotations that, taken into account, could save time in the labeling process. For example, if the driver is sending a message with the left hand, this should be annotated in the `driver_actions` level as “texting-left”, but this action also could imply that the person is only using the right hand to drive (since the left hand is being used to text). Therefore, in the `hands_using_wheel` level, there should be the “only right” label. Other relations are the talking-related labels as “phonecall-left” in the `driver_actions` level and “talking” in the `talking` level. In the annotation tool TaTo, this functionality of applying logical annotations is integrated; once the driver action level annotations are complete, a dedicated key press propagates these logical annotations to the remaining levels. This approach prevents annotators from having to begin annotating from scratch.

The TaTo tool, in its second version, offers the advantages of pre-annotations described above. To know the impact of pre-annotations, the time of annotation spent on videos from different recording sessions with both versions of the tool (V1 does not offer pre-annotations) was measured. The results of the improvement in annotation times are shown in Table 4.10. When comparing the annotation time per video between the different versions of the tool, it becomes apparent that there has been a significant improvement in efficiency, with an average reduction in annotation time of 66%.

This not only shows that semi-automatic annotation strategies can bring advantages over a traditional manual annotation process but also underscores the importance of incorporating such strategies into the dataset creation process to achieve more efficient and accurate annotation, ultimately resulting in better quality datasets for training and testing machine learning models.

Semi-automatic labeling strategies offer a range of additional benefits over traditional manual labeling approaches. Firstly, they enable faster labeling of large datasets, reducing the time and cost required for manual annotation. This leads to improved efficiency and can ultimately result in significant cost savings. Secondly, automated algorithms used in semi-automatic labeling provide a more consistent labeling process, minimizing human errors and ensuring high-quality data. Thirdly, semi-automatic labeling techniques can be customized to suit different labeling requirements and data types, allowing for more efficient annotation across a wide range of applications. Fourthly, the combination of human annotators and machine

Session	TaTo Version	# Videos	Total Time (h:min:s)	Time/Video (h:min:s)	Improvement
s1	V1	4	13:02:00	3:25:00	56.10%
	V2	1	1:30:00	1:30:00	
s2	V1	3	16:10:00	5:36:00	63.69%
	V2	2	4:04:00	2:02:00	
s3	V1	2	2:20:00	1:10:00	81.43%
	V2	3	0:41:00	0:13:00	
s4	V1	2	19:45:00	8:22:00	63.55%
	V2	3	9:10:00	3:03:00	

Table 4.10: Comparison of annotation times using v1 and v2 of TaTo

learning algorithms in semi-automatic labeling can result in more accurate and reliable annotations, improving the overall quality of the labeled data. Finally, semi-automatic labeling approaches can be scaled up to handle large volumes of data, making it possible to annotate datasets of any size or complexity, and enabling the creation of more comprehensive and diverse training sets for machine learning models. Overall, these additional benefits make semi-automatic labeling a powerful tool for efficient and accurate dataset creation.

4.6 DMD discussion

4.6.1 Published material of the DMD: temporal action recognition

Distraction and drowsiness detection are use cases of the DMD. The first approach for distraction detection was made in [162], where the DMD material was annotated with some distraction-related labels that led to the definition of a training dataset called dBehaviourMD. Other iterations in the annotation process resulted in the creation of labels for distraction detection that were used in the research of [25].

Following these efforts and thanks to the metadata methodology and sensor configuration defined in this work, a descriptive and detailed annotation criterion was built which can support distraction detection through action recognition algorithms. The annotations defined for this specific task are temporal, and the labels, organized by levels, are presented in Table 4.2.

The label organization by levels of annotation allows making a multi-label annotation of the videos, offering a better description of the scene in one frame. As a result, this temporal distraction-related part of the DMD has the class distribution, per annotation level, presented in Figure 4.11.

This part of the dataset comprises 1,837,041 frames, equivalent to approximately 1021 minutes of video. It's important to note that the percentages of labels presented in the diagram

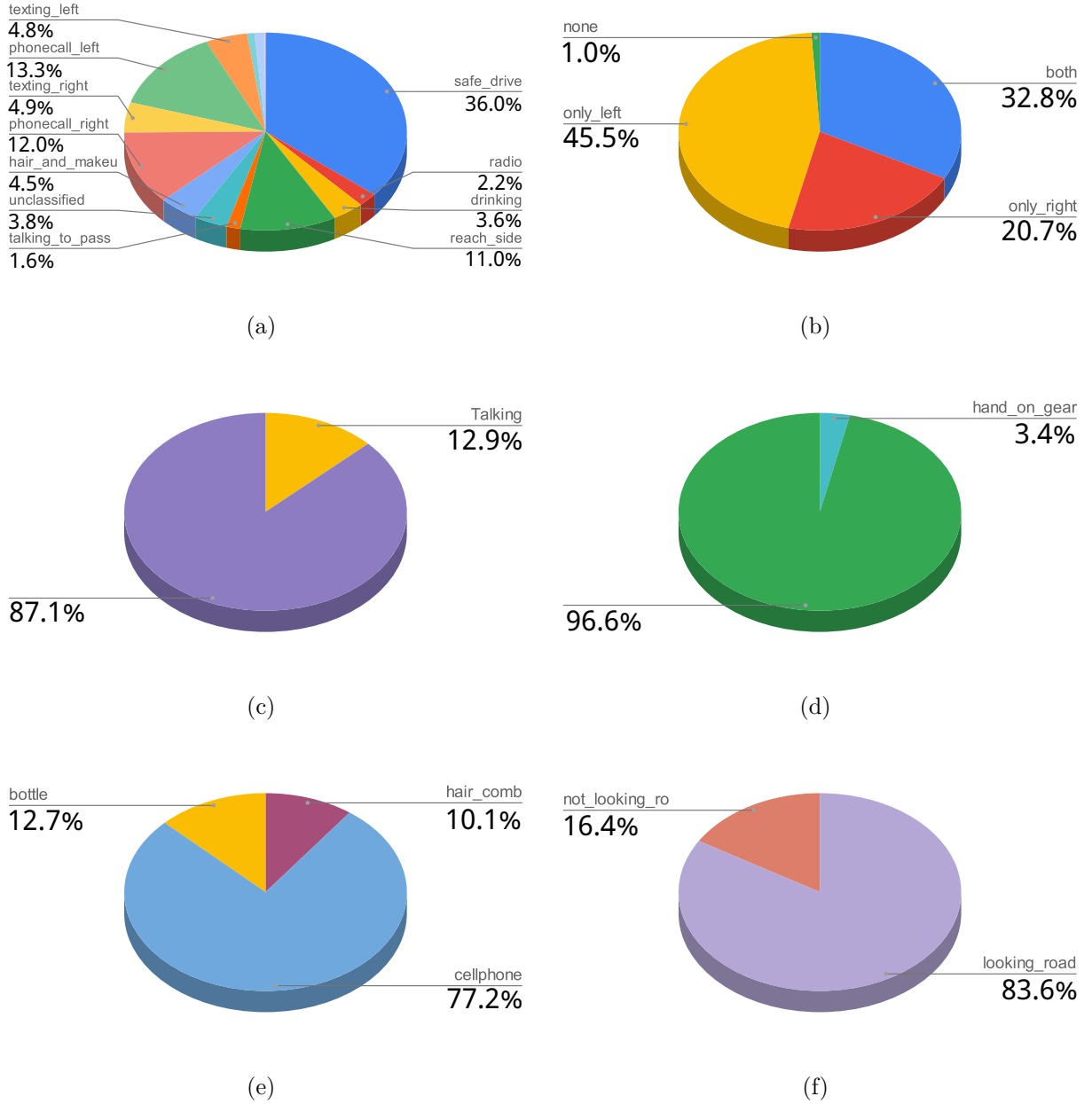


Figure 4.11: DMD classes distribution for: (a) Driver_actions (b) Hands_using_wheel (c) Talking (d) Hand_on_gear (e) Objects_in_scene (f) Gaze_on_road.

correspond to the proportion of annotated frames within each level of annotation, rather than the overall number of frames. For example, for the frames annotated with the “gaze on road” level criteria (frames where the face camera is available), in 84% of them the person is looking at the road and in 16% is not looking at the road.

Recently, sequences obtained through the drowsiness protocol and gaze/hands protocols underwent annotation. Annotation instructions were defined to guide a team of annotators through this process. Consequently, the labels outlined in Table 4.3 for drowsiness and

Table 4.4 for gaze were generated through this annotation process.

Since there are few datasets dedicated to driver monitoring, and to better contribute to the scientific community, the DMD has been published on its website ¹ for free to download under non-commercial use agreement. At this point, about 80% of the material related to distractions, drowsiness and gaze activities of the DMD is public, the other 20% will remain unpublished for future benchmarking exercises. The annotations files use the OpenLABEL content description format [6], [149] and the content of the annotations follow the label taxonomy defined in Section 4.3.1.

4.6.2 Ongoing and future work

A procedure has been developed to ease access to the DMD. The steps for the download process are shown in Figure 4.12. On the website, there is a download form where the users enter their contact details. This form is sent to a mailing service platform where this information is stored. For each record added, the storage service trigger is activated to give access to the registered email user to the DMD. Once access is given to the storage service, an email is sent to the user notifying them that they can now obtain the dataset. By this time, 1364 users ² have requested the dataset through the website form.

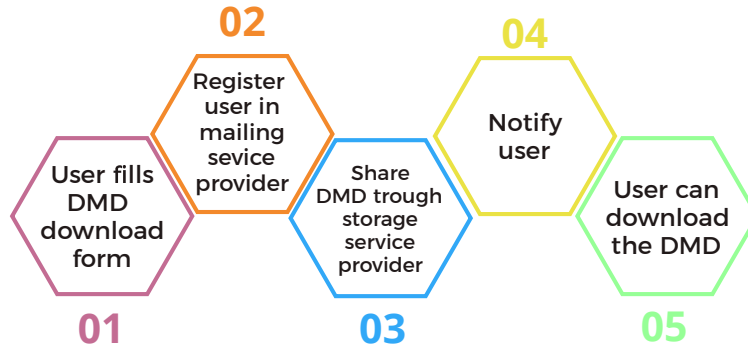


Figure 4.12: Procedure to distribute the DMD

An adequate structure to share the dataset was also defined. The material is distributed mainly by groups; the same groups in which the recordings were organized. Then, it is organized in folders by subject to finally group the material by recording sessions. Regarding the nomenclature, each video has in its filename information about the group to which it belongs (A, B, C, etc.), the subject (1, 2, 3 and ...), the session (s1, s2, s3 or s4), the timestamp when it was recorded, the information channel (RGB, depth and IR) and the camera (body, face, hands and mosaic) or annotation (ann_distraction). This structure is represented in Figure 4.13.

The construction and preparation of the Driver Monitoring Dataset (DMD) is an ongoing project aimed at creating a comprehensive and dynamic repository of driver behavior data, adapting alongside advancing technologies [163]. Presently, the research community has access

¹<https://dmd.vicomtech.org/>, accessed on 15 March 2024

²Total requests at 15 March 2024

to the initial release of the DMD, which includes a diverse array of multi-modal videos and temporal annotations pertaining to distraction, drowsiness, and gaze behavior across a varied demographic of drivers. Future efforts within the DMD project involve annotating geometric features outlined in the taxonomy (see Section 4.3.1) and generating additional recordings.

As outlined above, the planned enhancements to the DMD entail a diverse range of annotations structured around the proposed metadata taxonomy (see Figure 4.3). This taxonomy can be extended if new DMD recordings are generated which could include new scenarios, for instance driver interaction with other objects different than the defined initially. The primary goal is to continually advance the annotation process of the DMD, thereby expanding its potential applications. Furthermore, the DMD is designed as an open project, encouraging community involvement and utilization. Consequently, ongoing efforts will involve consistent publication of both materials and tools developed for dataset processing and annotation.

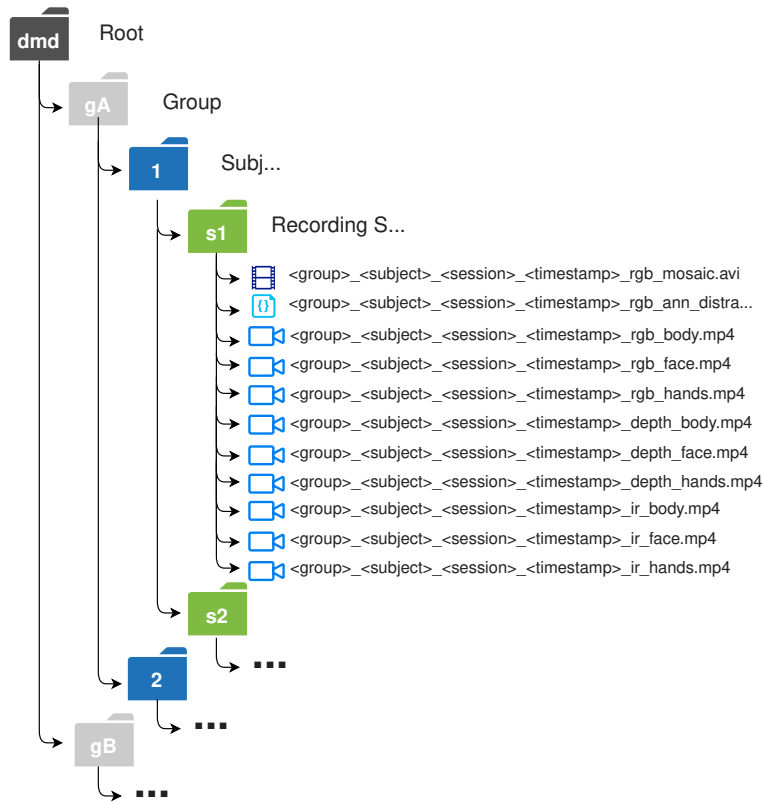


Figure 4.13: DMD file structure

4.7 Conclusions

Visual data is one of the most important assets to build robust and complete machine vision systems. The process of generating a vision-based dataset is not a trivial and easy task and requires careful planning and execution to avoid redundant work and provide the material that meets the volume, variety and quality requirements of computer vision systems.

In this section of the thesis, a methodology is introduced for creating a multi-modal, multi-

sensor computer vision dataset specifically designed for driving monitoring applications. Furthermore, this approach is showcased in the construction of the DMD, a robust dataset aimed at improving vision-based driver attention and alertness monitoring systems.

It was demonstrated that a previous phase of analysis of the DMS algorithm requirements based on the study of driver’s physiological signs of inattention (for both fatigue/drowsiness and distraction) is needed to define the expected taxonomy of events that are needed to be recorded and later annotated. In the presented framework, it was shown, it is possible to define several capture protocols that cover the required annotation taxonomy. In addition, based on the recording planning, the selection of the used sensors was specified. For the DMD, the required sensors include several RGB and depth cameras.

Since the DMD was targeted to collect a high amount of driving data, this chapter presents the challenges of the post-processing phase, where the raw data was prepared and formatted for consumption by annotation tools and DL training algorithms. At the time this thesis was prepared, the DMD is one of the largest and richest public datasets for developing multi-modal video-based driver monitoring systems. As a measure of the relevance the DMD has acquired in the field, the two main related publications [161], [162] have been cited in 83 international research publications.

Once the data requirements are addressed through a dataset creation strategy, resulting in an annotated dataset such as the DMD, the development of contemporary vision-based DMS necessitates the identification of algorithms and method characteristics that can extract visually discernible features to evaluate the driver’s attentiveness. These characteristics facilitate the establishment of a software framework capable of supporting various computer vision algorithms. By defining this framework, it becomes possible to establish shared interfaces between methods, streamlining development efforts and optimizing computational resources.

Chapter 5

Real-time modular software framework

5.1 Introduction

Systems that measure the attentiveness state of the driver are known as Driver Monitoring Systems (DMS). There have been numerous research initiatives aiming to give steps towards the introduction of DMS in vehicles: VI-DAS, Adas&Me, HaveIT, Deserve, Interactive, and AdaptiVE, are some examples of European-funded projects in which efforts to introduce DMS advanced driver assistance systems were developed and tested [48]. For instance, the VI-DAS project¹ adopted the design of a next-gen 720° connected ADAS approach by integrating both surround scene analysis and driver status monitoring. Advances in sensors, data fusion, machine learning and user feedback provided the capability to better understand driver, vehicle and scene context, facilitating a significant step along the road towards truly semi-autonomous vehicles. On this path, there is a need to develop tools that allow rapid integration of driver monitoring systems in the design of vehicle automation that can efficiently and securely hand over and back control to the driver [14].

This DMS framework is designed to streamline the integration of DMS functions by component integrators, including OEMs or Tier 1 suppliers, with various vehicle systems. These systems encompass exterior environment perception, local dynamic maps (LDM) localization, vehicle control, and human-machine interfaces. By employing this framework, the process of integrating DMS functionalities becomes more efficient and facilitates seamless interaction with other critical components of the vehicle.

The advances of ML and DL have rocketed the development of ADAS and autonomous vehicle systems based on CV. In particular, DL has allowed the improvement of accuracy of a variety of outside environment sensing systems [120], [123]. Those improvements have been applied to detect the surrounding objects such as vehicles [196], pedestrians [170] or traffic signs [105] with high accuracy levels. Moreover, DL methods have also been used to analyze human faces and bodies in the context of driver monitoring. Most of these systems use visual information of the driver to determine the attentiveness level and take actions [58]. However, DL-based methods have increased the computation needs of hardware architectures, especially

¹<http://www.vi-das.eu/>

when several systems run simultaneously. In addition, many of the functionalities integrated into vehicles can be categorized as safety-critical, meaning that it is expected to perform in real-time to deliver on-time the perception information to the situation awareness modules. Therefore these requirements make it crucial to have an efficient software architecture that seizes the advantages of DL in CV without compromising performance and accuracy.

To achieve these objectives, using camera sensors for driver monitoring is becoming the trend in the automotive industry due to the reduced cost of the hardware and the great advances in CV and machine learning technologies for human analysis [58], [155], [193]. In addition, these passive sensors result very convenient in automotive scenarios due to their non-intrusive feature compared to other physiological sensors such as ECG, EEG and EOG [27], [138].

Typical methods available in the context of driver monitoring include algorithms for human face and body analysis. Nowadays, it is possible to find several implementations of CV algorithms that extract information from faces and body. For instance, open source libraries such as OpenCV, DLib [115], OpenFace [10], etc. already provide robust algorithms and models to perform tasks such as face detection, 2D landmark model fitting, head pose estimation or gaze estimation. Based on these perception solutions, different ad-hoc architectures that tackle specific responses of driver state (i.e. drowsiness [133], distraction [118], [138] or cognitive workload [129]) have been proposed.

However, modifying a driver monitoring application to incorporate newer perception models and sensors or additional processing steps can be difficult due to interface differences between the algorithm steps. Moreover, the integration of these driver monitoring pipelines into the whole intelligent vehicle system is very time-consuming and usually involves optimizing the performance and homogenizing the data interfaces. This lack of a global perspective and scalable capabilities limits their application in fully integrated intelligent vehicles. Hence, it is necessary to find a common framework for applying vision-based methods that integrate multiple DMS functions with other ADAS/AD functions such as obstacle detection, visual odometry, traffic sign recognition, etc.

The work proposed in this thesis addresses these challenges by providing three contributions: (i) a criteria for classifying features typically used in DMS, attending to the physiological characteristics of the driver, (ii) the definition of a software framework for efficiently combining those features in structured software pipelines which can be used for rapid prototyping, validation and integration of DMS in modern intelligent vehicles, and (iii) the application of this framework in industry-level use cases for ADAS and semi-autonomous vehicles. This framework is implemented as a software library called *DMSLib* which is compatible with the automotive-level industry platform RTMaps for real-time multi-sensor data processing.

5.2 Related work

5.2.1 Algorithms and frameworks

There is a lack of consensus to define the driver's state, which causes several problems. First, it can make the interpretation and comparison of research findings across studies difficult,

or even impossible. Secondly, similar works may be measuring slightly different constructs and measuring different outcomes. Usually, driver state has been related to driver inattention and driver distraction (see Chapter 2). In the same way, the works found in the literature show inconsistency in the definition of these two physiological mechanisms. Some authors have defined complex taxonomies to characterize driver inattention in terms of the different mechanisms by which inattention may arise [173].

Variation in driver state can be produced by two main causes: (a) distraction and (b) fatigue. Both of them are grouped into the driver's inattention mechanisms which can show different physical and physiological responses. In the past years, different attempts have been made to understand and assess the driver's state through a variety of objective and subjective methods to mitigate the effect of human errors and reduce accident fatalities [109]. The efforts made to develop DMS have contributed to improving existing ADAS and In-Vehicle Information Systems (IVIS).

The purpose of a DMS is to monitor the attention state of the driver. If driver inattention is detected, different countermeasures should be applied to maintain driving safety, depending on the types and levels of inattention. In the past decade, there has been an active interest in this type of system. The first international conference on driver distraction and inattention was held in 2009. Some attempts were made to understand and assess the driver's state through a variety of objective and subjective methods to mitigate the effect of human error and reduce accident fatalities. However, there is still a great need to develop more reliable and fully functional DMS systems using cost-efficient methods in a real driving context. The advances in signal processing and computer vision techniques are now generating great interest in this field as it could be possible to develop reliable DMS. Hence, introducing elements of DMS to existing ADAS and IVIS is of vital importance for future vehicles to increase safety and driving comfort.

This section analyses the main works devoted to detecting diversion in driver attention by means of measuring and identifying some of the visual, physiological or performance signs of inattention described in the previous section. It is separated according to the following types of measurements:

- Subjective measures
- Driver physiological measures
- Driving performance measures
- Driver Visual measures
- Hybrid measures

Subjective measures

Subjective approaches involve asking users directly about their fatigue, attention, and workload levels. While these approaches are relatively easy to implement, they are not automatic. Unfortunately, they may not be suitable for commercial DMS, as they typically rely on self-reported measures taken before or after driving activity. As a result, they cannot provide

a real-time reference of the driver state. Despite this limitation, these methods can still serve as a valuable starting point for objective DMS.

One of the most commonly used tools for subjective self-assessment of sleepiness is the Karolinska Sleepiness Scale (KSS) which can categorize the driver’s fatigue and drowsiness in 9 levels [107] (see Table 5.1. A study [5] investigates the reliability of the KSS using EEG, behavioral, and other subjective indicators of sleepiness. They showed that the scale is highly correlated with EEG and behavioral variables. They conclude that the KSS has high validity for measuring sleepiness.

In [95], the relation between the KSS and blink duration and lane drifting (measured as the SDLP in a high-fidelity moving-base driving simulator) was studied. The authors found a close relationship between these subjective and objective measurements. Their experiments showed a significant effect of the KSS on both blink duration and SDLP.

KSS	Description
1	Extremely alert
2	Very alert
3	Alert
4	Rather alert
5	Neither alert nor sleepy
6	Some signs of sleepiness
7	Sleepy, no effort to stay awake
8	Sleepy, some effort to stay awake
9	Very sleep, great effort to keep awake, fighting sleep

Table 5.1: Karolinska Sleepiness Scale (KSS) [107]

Physiological measures

Today, researchers explore various objective methods to measure driver state which implies the use of intrusive measurements that correspond to certain physiological phenomena, such as heart rate (pulse detection), brain activity (EEG), eyelid closure and blinks (eyelid sensor) or skin conductance (electrodes) [139].

These physiological phenomena can be divided into two main categories. The first category consists of biological measures, which correspond to internal measurable parameters such as heart and brain activity, electrodermal activity, temperature, etc. The second category consists of physical visual measures, which include eyelid activity, eye gaze, head movement, body pose, and other observable physical characteristics. To obtain a measure of driver state (fatigue, distraction, cognitive workload and stress) different correlations between these indicators and the current driver state are performed. However, the intrusive typology of these sensors makes them difficult to be implemented in commercial applications.

Driving performance measures

The in-cabin human state can be analyzed first by observing direct and indirect cues. Direct methods involve analyzing the physiological cues of the user to obtain features such as head pose, eyelid aperture, gaze direction, feet, body and hand pose, brain activity, heart and skin parameters, etc.; either using intrusive or unobtrusive sensors [156]. Indirect methods involve extracting the activity of onboard sensors such as vehicle dynamics, phones, smart watches, or GPS. Then, based on the direct and indirect measures, it is possible to infer and predict the human state and behavior [64].

Driver visual measures

Other methods use computational algorithms to sense and predict the state of the driver. CV and ML such as gaze-tracking [77], eye tracking[63], body and head pose [103] are promising approaches to solve this problem as they use unobtrusive sources of multitude information of driver condition. By analyzing visual-based metrics that describe various characteristics of the physiological state of the human, the method is capable of predicting the temporal relationship of the data to generate a driver state model [114].

Face and eye movement features (e.g. head pose, eye blinks, gaze direction, yawning, etc.) are typically considered for deriving driver cognitive states, such as driver distraction and fatigue [40]. In particular, eye-related metrics have shown very promising results to infer the fatigue state of the driver[193]. Actually, some car manufacturers have released basic DMS in which eye analysis is included to monitor driver state [180], [213].

The most relevant CV processes to detect driver inattention (fatigue and distraction) include:

- Head orientation detection
- Face detection
- Face landmark detection and modeling
- Eye detection
- Pupil detection
- Eyelid aperture modeling
- Yawn detection

By a mixture of these CV techniques and ML methods, many authors have developed different architectures to determine the driver state.

Fatigue detection using visual measurements Most of the works for fatigue detection using visual features are based on the analysis of the face and eyes. In [15], PERCLOS, eye closure duration (ECD), blink frequency, nodding frequency, fixed gaze, and frontal face pose were normalized and used as inputs to a fuzzy inference system (FIS) for fatigue detection. A FIS is famous for its well-known linguistic concept modeling ability. The fuzzy rule expression is close to an expert natural language. A fuzzy system then manages the uncertain knowledge and infers high-level behaviors from the observed data. On the other hand, because it is a

universal approximator, FIS can be used for knowledge induction processes. Finally, three variables (fixed gaze, PERCLOS, and ECD) were determined to be crucial cues for detecting a driver's fatigue. By fusing them with a fuzzy system based on fuzzy decision trees (FDT) [94], a final fatigue detection accuracy of 98% was achieved.

In [57], first the face was located. Then, Gabor wavelets were applied to the face area to obtain different scale and orientation features of the face. Features on the same scale were fused into a single one to reduce the dimension. Finally, an AdaBoost algorithm was used to extract the most critical features from the dynamic feature set and construct a strong classifier for fatigue detection. It was reported that this method worked well on a wide range of human subjects with different genders, poses, and illuminations.

The work done in [66] produces as a result a classification of 18 features of eye movements for drowsiness detection. They construct the feature vector by using sequential floating forward selection (SFFS) to select the most relevant features to construct a statistical classifier. The advantage of SFFS over feature transform techniques (such as principal component analysis (PCA) or linear discriminate analysis (LDA)) is its high transparency because the selected features remain unchanged. An artificial Neural Network was trained to detect drowsiness. They conclude that as long as the blinking signals were correctly detected (high confidence), the drowsiness detection accuracy could reach 82.5%.

There is a substantial body of research suggesting various methods for fatigue detection. Each of these methods typically involves a feature extraction step, followed by the application of a machine learning classifier. Machine learning algorithms, such as Support Vector Machines (SVM), Bayesian Networks, and Adaboost, have been employed in this context. Additionally, deep neural networks have gained considerable attention in recent years due to their promising capabilities.

A mixture Gaussian model was used in [34] to model the “normal behavior” statistics from the eyes closure duration and frequency of eye closure for each person to identify anomalous behaviors. This work derived the following three factors from the blinking waveform: 1) the length of a blink; 2) the closure rate; and 3) the blink rate. These factors were then weighted using a multiple regression analysis for each individual to calculate the drowsiness level.

In addition to analyzing eye movements, some researchers also examine mouth activity as a means to assess driver drowsiness. Fan et al. (2010) employed Linear Discriminant Analysis (LDA) as a feature extractor to categorize mouth activity into two states: 1) normal, and 2) yawning.

Distraction detection using visual measurements Visual distraction usually involves detecting the head orientation and eye gaze of the driver. A complete review on head pose estimation could be found in [145].

In [116] two different algorithms for gaze-based driver distraction detection based on the eye-tracking data were described and compared. One algorithm relied on the metric “percent road center” (PRC) of gaze direction, where a PRC of more than 92% was considered indicative of a gaze concentration that results from cognitive distraction, whereas a PRC below 58%, computed over 1 min, was an indicator of visual distraction. Fixations were used for the

computation of PRC. The second algorithm was based on a 3D world model with different interior zones such as the windshield, speedometer, mirrors, and dashboard and on the time that the driver spends glancing at those zones. When the driver looks away from the “field relevant for driving” (FRD) a timer is activated and decreased an attention buffer of a maximum of 2s. When the driver’s gaze returns to the FRD the buffer increases. When the buffer reaches zero, the driver is considered distracted. When further conditions were met (direction indicator not activated, speed above 50 km/h, no brake activation, and no extreme steering maneuvers), a warning was issued. The results showed that both algorithms have potential for detecting driver distraction and fully attentive drivers had a PRC of about 70% to 80%.

The use of eye gaze information and head pose is very common in driver distraction systems. For instance, in [166] the distraction level is modeled by using this information. In this study, the visual distraction level was assumed to be non-linear. This means that the visual distraction level is time-dependent on the visual focus. When the driver looked away from the road scene, visual distraction increase with time; but when the driver refocused on the road scene, it nearly instantaneously decreases. The authors first used a distraction calculation to compute the instantaneous distraction level. Then, a distraction decision maker determined whether the current distraction level represented a potentially distracted driver.

5.2.2 DMS regulatory environment

The introduction and advancement of automated driving were accompanied by high expectations at the beginning of the century. However, the anticipated technological progress fell short of fulfilling these promises. It wasn’t until 2022 that the first L3 systems became available in vehicles, such as those offered by Mercedes [140] and Honda [89]. Other manufacturers, including Hyundai and Volvo Cars, are expected to also present L3 systems. Currently, L3 systems are restricted to lower speeds (up to 60 km/h), specific highways with well-defined conditions, and traffic congestion scenarios in favorable weather conditions. Overcoming the challenges posed by higher speeds and the implementation of L4 automation systems will require tackling numerous complex obstacles.

Currently, OEMs are focusing their development efforts on bridging the gap between L2 and L3 automation levels. This has resulted in the emergence of L2+ automated vehicles [181], which are equipped with advanced L2 functionalities. These vehicles can drive autonomously in a wider range of scenarios but still require the driver to continuously monitor the driving task. Most importantly, the driver must be prepared to regain control quickly if the automation fails. The complexity arises when attempting to transition from level 2 to level 3 or higher automation levels, as complete reliance on automation and removing the driver from the driving task presents significant challenges. In L3 automated vehicles, the system must accurately predict when the driver needs to take over and assess the driver’s ability to do so effectively. These additional requirements, which do not apply to L2 or L2+ automated vehicles, pose complex challenges that are unlikely to be fully resolved in the near future. Therefore, the prevailing trend is to keep the driver engaged in the driving process, even as the control becomes increasingly automated.

The introduction of SEA L3 and higher levels of automated vehicles on the market is dependent on the regulations of these new systems. A distinction can be made between two different kinds of regulations: regulations pushing new ADAS to be integrated into vehicles entering the market and regulations allowing cars with higher automation systems to enter the road.

The transition to making DMS mandatory in new vehicles has already begun in the European Union. In 2019, the EU's Council of Ministers approved a new Vehicle General Safety Regulation [53], which mandates that from July 2024 all newly produced automobiles in the EU be equipped with enhanced safety features, including driver monitoring technology.

Currently, these regulations apply exclusively to new vehicles with some level of automated driving capabilities. However, by 2026, they will be expanded to encompass all newly manufactured cars in the European Union. The introduction of this legislation is expected to have a substantial impact, potentially preventing over 100,000 car accidents in the following decade. As of July 2022, the regulation has already come into force, mandating new cars to be equipped with an attention warning system to address driver drowsiness. This system measures the driver's alertness and issues warnings when necessary. Furthermore, between July 2024 and July 2029, new vehicles must gradually incorporate an advanced driver distraction warning system. This system aids drivers in maintaining focus on the traffic situation and alerts them when they become distracted.

In addition to legislative measures, the EU is actively promoting DMS through its New Car Assessment Program, EuroNCAP. To achieve a five-star safety rating, new cars are now required to incorporate driver monitoring features. EuroNCAP is enhancing its protocols to evaluate higher levels of automation, as outlined in their Vision 2030 [51]. These updated protocols emphasize the importance of DMS, with penalties and rewards linked to the performance of assisted driving technologies. This framework encourages original equipment manufacturers (OEMs) and Tier 1 suppliers to enhance their systems. The EU recognizes the growing significance of assisted and automated technologies in reducing traffic accidents and evaluating the effectiveness of these functionalities.

Moreover, China, a country where more than 4 million automobiles were sold in 2018 prior to the pandemic's sales impact, is gradually advancing the implementation of DMS as a mandatory safety feature [200]. However, progress in this regard has been slower compared to the European Union. It is expected that a nationwide mandate will be introduced in the near future. Given the influential positions of the United States and European markets, it is reasonable to expect that the world's most populous country will make substantial investments in driving safety.

In the United States, Congress is also taking legislative action to mandate driver monitoring systems in new vehicles. In July 2020, the House of Representatives passed the Moving Forward Act, a bill aimed at enhancing driving safety by investing in infrastructure and road safety measures. This bill contains a provision similar to the EU's safety regulation, which would make DMS mandatory for all newly manufactured vehicles sold in the US [7]. Although the Moving Forward Act still awaits approval from the Senate and the President, comparable legislation has been introduced in the Senate by members of the Senate Committee on Commerce, Science, and Transportation.

5.2.3 DMS systems used by OEMs

The most renowned auto companies have shown interest in the research on DMS systems, including Toyota, Lexus, Ford, Volvo, Mitsubishi, BMW, Mercedes-Benz, Jaguar-Land Rover, Saab, Tesla and others. This section presents the first efforts in the industry to develop and integrate DMS.

The first driver monitoring system presented by Toyota was in 2006 for the latest Lexus models [207]. The system was able to determine the direction of the driver's face through a front-facing camera with near-IR illumination. It monitors the position and angle of the driver's head while the vehicle is in motion. If the Advanced pre-crash Safety system detects an obstacle ahead, and at the same time, the DMS establishes that the driver's head has been turned away from the road for very long, the system automatically activates pre-crash warnings. If the situation persists, the system can briefly apply the brakes to alert the driver. In 2008 [207], the Toyota Crown System went further. It can detect if drivers become sleepy by monitoring their eyelids. Toyota's solution combines driver face orientation and environmental obstacle detection to determine accident potential and utilizes eyelid activity to identify drowsiness. The company emphasized the importance of such a system, it claimed that critical safety warnings can be issued to the driver earlier than it would be possible without such a system. Figure 5.1 depicts the general operation of the driver monitoring system of Toyota, and Figure 5.2 describes the phases of their collision avoidance system which consider the driver state information.

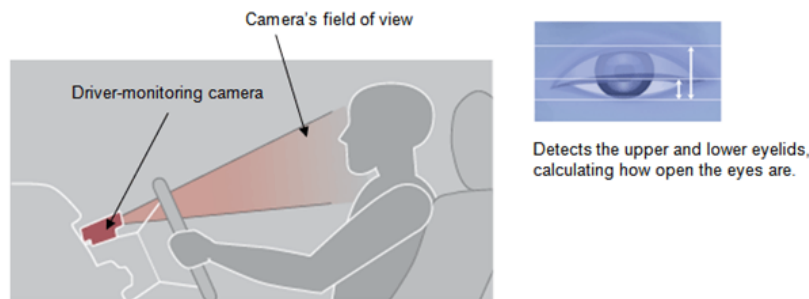


Figure 5.1: Toyota DMS analyzing face orientation and eyelid analysis [207]

Ford has collaborated with MIT's New England University Transportation Centre to understand the correlation between stressors and driving performance and identify technological advancements that both mitigate stress and create a more enjoyable experience. From this collaboration an in-vehicle stress tracker based on heart activity measurements. A later version of the system combined data from a driver i.e., heart rate, sweat data, temperature, and breathing patterns, with data from a car, including acceleration patterns, steering adjustments, and breaking [194]. The prototype consisted of multiple sensors embedded into the seatbelt and steering wheel.

Volvo and VTT Technical Research Centre of Finland have also developed a method for monitoring driver status through front-facing camera [76]. The method is based on an assessment of the driver's eye direction and behavior. Volvo also expresses the advantages of driver sensors, by monitoring eye movements, the car would be able to adjust both interior

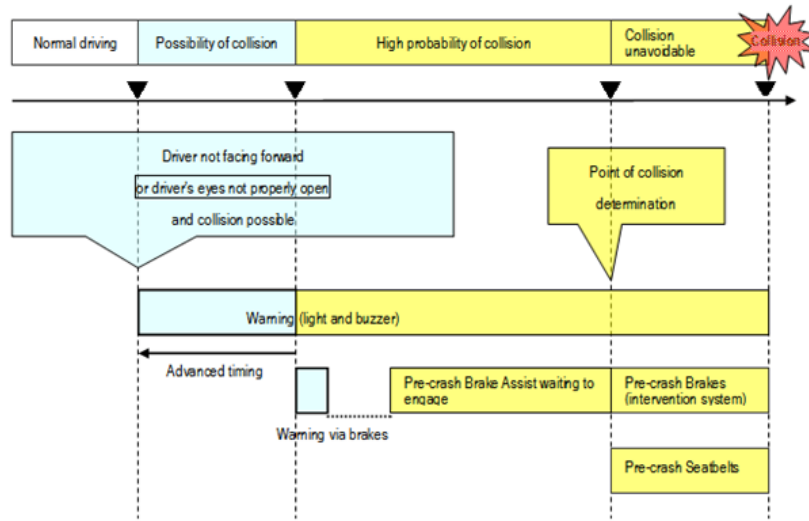


Figure 5.2: Toyota DMS sequence of system operation [207]

and exterior lighting to follow the direction in which the driver is looking [213]. The project was in the research stage from 2004 to 2008 and the system is now commercially available in the new Volvo XC90. However, the company is in continuous research to find ways to fuse different driver sensors.

Mitsubishi Electric has developed a machine-learning technology that detects cognitive distraction in drivers [141]. They state that although systems exist to detect drivers who are visually distracted due to drowsiness or inattentiveness, the detection of cognitive distractions has been difficult because symptoms sometimes appear in a driver’s behavior or biological patterns, rather than in their face or eye movements. The system combines driving performance e.g., steering wheel angle, physiological indicators such as heart rate, and head position to determine the current drive state and compare it to the “appropriate” driver state predicted by a deep learning model, as shown in Figure 5.3. If a distraction is detected, the relevant alerts are presented to the driver.

BMW has worked on DMS incorporating a wide range of sensors to acquire information from three sources: a vehicle, environment, and a driver [88]. A front-facing camera was used in their experimental setup for head position and point of gaze extraction. The driving performance metrics were also collected through Controller Area Network (CAN) i.e., acceleration and speed. The company discussed the importance of adopting machine learning e.g., Neural Networks and Support Vector Machines, for better drivers’ intention prediction. The setup was aimed to complement LDWS making it adaptable to contextual, performance, and driver state metrics. For example, LDWS is set to a higher awareness level if a driver is known to be fatigued. The company highlights the current limitations of modern ADAS and IVIS, that is the absence of driver information.

Mercedes-Benz introduced Attention Assist into its series production vehicles in 2009 [220]. Attention Assist works by first observing a driver’s behavior and then uses this information to create a unique driver profile. During operation, a series of tests continually monitor the driver input concerning this profile, and if a deviation is encountered, the system then

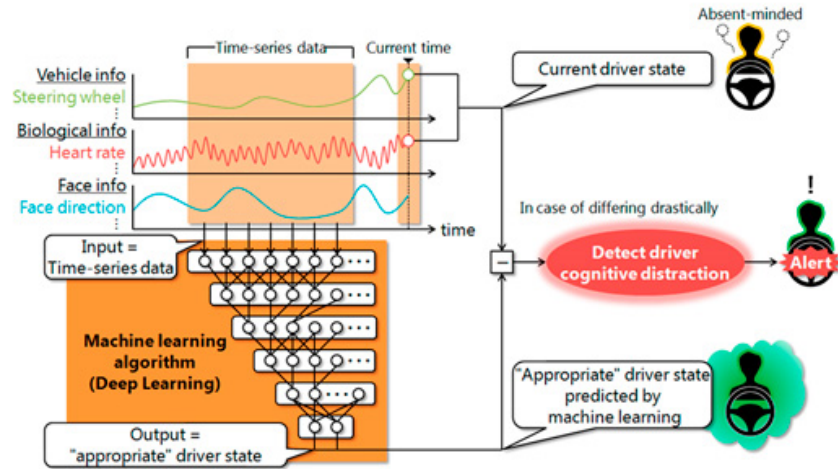


Figure 5.3: DMS proposed by Mitsubishi [141]

determines whether the deviation is a result of fatigue. If it is, Attention Assist both visually and audibly alerts the driver that it is time to take a break. The factors taken into account to determine a driver's profile include the speed, longitudinal and lateral acceleration, angle of the steering wheel, the way that the indicators and pedals are used, certain driver control actions, and even various external influences such as a side wind or an uneven road surface. The Attention Assist system only uses vehicle parameters to determine driver drowsiness.

Jaguar Land Rover is also exploring ways to measuring driver's state [104]. The company has presented its projects under the "Sixth Sense" research initiative. One of these is the "Mind Sense" project, which aims to detect the driver's concentration level by keeping track of brain activity. The system detects brainwaves through the hands via sensors embedded in the steering wheel. If the system detects fatigued driving it will initiate pedal or steering wheel vibration to bring the driver's concentration level back into optimal state. Moreover, the "Driver Wellness Monitoring" project consists of seats embedded with sensors that keep track of heart and respiration emitted vibrations and derive the driver's stress level. The company is planning to link driver state metrics with in-car ADAS to mitigate distracted and stressful driving i.e., automatically change in-car ambient lighting, audio settings, and climate control.

Saab has developed a Driver Attention Warning System [180] designed to counter the following two symptoms of driver inattention: 1) driver drowsiness and 2) distraction. The system utilizes two miniature infrared (IR) cameras: one camera installed at the base of the driver's A-pillar and the other camera at the center of the main fascia, which is focused on the driver's eyes. It also uses the SmartEye software to get eyelid, gaze, and head orientation information. In their algorithm, the driver's eye blinking frequency is measured. If a pattern of long-duration eyelid closures is detected, it indicates the potential onset of drowsiness. A three-level warning interface was designed for drowsiness detection. Distraction, on the other hand, was detected by tracking the driver's gaze. As soon gaze moves away from what is defined as the "primary attention zone"—the central part of the windshield in front of the

driver— a timer starts counting. If within 2 seconds of the timer being triggered the driver's eyes and head do not return to the "straight ahead" position, it is considered a distraction. In a case that involves peripheral tasks such as looking in the rear-view mirror, a side mirror, or turning a corner, the timer's elapse time becomes longer. Once the driver distraction has been detected, a seat vibration signal will be issued to warn the driver. However, there is no report about the robustness of this system during daytime and night-time driving under different kinds of weather conditions, providing no driver status ground truth as a reference.

The above are some examples of the first approaches OEMs have been working on to assess the driver state. Since then, OEMs have focused on improving their systems. Especially driven by the new regulations imposed in the European Union and by the EuroNCAP, the main lines of work have been on the following topics: improve the current methods to monitor the driver to be able to provide more advanced metrics for distraction, inattention and impaired drivers; extend the monitor capabilities to the whole cabin to detect child presence, unattended objects, etc.; and finally, to improve the robustness of the systems to be able to work under different and challenging conditions.

Commercial vehicles equipped with DMS

Since 2018, several new models were introduced to the market, with significant enhancements in the area of Driver Monitoring Systems as well as preparation of transitions between manual and semi-automated driving modes.

Audi A8: The 2019 Audi A8 includes a Driver Availability system, as well as a Traffic Jam Pilot (TJP) (AD Level 3) function available on dual-carriage roads, up to 60 kph. The driver monitoring camera as well as the illuminators are located above the instrument cluster, see the picture below.

In the TJP function, the Instrument Cluster provides the information to the driver whether the function is available, is activated or if a manual takeover is requested. When the TJP function is available, a white contour is displayed in the instrument cluster. When the TJP function is activated, a green contour is displayed in the instrument cluster. When the takeover is requested by the system, a message is prompted in the instrument cluster along with a red contour and a warning sound is activated.

Cadillac CT6 In the Cadillac CT6, the driver monitoring camera is located at the bottom of the instrument cluster, as shown in Figure 5.4. The driver monitoring system allows the driver to remove the hands from the steering wheel when the Super Cruise (AD Level 2) function is activated. When the driver is in manual control of the car, the LED stripe is illuminated in blue. When the Super Cruise is activated, the top of the steering wheel rim is illuminated with a green LED stripe. When a takeover is requested by the system, the LED stripe on the steering wheel is illuminated in red.

DS 7 The Driver Attention Monitoring on the 2019 MY DS7 performs an analysis of eye-opening and the analysis of gaze direction. The system uses an infrared camera, fitted on the upper part of the steering column (shown in Figure 5.5), to detect the driver's face



Figure 5.4: Cadillac CT6 dashboard, DMS and HMI

and eyes. The system can detect the driver's face and eyes and determine whether the driver is looking at the road ahead or not, to determine the level of distraction of the driver. In addition, the system also assesses the closing of the driver's eyelids to estimate the level of alertness. However, the manufacturer cautions that the system may not function properly if the camera is occluded, the driver is wearing a certain type of spectacles or sunglasses or driving in unusual or illegal positions.

Subaru Forester The DMS in the Subaru Forester is used to recognize the faces of different drivers and automatically set predetermined preferences such as seating position, door-mirror angles, and air-conditioning settings. It also uses facial recognition technology to alert the driver of fatigue or distraction conditions. The used camera is located on the upper part in the center of the dashboard (shown in Figure 5.6). This location is convenient to manage Left Hand Drive and Right Hand Drive configurations and to prevent occlusions from the steering wheel. A drawback of this configuration may exist with a higher incidence angle between the driver monitoring camera and the driver's face.

Mercedes S-Class Starting in 2023, Mercedes will introduce their first SAE-L3 vehicles. As part of their solution, they will incorporate Driver Assist, a system that primarily analyzes the driver's face to detect signs of disengagement. The DMS will be utilized to monitor the driver's level of attention and assess their readiness to regain control in situations that fall outside the ODD.

Tesla Vehicles Since 2021, Tesla has been gradually implementing driver monitoring features in its vehicles. The initial feature introduced was the Cabin Camera, which was integrated into the Model 3 and Model Y vehicles. Positioned above the rearview mirror, this camera is employed to monitor the driver's attention. It should be noted that the camera does

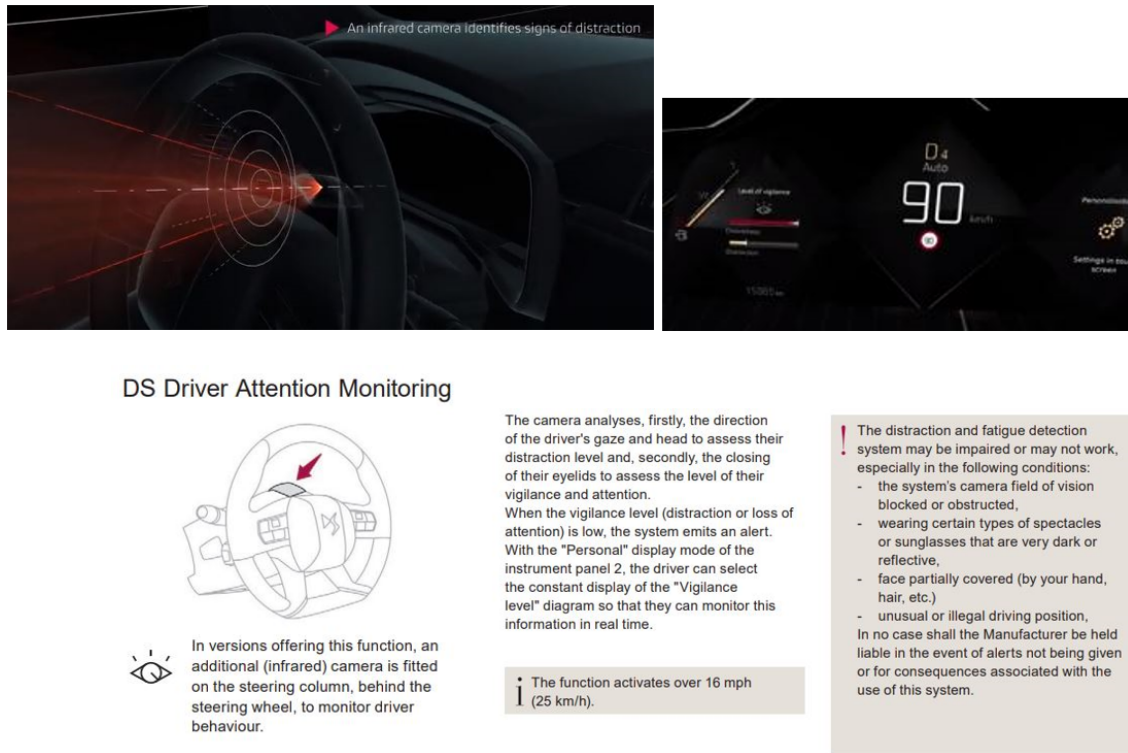


Figure 5.5: DS 7 Driver Attention Monitoring



Figure 5.6: Subaru Forester DMS integration

not serve the purpose of driver identification, but rather to detect and alert the driver in case of inattentiveness towards the road. However, there have been reports of some issues with the existing DMS in Tesla vehicles under specific conditions [30]. It is expected that Tesla will release an update in 2023 [206] to enhance driver monitoring functionalities, expanding the capabilities to detect driver distraction and drowsiness utilizing the driver monitoring camera.

The preceding paragraphs exemplify the active efforts of the automotive industry in the development and integration of Driver Monitoring Systems (DMS) into their vehicles. While some DMS functionalities are being incorporated into vehicles, there remains a need for continued development and improvement of the technology. This is crucial to enhance driving safety and effectively meet the evolving requirements set forth by legislation. The industry recognizes the ongoing need to advance DMS technology to ensure optimal driver safety and compliance with changing regulations.

5.2.4 DMS component suppliers

Commercial products and technological activities

Many companies have shown their interest in recent years to develop technological solutions for detecting and monitoring dangerous levels of driver inattention. Some of these projects are still in development, validation testing or early prototyping. For non-intrusive measurement, these devices mainly utilize video cameras and computer vision technologies. Recently, Cardio-ID startup has presented a steering wheel device to obtain electrocardiographic signals (ECG) to identify the driver's identity and health state [26]. With the advances in personal devices such as smartphones and wearable devices, some companies have started to release solutions for driver state monitoring based on these devices using their embedded sensors.

Some commercial solutions, such as iOnRoad and Augmented Driving focus on monitoring the outside of the car to detect other vehicles and lane change. Besides, the app (CarSafe) [223] also monitors driver behavior. This application simply exploits CV and ML algorithms to monitor and detect whether the driver is tired or distracted using the front-facing camera, while at the same time tracking road conditions using the rear-facing camera.

Regarding wearable devices, one of the devices that has captured more attention within the automobile industry is Google Glass. In [99], the authors presented how the information about eye blinking frequency and head motion patterns gathered from Google Glass sensors can be used to categorize and analyze different types of high-level activities of the user.

In addition to smart glasses, Fujitsu launched in 2015, the FUJITSU Vehicle ICT FEELythm [68], a wearable sensor that detects when drivers are drowsy based on their pulse. More recently, other wrist-worn wearable consumer devices have been used to monitor driver drowsiness. In [119], Empatica E4 wristband was used to obtain heart rate signals. Although the authors reported good results, the device was not designed for automotive applications, and it is not clear the user acceptance of wearing a wristband while driving.

Consumer electronics devices

Attention Technology, Inc. has designed and developed the DD850 Driver Fatigue Monitor (DFM). The DFM is a video-based drowsiness detection system that works by measuring slow eyelid closure. It is designed to mount on a vehicle's dashboard just to the right-hand side of the steering wheel and provides a continuous measurement of eye position and eyelid closure [11].

In particular, DFM estimates the percentage of eyelid closure (PERCLOS) to determine

drowsiness, which is the proportion of time that the eyes are closed 80% or more over a specified time interval. DFM uses a structured illumination approach to identify the driver's eyes. This approach obtains two consecutive images of the driver using a single camera. The first image is acquired using an IR illumination source that produces a bright-pupil image. The second image uses an IR illumination source at a different wavelength to produce an image with dark pupils. These two images are essentially identical, except for the brightness of the pupils in the images.

A third image is obtained, which is the difference between the two input images, enhancing the bright eyes and eliminating all image features, except for the bright pupils. The driver's eyes are identified in this third image by applying a threshold to the pixel brightness. The bright-pupil effect utilized by DFM is a simple and effective eye-tracking approach for pupil detection based on a differential lighting scheme. However, the success of the bright-pupil technique strongly depends on the brightness and size of the pupils, which are often functions of face orientation, external illumination interference, the distance of the subject from the camera, and race. For real-world in-vehicle applications, sunlight can interfere with IR illumination, reflections from eyeglasses can create confounding bright spots near the eyes, and sunglasses tend to disturb the IR light and make the bright-pupil phenomenon appear very weak.

Seeing Machines develops advanced computer vision systems for research on human performance measurement, advanced driver-assistance systems, and transportation [188]. Their main product, e-DMS, provides head and face tracking, as well as eye, eyelid, and gaze tracking for human subjects, using a noncontact camera sensor (illustrated in Figure 5.7)). The solutions provided aim to cover the requirements imposed by European and United States regulations by gradually incorporating more features such as occupant detection, child detection and action recognition. In addition, they offer this solution in dedicated embedded SoC, claiming faster integration times. While the device has demonstrated good performance in a simulated environment, it faces several challenges when operating in real-world driving conditions, which limits its robustness. Consequently, the company has acknowledged the need for additional data that can capture edge cases. Recently, they announced their intention to enhance the robustness of their algorithms by incorporating synthetic data into their training dataset [44].

Smart Eye AB is another company that provides computer vision-based software that detects human face/head movement, eye movement, and gaze direction. Their main product, the Smart Eye Pro, is a machine vision system that estimates head pose using a simple and robust method based on tracking individual facial features and a 3D head model. Although the face is tracked, the gaze direction and eyelid positions are determined by combining image edge information with 3D models of the eye and eyelids. One major advantage is that eye and head tracking can continue, although one camera is fully occluded or otherwise non-operational. This approach also allows for large head motions (translation and rotation). Smart Eye offers hardware-agnostic solutions specially designed for automotive in-cabin real-time measurements of the driver's head pose, gaze direction and eyelid closure. A specialized algorithm for the detection of driver fatigue and attention is not provided by this company [197].



Figure 5.7: Image analysis of Seeing Machines DMS [188]

Affectiva is a spin-off from the MIT Media Lab that focuses on estimating the emotional state of drivers using a combination of camera-based driver monitoring and sound interpretation. The company's developed system uses cameras and microphones to monitor the status of both the driver and passengers (as shown in Figure 5.8). Affectiva refers to its emotion recognition and detection technology as "artificial emotional" intelligence or Emotion AI [221]. In 2021, Smart Eye acquired the company with the intention of integrating Affectiva's emotion detection technology into Smart Eye's Interior Sensing solutions, to gain deeper insights into the emotional state of vehicle occupants [2].

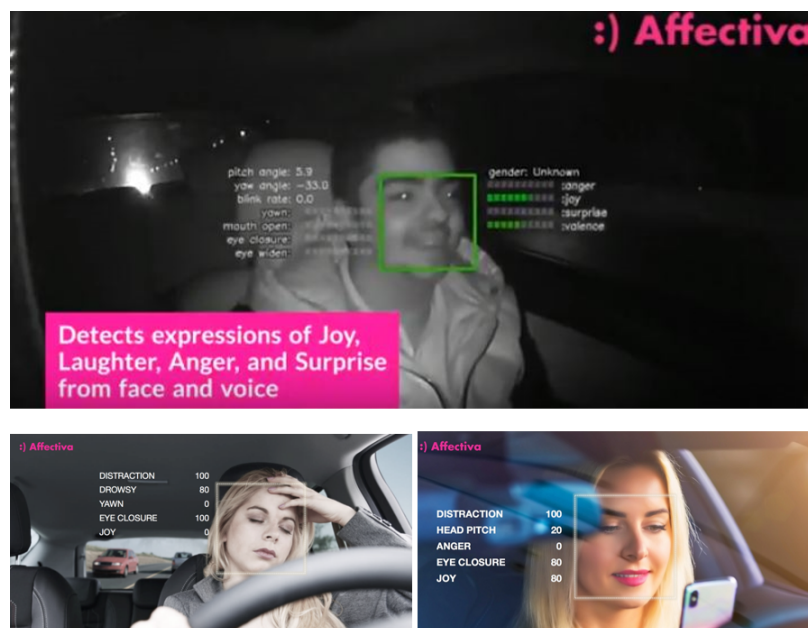


Figure 5.8: Affectiva DMS detection of emotional state

5.3 DMS methods classification

DMS comes in various forms and serve different purposes, each requiring a varying degree of technological complexity. As CV and DL technologies continue to evolve, their application has also had an impact on the development of DMS. To better categorize DMS methods based on the underlying technology’s complexity and specificity, it is suggested to create a general platform that integrates real-time DMS.

Therefore, this work proposes to classify methods included in modern DMS into three main categories, according to their level of application specificity (see Table 5.2):

Low-level analysis	Physiological analysis	Driver status analysis
Face detection	Eye gaze estimation	Distraction classification
Face landmark fitting	Eyelid Aperture Estimation	Drowsiness classification
Eye tracking	Blink detection	Emotional state classification
Head pose estimation	Blink analysis	Facial expression classification
	Face recognition	

Table 5.2: DMS analysis categories and examples

1. **Low-level image analysis:** This is the foundational level of driver monitoring, where algorithms process input from sensors such as cameras or depth sensors to generate measurements from images or depth maps of the driver’s face and upper body. These algorithms have broad applications and mature development, with many stable implementations available in open-source libraries. This makes it easy to incorporate and swap out methods for evaluating accuracy within complex pipelines. Examples of these methods are: object detection [174], 2D landmark model fitting [111], [225], object tracking [147], body pose estimation [81], head pose estimation [18].
 - **Input:** images captured with installed sensors (cameras) inside the car: standard images or depth maps (only for stereo or 3D cameras).
 - **Output:** low-level features or measurements, directly observable from the images, such as the face position (bounding box), face alignment (2D shape model), eyes position (bounding box) and alignment (2D shape model), etc.
2. **Physiological analysis:** These modules generate mid-level features that reflect specific physiological metrics from the driver, which are then used to classify the driver’s condition. This category includes all methods that extract visual physiological measures from the human face and body. Some examples include facial movements detection[114], eye gaze estimation[77], [146], eyelid aperture estimation[164], blink detection[9], yawning detection[159], etc. While some of these methods can be found in open-source and commercial libraries, they may not be as mature as low-level features. Generally, these measurements can be generated using images and low-level features.

- **Input:** images and low-level features.
 - **Output:** specific measurements related to physiological status: blinking patterns, gaze estimation, head pose analysis and any other feature that can be correlated to distraction or fatigue levels.
3. **Driver status analysis:** These modules perform higher-level analysis using previous measurements to generate driver status descriptors. This category includes all algorithms and methods specific to driver state monitoring, such as fatigue detection [178], distraction detection [118], and emotional state estimation [125]. The methods in this category generally use machine learning techniques to learn how to classify high-level measures of the driver's state from lower-level and physiological features.
- **Input:** images and low and mid-level features from the images: faces, eye, gaze estimation, blink patterns.
 - **Output:** qualitative and/or quantitative measurements of distraction and fatigue levels of the driver (and any other descriptors that might be correlated with the driving capabilities).

5.4 DMS software framework

Building DMS involves the integration of several technologies including CV, DL or signal processing. The design process of a full DMS should not be affected by integration barriers produced by incompatible interfaces and data management strategies. The definition of a software framework that defines the interfaces between components but at the same time allows flexibility to implement different types of methods will enable better and faster development of DMS.

A modular approach is proposed for the rapid prototyping of DMS by defining atomic processing units and the interface between these units. The processing algorithm is encapsulated in an abstract processing unit named *Analyzer*, which is responsible for performing a specific task from any of the levels as defined in Table 5.2. The process can be as complex as necessary, but for the purpose of modularity, each *Analyzer* should perform the minimal process that solves the problem it was designed for. By breaking down a DMS system into pieces, a software integrator can achieve a modular architecture where an *Analyzer* can be replaced by a new version of the algorithm at any point during development.

The processing methods (i.e. *Analyzers*), generate sets of results that are grouped into entities called *Measures*. These results may vary in nature or type and can be used as input for downstream methods in the pipeline. Some examples of *Measures* are included in Table 5.3.

An *Analyzer* communicates with other components by accepting a specific set of interface objects (refer to Figure 5.9(a)). The available elements in the DMS framework include (i) sensor data (such as camera images), (ii) measures from other *Analyzers*, and (iii) parameters and models loaded from databases. By linking the basic processing units in chains of *Analyzers*, complex pipelines can be constructed that generate the desired measurement objects, as shown in Figure 5.9(b).

Measure	Type	Measure	Type
face bounding box	rectangle	body pose	point list
face 2D landmarks	points list	driver action label	integer and string
eye aperture level	float value (0-1)	fatigue level	integer and string
eye gaze vector	3D vector	distraction level	integer and string
eye blink metric	float value	emotion label	integer and string
head pose	3 float values	face identity	integer value

Table 5.3: Examples of *Measure* elements in DMS software

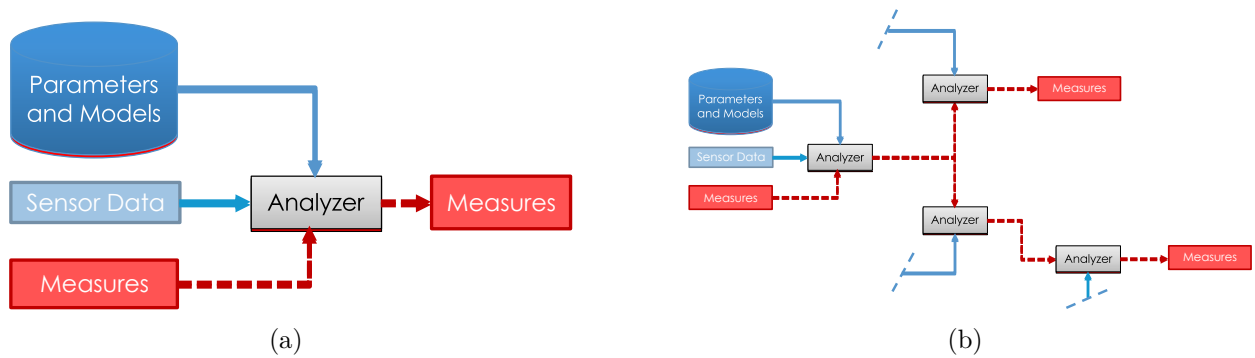


Figure 5.9: Proposed framework for DMS pipeline generation: (a) Atomic processing unit *Analyzer* with its interfaces (b) Pipeline construction from basic units

The *Measures* entities in the proposed software framework serve as input/output interfaces that enable data flow in the pipeline. Furthermore, the framework facilitates the easy replacement of algorithm implementations for testing and validation purposes, for *Analyzers* that share the same *Measures* interface. Additionally, a single *Measures* object can be shared among multiple *Analyzers*, reducing the need to perform the same task repeatedly. This is often the case for low-level features that are commonly used by several complex algorithms. For instance, a face detection *Analyzer* can be reused by algorithms such as face identification, eye blink analysis, and head pose estimation.

Parameters and *Models* objects are input information provided to each of the processing *Analyzer* and consist of a set of values and files required by the algorithm to correctly perform the designed task. These parameters are loaded at the beginning of the pipeline execution to reduce loading overhead during runtime. A characteristic of input *Parameters* and *Models* is that usually, they are unique to each *Analyzer* implementation. Therefore, their definition and implementation are done by the designer of the processing algorithm. Each designer will have the tools to build a container for this data (see Section 5.5).

The third interface component of the framework is the *Sensor Data* which contains the source information that will be processed by the *Analyzer*. The most common type of sensor data used in CV and DL algorithms are images taken from a video camera. Most DMS solutions

require images taken from a camera capturing the face of the driver. However, as DMS technology evolve new algorithms may need to introduce other sources of data such as body cameras to analyze body actions, fish-eye cabin cameras to sense the interaction with other occupants or depth cameras to improve accuracy. Therefore, each algorithm can have multiple input sensor data types. The proposed DMS framework is flexible to allow different types and number of input sensor data.

5.5 Implementation details

The proposed software framework is implemented as a C++ library, consisting of a set of modules that define abstract classes and child instances that implement algorithms (refer to Section 5.3) based on proprietary and third-party libraries. The choice of C++ as the development language is due to its widespread use in embedded systems and real-time applications. C++ is considered a more secure language than C, thanks to its use of string literals, enumeration constants, templates, and other features. Additionally, the overloaded functions and constructors in C++ are an asset for embedded systems programming. The object-oriented nature of C++ is also advantageous for developing complex embedded systems.

The ability to use *polymorphism* in C++ allows for the definition of base classes and derived child classes, making it possible to redefine methods for the derived classes. This object-oriented programming technique was applied to all defined elements in the proposed DMS software framework to enable modularity and scalability of the software library. Figure 5.10 illustrates the components of the proposed DMS software framework and their interrelationships. The figure shows the abstract classes that form the foundation of the library, along with specific instances of processing algorithms and generated data.

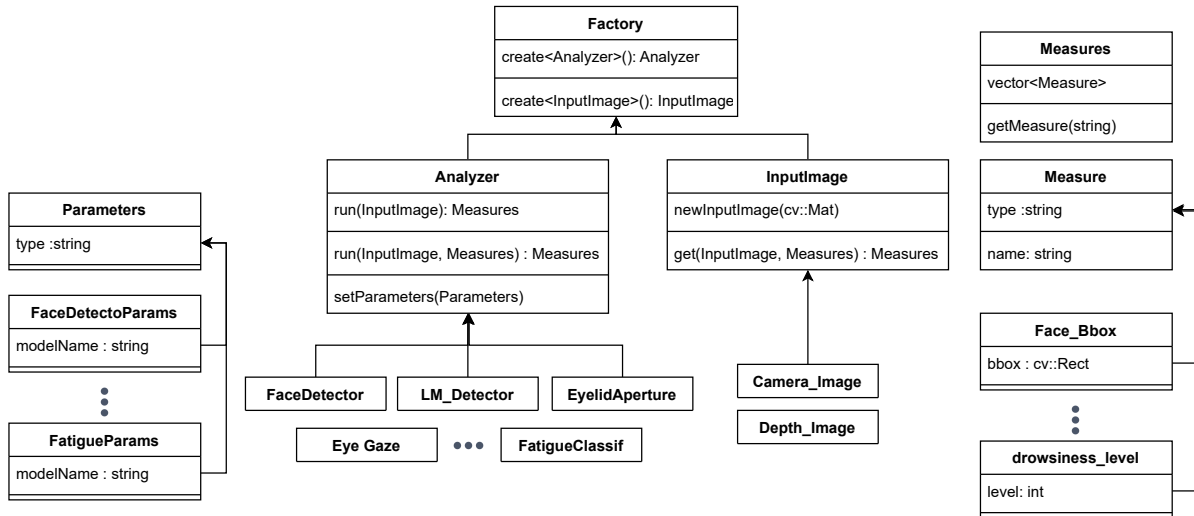


Figure 5.10: Interrelations between the components of the proposed DMS software framework based on a *Factory* design pattern. The arrows depict the inheritance relationship between classes.

5.5.1 Analyzers

As *Analyzers* are the main processing units of the DMS software framework, an abstract class is defined to establish the execution and configuration interface for all the implemented algorithms. In this class, the run methods and set parameters methods are defined. A set of derived classes can then be defined. Each child class will have the implementation of one DMS task, which could correspond to a low-level, physiological or driver status type of algorithm.

To access an *Analyzer* object *a*, it is proposed to use a Factory pattern design. By defining an interface class *Factory* the user of the library can easily and securely access different implementations of the algorithms. For instance, in the proposed approach it is possible to integrate third-party implementation methods such as the ones available in OpenCV detectors (i.e. face and eye detection), Dlib [111] (i.e. face alignment), Viulib ² (i.e. face recognition, eyelid aperture estimation [164], 3D face tracking [77]), OpenFace [10] (i.e. face detection, face alignment) and Intel RealSense SDK [96] (which offers a variety of face analysis measurements).

5.5.2 Measures

For *Measures* instances, two top-level classes are defined. The first class is called *Measures* and is designed as a container of individual values and results from the processing algorithms. These values are stored in the second type of class, the *Measure* class. This class is an abstract class from which a diverse set of derived child classes can be defined. For the implementation of this software framework, the following *Measure* types are defined: face bounding box, eye bounding box, eye corners, face 2D landmarks, face 3D landmarks, eye contour, eyelid aperture, eye gaze vector, eyes point of gaze 3D, head pose, driver action, driver fatigued state. However, as new algorithms are incorporated into the framework, new types of *Measures* classes can be added to the library.

5.5.3 Parameters and Models

The defined method to pass parameters and models to the *Analyzer* objects was solved by defining an abstract class called *Parameters*. Each *Analyzer* designer will define its required *Parameters* class for their method. Some examples of the possible data a *Parameters* class can implement are numerical parameters, on/off flags and string paths to external model files. Then, the *Parameters* class can be passed to the *Analyzer* instance thanks to the abstract class interfacing between classes.

5.5.4 Sensor Data

The input sensor data such as images and depth maps can be encapsulated in the *InputImage* class. This class allows the definition of other two types of derived classes: *CameraImage* and *DepthImage*. Inside these classes commonly used third-party libraries such as OpenCV can be used to store the actual content of the images. The advantage of using an image container

²<https://www.viulib.org>

is that pre-processing steps can be done at the beginning of the pipeline. Therefore, reduce the unnecessary redundant time spent in image handling. For instance, two *Analyzer* objects in the pipeline may require an input image with different resolution or color channels than the input camera configuration. Then, the *InputImage* class will convert the image to the desired resolution and color channels only once and store the result for future usage by other *Analyzer* objects.

5.5.5 RTMaps integration

The proposed DMS framework has been integrated and deployed using the RTMaps middleware [97]. RTMaps is an asynchronous, high-performance middleware that allows the definition of a platform for acquiring, recording, storing, and replaying data from multiple sensors and processing algorithms. With RTMaps, complex, diagram-based pipelines for operating intelligent vehicles and robotic systems can be constructed using a graphical user interface. The advantage of this approach is twofold: first, it enables users to quickly create applications using drag-and-drop techniques to connect components; second, it helps development teams to focus on defining clear boundaries and interfaces, which is crucial for large projects involving autonomous and intelligent vehicles. Clear interface definitions are necessary due to the number of participants involved. The interface for components includes inputs, outputs, and properties. Once a task's interface is defined, replacing one algorithm with another is straightforward. A component can be replaced by another, and the work is done. The proposed DMS framework follows this approach, with clearly defined interfaces, and the C++ library makes it easy to build RTMaps components that can be integrated into complex RTMaps diagrams.

Each *Analyzer* is encapsulated as an RTMaps component, where instances of the *Measures* class are used as the interface between components. This approach enables DMS developers to construct graphical pipelines for various DMS functionalities, as shown in Figure 5.11.

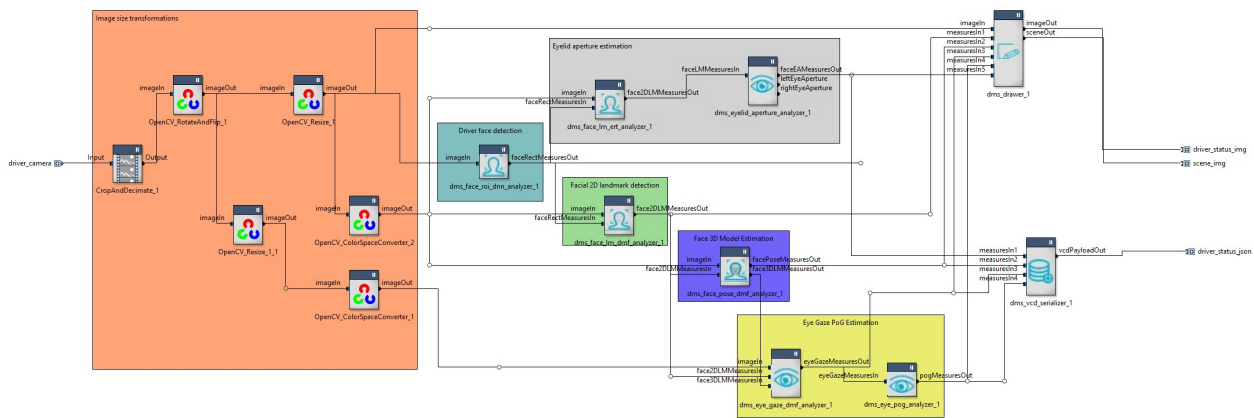


Figure 5.11: DMS pipeline example as a combination of RTMaps components. Each *Analyzer* can produce and consume *Measures* objects.

Furthermore, the connectivity of DMS RTMaps components extends beyond their standalone functionality, as they can be seamlessly integrated with other intelligent vehicle tasks such as external sensing, localization modules, situation awareness, and actuators. Facilitating communication between these components, the *DMSLib* library has implemented a parsing function that enables the exportation of results from the *Analyzers* to the *Measures* class in the *Video Content Description* (VCD) metadata structuring format [149]. VCD has gained significant traction due to its status as the first open-source metadata structure, accompanied by a comprehensive set of tools. This format demonstrates its versatility in accurately representing complex scenes by offering flexibility in labeling 2D and 3D objects, pixel-wise labels, actions, events, context, semantic relations, odometry, and calibration.

VCD follows the standardization labeling format procedure of the OpenLABEL project [6] carried out by the ASAM consortium. The OpenLABEL format aims to define a standardized format for describing labeled data between different software tools and hardware devices. The OpenLABEL format uses the JSON format as the underlying text format. The open-source nature of the VCD format allows developers of ADAS and Intelligent Vehicles methods to access and analyze the output of DMS pipelines, thus providing a possibility for enhancing their analyses.

In addition, the DMS components can be configured in RTMaps to do its process in *separate CPU threads*. Therefore, a complex pipeline can benefit from multi-core hardware setups to improve its performance. RTMaps has built-in configurations which allow enabling or disabling running a component in separate threads.

5.5.6 Data synchronization

The DMS components make use of RTMaps *circular buffer* communication model to perform data synchronization. A circular buffer is a FIFO data structure that considers memory to be managed circularly; that is, the read/write indices loop back to 0 after it reaches the buffer length [152]. This is achieved by two pointers to the array, the *head* and the *tail* pointer. It has a fixed size allocated once at the system run-time. It uses the tail and head pointers to indicate where to read or write, respectively, input and output data (see Figure 5.12).

Accordingly, each time a new data sample is added (write) into the buffer, the head pointer is incremented and likewise, when the data is being read the tail pointer is incremented. At the beginning tail and head are both at index 0. The modulo operation is performed to reset the head or tail to zero, every time the maximum index is reached.

Circular buffers are frequently used to solve the single produce-consumer problem. That is, one thread of execution is responsible for data production and another for consumption. In the very low to medium embedded devices, the producer is often an Interrupt Service Routine (ISR)(e.g. data produced from sensors) and the consumer is the main processing unit [153].

The utilization of circular buffers in the DMS framework offers many benefits, such as:

- A high degree of communication predictability: Since the memory address and the size of the buffer structure never change, it is always known for producer and consumer tasks where to write/read with the help of tail and head pointers.

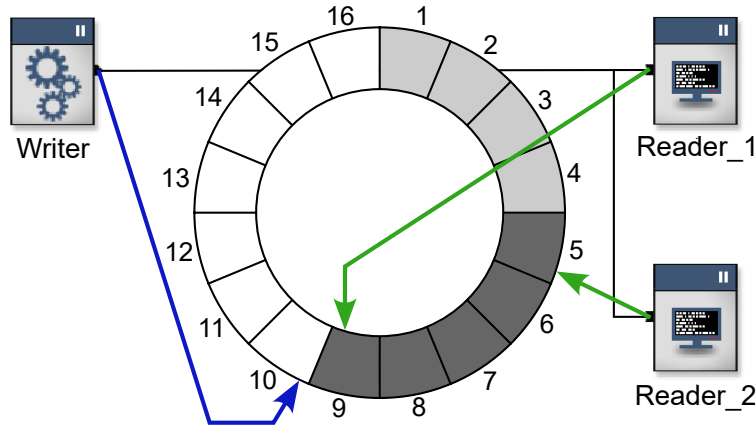


Figure 5.12: Representation of RTMaps circular buffer for writing and reading data of components

- No dynamic memory allocation: Making local copies during the execution is a resource-consuming operation that increases the degree of uncertainty regarding the data management in real-time.
- The circular buffer is easy to implement.

In a functional DMS pipeline, each buffer is specifically allocated to store data samples associated with a unique data variable (*Measures*). These buffers can be shared among multiple *Analyzers*. While multiple consumer *Analyzers* can simultaneously read from a shared buffer, only one producer can write to it. It is important to note that when a producer is writing data into a particular buffer slot, other tasks cannot access that slot to avoid introducing inconsistencies in the data. Consequently, a consumer *Analyzers* retrieves data from the buffers where its consecutive predecessors (producers) write their output.

The size of the circular buffer and the speed at which components write and read the data significantly impact the lag and synchronization necessary for maintaining consistent data. In RTMaps, a DMS integrator has the flexibility to adjust the producer buffer size to prevent data overflows during pipeline run-time, ensuring smooth operation.

The integration of the structuring framework for DMS with RTMaps has enabled the testing of multiple use cases across diverse platforms, including regular CPUs and specific automotive embedded systems such as Drive PX2. This integration has resulted in rapid prototyping times and ensured real-time performance.

5.6 Conclusions

This chapter has presented a methodology for organizing DMS algorithms by employing a framework that utilizes classes with shared interfaces to encapsulate processing tasks. It was proposed to use abstract classes to define each element of the software framework. The proposed framework was integrated into automotive-level middleware RTMaps to take advantage of the multi-threading and circular buffer functionalities.

In recent years, there has been a noticeable increase in interest from both research institutions

and the industry in advancing the development of DMS. This interest has particularly intensified following the introduction of new regulations and updates in vehicle testing rating protocols. As a result, OEMs and Tier 1 stakeholders are required to quickly incorporate DMS functionalities into their vehicles while maintaining software quality.

Currently, the solutions offered by suppliers are isolated black boxes in relation to the rest of the vehicle's functionalities. However, nowadays, OEMs require DMSs to be seamlessly integrated with the advanced features of the vehicle. Therefore, it is essential to design DMS solutions based on a modular and interfaceable software architecture. Consequently, there is a pressing need for a framework that enables rapid prototyping and seamless integration of various DMS algorithms and methods. This thesis proposes a novel framework that aims to facilitate this task with the modular integration of different DMS algorithms and methods, thus paving the way for future advancements in this field. Notably, this is the first work that openly proposes a modular framework for implementing DMS architectures.

One of the main advantages of this approach is that it allows for the reuse of common processing tasks across multiple DMS functionalities, such as drowsiness detection or distraction detection. This eliminates the need for duplicated computational resources. The modular design of the proposed software framework significantly reduces the time required to bring different driver monitoring solutions to market while minimizing development costs.

In future work, one area of focus will be optimizing the circular buffer size to effectively utilize hardware memory resources, especially when running multiple processing units simultaneously. By fine-tuning the buffer size, we can ensure better availability of data consistency and prevent data loss. This optimization will contribute to enhancing overall system performance and reliability.

After proposing the software framework that facilitates the definition of DMS architectures, the subsequent step in the pipeline involves applying various CV and DL methods. These methods are utilized to construct software components capable of integration into real vehicles, enabling the monitoring of the driver's state. Furthermore, these methods will greatly benefit from the availability of training and validation data obtained from DMD.

Chapter 6

Driver monitoring applications

6.1 Introduction

Traditional DMS methods utilize vehicular inputs (e.g., steering wheel angle, pedal action, lane deviation) [20] or biological signals (e.g. heart, brain, skin signals) [19] from driver-attached devices to assess fatigue and distraction. However, in highly autonomous vehicles, these features may become less available or impractical due to limited driver interaction or the expense of intrusive sensors, posing challenges for continuous monitoring in real-world applications.

A DMS based on visual appearance needs to cover potential applications such as Fatigue Analysis, Distraction Detection, Behavior Analysis or Action Recognition. These high-level functions can only be provided by building a chain of low-level and mid-level modules. Low-level functions take images as input and detect elemental features such as the face or the eyes, while mid-level functions analyze specific features of the detected elements, such as gaze, eyelid aperture, gestures, etc.

This thesis proposes the construction of DMS applications as networks of *Analyzers* (see Chapter 5) which can be easily interconnected in pipelines that produce highly-specific outputs, as shown in Figure 6.1. For instance, a fatigue analysis function of a DMS usually would imply the usage of steps like face detection, eye localization, eye tracking, eyelid aperture measurement, and finally drowsiness classification. An additional use case could be defined for distraction analysis and would imply the detection of face, eye localization, eye gaze estimation and computation of eyes-off-the-road time.

The proposed DMS framework facilitates the creation of flexible pipelines tailored to application-specific use cases. Each module can be implemented using third-party libraries or proprietary novel methods and connected to others through the defined interfaces (i.e., *Measures*, *Sensor Data*, and *Parameters*). These interfaces are based on the expected output of each processing *Analyzer* defined in the DMS architecture.

As analyzed in Chapter 2, drivers exhibit certain observable behaviors such as eyelid and head movements that correlate significantly with distraction and fatigue. Besides, the advances in

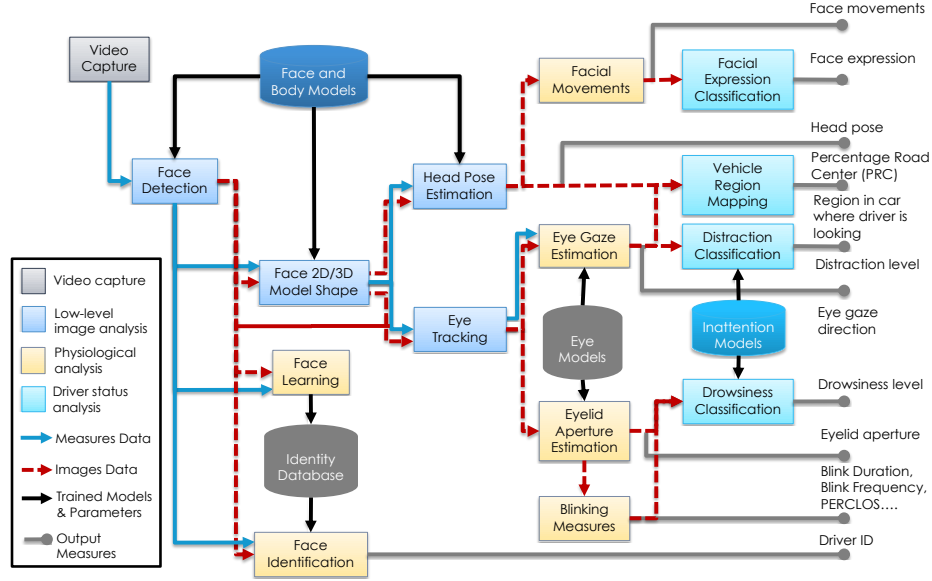


Figure 6.1: Example of DMS architecture build based on proposed framework

CV research have made it possible to robustly extract observable features from the driver's face with unobtrusive sensors [193]. As the scope of this thesis is to provide novel mechanisms, data and methodologies to develop DMS applications, this chapter describes the implementation of these tools in modern applications of DMS. There are three DMS use cases considered in this study: (i) eyelid aperture estimation (Figure 6.2), (ii) eye gaze estimation (Figure 6.3) and (iii) action recognition (Figure 6.4).

For the case of eyelid aperture estimation, the presented pipeline integrates a novel method to compute the normalized eyelid aperture level of the driver. The method relies on modeling the personal visual characteristics of each driver by online estimating the nominal maximum aperture level which then allows the normalization of the eyelid aperture level. The details of the proposed method are described in Chapter 3.

The other use cases described in this chapter (i.e. eye gaze estimation and action recognition) address the work carried out in this thesis to adapt and integrate vision-based components developed by research colleagues at Vicomtech to exemplify the integration of third-party components into the proposed DMS framework. These methods also benefit from the availability of data generated by the DMD dataset, which can be used to build the required models.

The presented DMS use cases share components, simplifying integration, implemented via various techniques. Face detection employs OpenCV[211] cascades and DNN models [224], while landmark detection utilizes DLib's ERT algorithm [111] and 3D CLNF face tracking [77]. For these low-level features, the circular buffer (see Figure 5.12) size was defined depending on the complexity of the algorithm being 12 for low complexity and 16 for those algorithms that require more computation resources such as the ones performing DL inference. The eyelid detection and gaze estimation use cases were validated in a real testing car and a driving simulator by guaranteeing the correct real-time processing of the complete pipeline.

6.2 Eyelid aperture estimation

Blink detection plays a vital role in driver monitoring systems as it enables the assessment of the driver's drowsiness and alertness levels. By monitoring the frequency and duration of eye blinks, the system can detect signs of drowsiness or fatigue and subsequently alert the driver. To accurately infer the driver's drowsiness state, it is essential to perform a robust estimation of the eyelid aperture level. This estimation serves as a critical component in the blink detection process.

One practical application of eyelid aperture level estimation is the implementation of drowsiness metrics, such as the **Percentage of Eyelid Closure over Time (PERCLOS)** method [117]. PERCLOS measures the percentage of time the driver's eyelids remain closed and is widely used in various fatigue detection studies [15], [27], [183]. This metric ensures that the driver's eye closure rate remains within a healthy range, providing a valuable indicator of fatigue levels (see Section 2.3.2).

For the purpose of this thesis, an eyelid aperture estimation pipeline was integrated as a DMS application using the proposed features of the DMSLib. The pipeline consists of the following components: video capture, face detection, facial 2D landmarks extraction, eyelid amplitude extraction, online estimation of open and closed amplitude levels, and eyelid aperture estimation (see Figure 6.2).

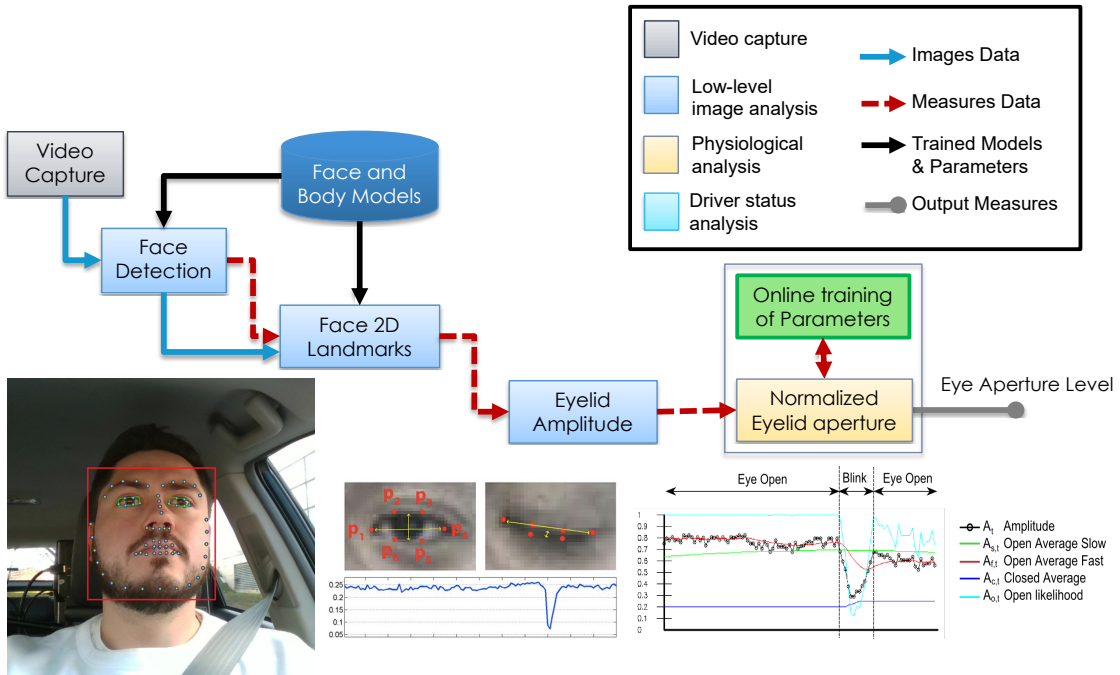


Figure 6.2: Eyelid aperture estimation pipeline

The pipeline components are implemented as DMSLib's Analyzer objects. In the following paragraphs, each component is described in detail, including the input and output measures, the sensor data required, and the parameters used. The variable types are specified within parentheses.

Video Capture The input to the pipeline is a video stream from a camera mounted inside the vehicle. The video stream is captured at 30 FPS and is packed as a DMSLib Image stream. The video stream is passed to the next component in the pipeline, which is the face detection component. Table 6.1 shows the interfaces of this component.

Video Capture				
	Name	Type	Default	Description
Parameters	Camera ID	int	0	The ID of the camera to be used for capturing the video stream
	Resolution	[int, int]	1280 × 720 pixels	The resolution of the video stream, the maximum supported resolution is used
	FPS	int	30	The frame rate of the video stream
	Color	string	RGB	The color space of the video stream
Inputs	-	-	-	-
Outputs	Image	DMSLib Image	-	The video stream captured by the camera

Table 6.1: Parameters and Interfaces of the *Video Capture* component

Face detection The first step of the pipeline is to detect the driver’s face in the input image. This task is performed using the MediaPipe implementation of a CenterNet convolutional network technique for face detector [42]. The algorithm is trained to detect frontal faces and is robust to variations in illumination and pose. Other implementations using DNN may also be used to perform this task. The output of this component is a bounding box that encloses the driver’s face. Table 6.2 shows the interfaces of the face detection component.

Face 2D Landmarks The next step in the pipeline is to extract the 2D facial landmarks from the detected face. This task is performed using the MediaPipe implementation of a 468-point 2D facial landmark detector based on MobileNetV2 [185] network architecture with customized blocks for real-time performance. The algorithm is trained to detect 468 facial landmarks, including the eyes, eyebrows, nose, mouth, and jawline. Other implementations using different DNN architectures may also be used to perform this task. When there are computation limitations, the DLIB implementation of a 68-point model, based on an ensemble of regression trees [111] was also integrated into the framework. The output of this component is a list of 2D points that correspond to the detected facial landmarks. Table 6.3 shows the interfaces of the face 2D landmarks component.

Face Detection				
	Name	Type	Default	Description
Parameters	Model Path	string	-	Path to the trained face detection model. The options are a cascade file or a DNN
	Model Type	string	Cascade	The type of the model used for face detection. The options are Cascade and DNN
	Process Size	[int, int]	640×360 pixels	The size of the image to be processed by the face detection algorithm
	Method Arguments	<i>JSON</i>	-	A key-value list of the parameters required by the face detection algorithm
Inputs	Image	DMSLib Image	-	The input image captured by the camera
Outputs	Image	DMSLib Image	-	The input image with transformations (resolution or color)
	Face BBox	DMSLib <i>Measure</i>	-	The bounding box of the face

Table 6.2: Parameters and Interfaces of the *Face Detection* component

Eyelid Amplitude The next step in the pipeline is to compute the eyelid amplitude. To perform this task the 2D landmarks for each of the eyes are used. The eyelid amplitude is calculated by measuring the ratio between the distances, in pixels, of the longer and shorter perpendicular dimensions of the eye in the image using the detected eyelid landmarks. The methodology employed in this pipeline is detailed in Section 3.3, using equation 3.1. The output of this component is the eyelid amplitude values for each eye. Table 6.4 shows the interfaces of the eyelid amplitude component.

Normalized Eyelid Aperture The final step in the pipeline is to compute the normalized eyelid amplitude. To perform this task the open/closed likelihood is obtained and then the probability of the eyes to be open is computed by applying Bayesian theory. The methodology employed in this pipeline follows the new method proposed in Section 3.3. Table 6.5 shows the interfaces of this component.

Face 2D Landmarks				
	Name	Type	Default	Description
Parameters	Model Path	string	-	Path to the trained face 2D landmark detection model. The options are a DNN or a regression tree ensemble model.
	Model Type	string	<i>DLIB</i>	The type of trained face 2D landmark detection model. The options are <i>MediaPipe</i> and <i>DLIB</i> .
	Process Size	[int, int]	640×360 pixels	The size of the image to be processed by the 2D landmark algorithm
	Method Arguments	<i>JSON</i>	-	A key-value list of the parameters required by the 2D landmark algorithm
Inputs	Image	DMSLib Image	-	The container with input images
	Face BBox	DMSLib <i>Measure</i>	-	The bounding box of the face, used to crop the image to the face region
Outputs	Image	DMSLib Image	-	The input image with transformations (resolution or color)
	2D Points	DMSLib <i>Measure</i>	-	The array of 2D points corresponding to the detected facial landmarks.

Table 6.3: Parameters and Interfaces of the *Face 2D Landmarks* component

Eyelid Amplitude				
	Name	Type	Default	Description
Parameters	-	-	-	-
Inputs	Eye 2D Landmarks	DMSLib <i>Measure</i>	-	Two arrays containing 2D points in pixels, representing the detected facial landmarks for each eye.
Outputs	Eye Amplitude Ratio	DMSLib <i>Measure</i>	-	The computed amplitude ratio for each eye

Table 6.4: Parameters and Interfaces of the *Eyelid Amplitude* component

Normalized Eyelid Aperture				
	Name	Type	Default	Description
Parameters	Method Arguments	<i>JSON</i>	-	A key-value list of the parameters required by the method to estimate eyelid aperture value. These parameters are detailed in Section 3.4
Inputs	Eye Amplitude Ratio	DMSLib <i>Measure</i>	-	The computed amplitude ratio for each eye obtained from eyelid landmarks
Outputs	Normalized Eye Aperture	DMSLib <i>Measure</i>	-	The normalized eye aperture of each eye is a value between 0 and 1, where 0 is closed and 1 is open

Table 6.5: Parameters and Interfaces of the *Normalized Eyelid Aperture* component

6.3 Gaze estimation

Eye gaze estimation is an important aspect of driver monitoring systems. It is used to detect the driver’s attention level and to determine if the driver is looking at the road or not. The objective of incorporating a reliable gaze estimation pipeline into a DMS is to identify potential indicators of both visual and cognitive distractions under any circumstances. However, eye gaze estimation methods face various challenges due to the complexity of the human eye and the variability of the environment such as illumination, severe head pose and occlusions.

The output of a gaze estimation pipeline is generally a 3D vector for each eye. Using this vector it is possible to map where the driver is looking at. To assess the driver’s distraction level and gain valuable insights, various metrics can be employed. One of the most prominent measures is the **Percentage Road Center** (PRC) [112]. This metric is defined as the percentage of time the driver is looking at the road center. To calculate the PRC, the gaze vector is projected onto the road plane and then the percentage of time the gaze vector is calculated inside a region of interest (ROI) that represents the road center. By computing the PRC separately for each eye and subsequently averaging the results, a final PRC value is obtained. The PRC metric serves as a reliable indicator of the driver’s attention level, with higher PRC values indicating greater attentiveness to the road.

The proposed eye gaze estimation pipeline of the following components: video capture, face detection, facial 2D landmarks extraction, facial 3D model fitting, head pose estimation, eye tracking and eye gaze vector estimation (see Figure 6.3).

The DMSLib framework offers the advantage of reusing various components within the pipeline. In this specific scenario, we can reuse the video capture, face detection, and facial 2D landmarks extraction from the eyelid aperture pipeline. As a result, the interfaces and parameters remain consistent (refer to Table 6.1, Table 6.2, and Table 6.3 respectively). This feature enables quick prototyping and reduces time to market for different DMS functions.

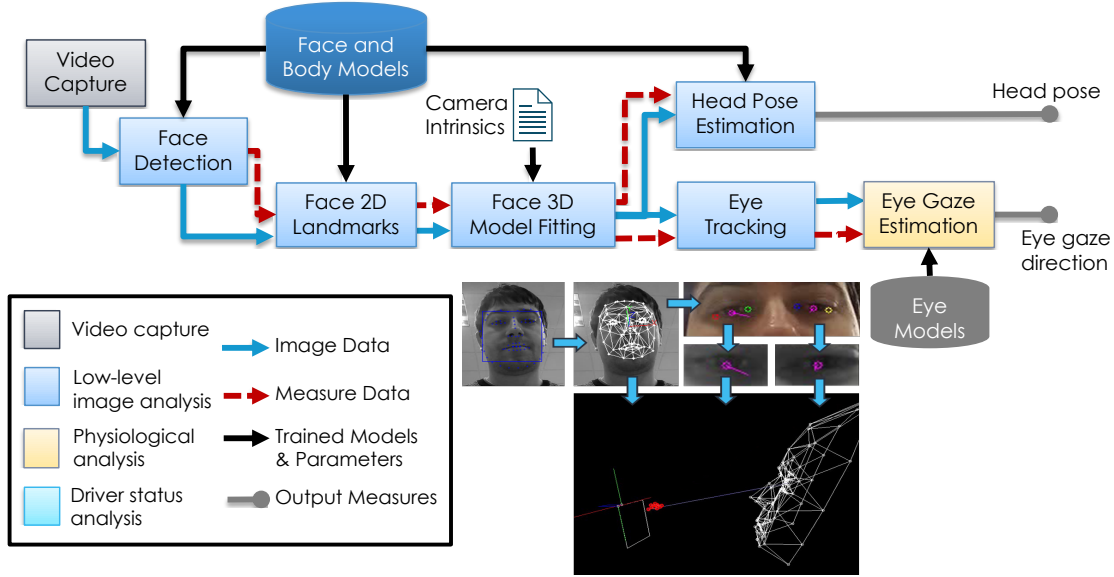


Figure 6.3: Driver gaze estimation pipeline

Face 3D Model Fitting This component performs an adjustment of a deformable 3D face model based on the 2D landmarks detected in the previous step, utilizing the method described in [77]. The 3D face model is parametric and consists of a collection of 3D vertices. These vertices are defined by a set of parameters that govern the shape and texture of the face. The fitting of the 3D face model involves minimizing the error between the detected 2D landmarks and the 3D vertices of the model. To ensure realistic human facial shapes, the distribution of 3D landmarks is constrained to prevent impossible configurations. Table 6.6 shows the interfaces of the face 3D model fitting component.

Head Pose Estimation The adjusted 3D face model provides the head pose by capturing the rotation and translation of the head with respect to the camera. It is represented by a 3D rotation vector and a 3D translation vector. The head pose is useful for distinguishing between "owl" and "lizard" gaze movements [63]. "Owl" eye gaze patterns are characterized by large head movements, while "lizard" gaze patterns involve less head movement with more significant eye movements. Table 6.7 shows the interfaces of the head pose estimation component.

Eye Tracking This component extracts the regions around the eyes from the input image and performs shape and intensity distribution normalization. Table 6.8 shows the interfaces of the eye-tracking component.

Eye Gaze Estimation Clipped eye images are inputted to a CNN-based eye gaze model for direction estimation [227]. The model outputs a 3D vector per eye. Correction is applied to mitigate the effects of significant head movements distorting eye appearance in images [77]. The user's gaze vector is determined as the weighted mean vector of both eyes, with the origin at their midpoint. Table 6.9 details the interfaces of the gaze estimation component.

Face 3D Model Fitting				
	Name	Type	Default	Description
Parameters	3D Model Parameters	<i>JSON</i>	-	A key-value relation list between the landmark and vertices.
	Camera Intrinsics	list[float]	-	Intrinsics parameters of the camera as a calibration matrix
	Process Size	[int, int]	1280×720 pixels	The size of the image to be processed
Inputs	Image	DMSLib Image	-	The container with input images
	2D Face Landmarks	DMSLib <i>Measure</i>	-	The array of 2D points corresponding to the detected facial landmarks.
Outputs	Image	DMSLib Image	-	The input image with transformations (resolution or color)
	3D Face Points	DMSLib <i>Measure</i>	-	The array of 3D points of the face model with respect to the camera

Table 6.6: Parameters and Interfaces of the *Face 3D Model Fitting* component

Head Pose Estimation				
	Name	Type	Default	Description
Parameters	Process Size	[int, int]	640×360 pixels	The size of the image to be processed by the 2D landmark algorithm
Inputs	Image	DMSLib Image	-	The container with input images
	3D Face Points	DMSLib <i>Measure</i>	-	The array of 3D points of the face model with respect to the camera
Outputs	Translation Vector	DMSLib <i>Measure</i>	-	Vector $t = \{t_x, t_y, t_z\}$, XYZ positions of the face with respect to the camera
	Rotation Vector	DMSLib <i>Measure</i>	-	Vector $r = \{r_x, r_y, r_z\}$, roll-pitch-yaw angles of the face with respect to the camera

Table 6.7: Parameters and Interfaces of the *Head Pose Estimation* component

Eye Tracking				
	Name	Type	Default	Description
Parameters	Method Arguments	<i>JSON</i>	-	A key-value list of the parameters required by the eye image normalization process
	3D Face Points	DMSLib <i>Measure</i>	-	The array of 3D points of the face model with respect to the camera
Inputs	Image	DMSLib Image	-	The container with input images
Outputs	Eye Images	DMSLib Image	-	The processed eye region images

Table 6.8: Parameters and Interfaces of the *Eye Tracking* component

Eye Gaze Estimation				
	Name	Type	Default	Description
Parameters	Model Path	string	-	Path to the trained gaze vector DNN model
	Method Arguments	<i>JSON</i>	-	A key-value list of the parameters required by the eye gaze estimation model
Inputs	Images	DMSLib Image	-	The container with input images of the eyes
Outputs	Gaze Vector	DMSLib <i>Measure</i>	-	The normalized eye gaze vector

Table 6.9: Parameters and Interfaces of the *Eye Gaze Estimation* component

6.4 Action recognition

The purpose of driver action recognition and classification is to monitor any physical distraction activities a driver might do while driving. Physical distractions may decrease driving performance and cause slow reaction time. Typical distracted behaviors include calling or texting on the phone, frequently talking to passengers, reaching behind, and multiple typical distracting tasks. For this purpose, the DMD has included a dedicated section for recording specific distraction actions.

The current methods for detecting and classifying driver actions involve analyzing the driver's complete body image to identify specific patterns of various actions. As in other computer vision tasks, DNNs have emerged as the predominant approach for performing these tasks. Notably, the contribution in this thesis of the DMD and the DMSLib software framework has significantly contributed to advancing action recognition methodologies. These advancements have resulted in the creation of diverse variants of action recognition methods, as evidenced in studies such as [162] and [25]. These works have demonstrated the effectiveness and applicability of DNN-based techniques in enhancing driver action analysis and classification.

In this section, a comprehensive pipeline for driver action recognition is proposed, which identifies the required vision-based processes to detect and classify driver actions. The objective of this section is to demonstrate how the DMSLib software framework can effectively integrate multiple image analysis tasks into a unified application. Furthermore, it serves as a reference architecture for future advancements in the field of driver action recognition.

The proposed pipeline is composed of the following components: body detection, body 3D landmarks estimation, body 3D pose estimation, object detection, and action recognition. The proposed pipeline is shown in Figure 6.4.

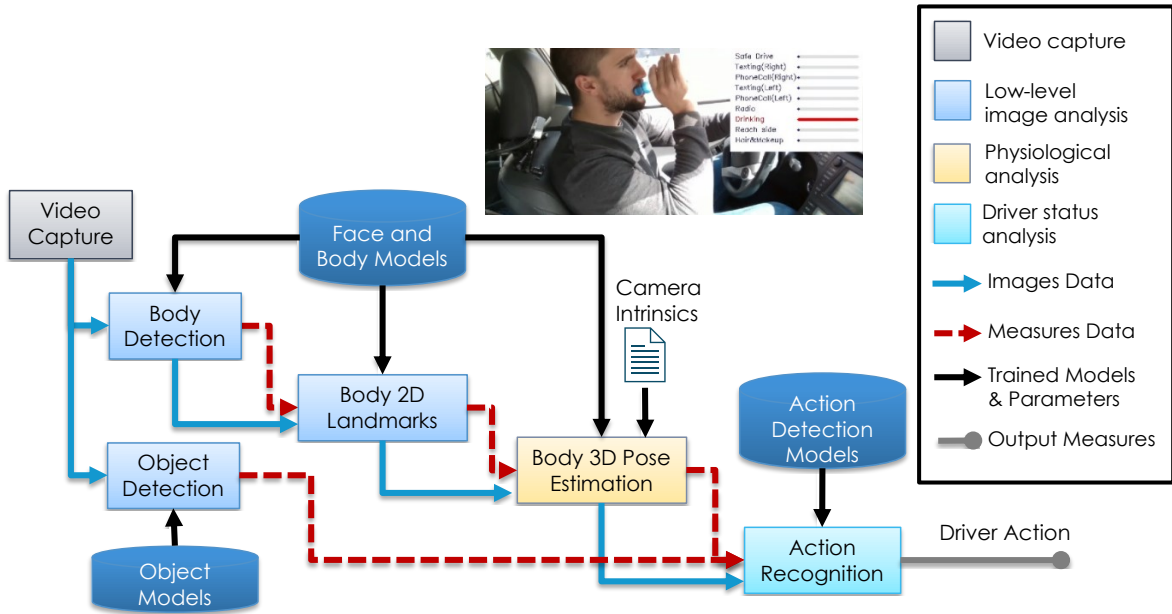


Figure 6.4: Driver action detection pipeline

Note that the Video Capture component is equivalent to the previously described pipeline. The difference is the camera source used for the action recognition pipeline is the body-facing camera. The following sections describe each component in detail.

Body Detection This component performs a person detection task to identify the driver's body in the image. The body detection component is based on the YOLO model [171] or equivalent DNN-based detection method. Table 6.10 shows the interfaces of the body detection component.

Body 2D Landmarks Similarly as with face 2D landmark detection, the body landmarks can also be computed. This task is performed using the MediaPipe implementation of a 33-point 2D body landmark detector based on BlazePose [12] network architecture. The model provides 3D coordinates, but the z-coordinate is up to scale (not metric) and obtained from synthetic data using the GHUM model (articulated 3D human shape model), fitted via an algorithm to the 2D key point projections. Then in this component, only the 2D landmarks are used. Table 6.11 shows the interfaces of the body 2D landmarks component.

Body 3D Pose Estimation This component takes the 2D body landmarks as input and estimates the corresponding 3D pose of the driver. To obtain the 3D coordinates of these landmarks, it is necessary to provide the intrinsic parameters of the camera. By using these parameters, a parametric 3D body model can be fitted to the 2D landmarks. Since the MediaPipe implementation provides the z-coordinate measured in "image pixels," the camera calibration parameters are used to convert the z-coordinate into metric units. Table 6.12 shows the interfaces of the body 3D pose estimation component.

Object Detection Driver distractions can arise from various actions involving different objects, such as cellphones, bottles, food, cigarettes, and makeup, among others. Detecting these objects can greatly aid in accurately identifying instances of driver distraction. One potential approach to implementing an object detection component is by using the YOLO implementation [171]. This model can be trained on the COCO dataset [130], which consists of 80 different classes of objects. Alternatively, the model can be trained using a custom dataset specifically tailored to detect objects relevant to the driver distraction task. Table 6.13 shows the interfaces of the object detection component.

Action Recognition The final stage of the pipeline involves detecting the action performed by the driver. This component utilizes the outputs from the preceding components, namely body pose and detected objects, to determine the action executed by the driver. To accomplish this, an action recognition component can be employed, which can be implemented using a DNN model. The method can classify individual images instantaneously based on the trained action classes or employ a temporal approach to classify a sequence of images. The temporal approach can be realized using either a 3D CNN [162] or a Recurrent Neural Network (RNN) or Long Short-Term Memory (LSTM) approaches [25]. Table 6.14 shows the interfaces of the action recognition component.

Body Detection				
	Name	Type	Default	Description
Parameters	Model Path	string	-	Path to the trained DNN body detector model
	Process Size	[int, int]	640×360 pixels	The size of the image to be processed by the body detection algorithm
	Method Arguments	<i>JSON</i>	-	A key-value list of the parameters required by the body detection model
Inputs	Image	DMSLib Image	-	The input image captured by the camera
Outputs	Image	DMSLib Image	-	The input image with transformations (resolution or color)
	Body BBox	DMSLib <i>Measure</i>	-	The bounding box of the driver's body

Table 6.10: Parameters and Interfaces of the *Body Detection* component

Body 2D Landmarks				
	Name	Type	Default	Description
Parameters	Model Path	string	-	Path to the trained body 2D landmark detection model.
	Model Type	string	<i>MediaPipe</i>	The type of trained body 2D landmark detection model. The options are <i>MediaPipe</i>
	Process Size	[int, int]	640×360 pixels	The size of the image to be processed by the 2D landmark algorithm
	Method Arguments	<i>JSON</i>	-	A key-value list of the parameters required by the 2D landmark algorithm
Inputs	Image	DMSLib Image	-	The container with input images
	Body BBox	DMSLib <i>Measure</i>	-	The bounding box of the body, used to crop the image to the body region
Outputs	Image	DMSLib Image	-	The input image with transformations (resolution or color)
	2D Points	DMSLib <i>Measure</i>	-	The array of 2D points corresponding to the detected body landmarks.

Table 6.11: Parameters and Interfaces of the *Body 2D Landmarks* component

Body 3D Pose Estimation				
	Name	Type	Default	Description
Parameters	3D Model Parameters	<i>JSON</i>	-	A key-value relation list between the landmark and vertices.
	Camera Intrinsics	list[float]	-	Intrinsics parameters of the camera as a calibration matrix
	Process Size	[int, int]	1280 × 720 pixels	The size of the image to be processed
Inputs	Image	DMSLib Image	-	The container with input images
	3D Body Landmarks	DMSLib <i>Measure</i>	-	The array of 2D points corresponding to the detected body landmarks.
Outputs	Image	DMSLib Image	-	The input image with transformations (resolution or color)
	3D Body Points	DMSLib <i>Measure</i>	-	The array of 3D points of the body model with respect to the camera

Table 6.12: Parameters and Interfaces of the *Body 3D Pose Estimation* component

Object Detection				
	Name	Type	Default	Description
Parameters	Model Path	string	-	Path to the trained DNN model
	Process Size	[int, int]	1280 × 720 pixels	The size of the image to be processed by the object detection algorithm
	Arguments	<i>JSON</i>	-	A key-value list of the parameters
Input	Image	DMSLib Image	-	The input image captured by the camera
Outputs	Image	DMSLib Image	-	The input image with transformations (resolution or color)
	Object BBox List	DMSLib <i>Measure</i>	-	The list of bounding boxes of objects with its associated class

Table 6.13: Parameters and Interfaces of the *Object Detection* component

6.5 Integration and validation in real testing setups

The proposed eyelid aperture and gaze estimation pipelines were integrated into an automation system of an instrumented vehicle and a driving simulator, in the context of the EU H2020

Action Recognition				
	Name	Type	Default	Description
Parameters	Model Path	string	-	Path to the trained DNN model
	Process Size	[int, int]	640×360 pixels	The size of the image to be processed by the action detection algorithm
	Arguments	<i>JSON</i>	-	A key-value list of the parameters
Inputs	Image	DMSLib Image	-	The input image captured by the camera
	3D Body Points	DMSLib <i>Measure</i>	-	The array of 3D points of the body pose model with respect to the camera
	Object BBox List	DMSLib <i>Measure</i>	-	The list of bounding boxes of objects with its associated
Output	Driver Action	DMSLib <i>Measure</i>	-	The detected and classified driver action

Table 6.14: Parameters and Interfaces of the *Action Recognition* component

project VI-DAS (GA 690772)¹. The VI-DAS project was dedicated to enhancing road safety through the development and implementation of ADAS and navigation aids. The aim was to achieve this in a manner that is both socially acceptable and tailored to individual needs. This was to be accomplished by leveraging a robust understanding of the overall traffic environment, while also taking into account crucial factors such as the driver's physical and mental condition, demographic profile, and behavioral tendencies.

6.5.1 Human-Centered methodology for scenario definition

The VI-DAS project employed a Human-Centered Methodology, beginning with a meticulous analysis of accidents and driving errors. This approach guided the design of VI-DAS prototypes at every stage of development. To initiate this process, the project utilized two primary pre-existing databases to identify common challenges faced by drivers based on real-world data: an Accident Database (referred to as EDA [28]) and a Naturalistic Dataset (collected from instrumented vehicles).

The EDA5 program (Detailed Study of Accidents) [28] compiled detailed information on the core components of the driver-vehicle-environment system involved in crashes. This data aids in understanding the human involvement in the malfunction process (human errors, violations, and instances of exceeding capacity), as well as their underlying causes and contextual factors. Through a combined analysis of these datasets and Police Reports, the aim was to identify typical accident scenarios that could potentially be avoided in the future with the integration

¹<http://www.vi-das.eu/>

of VI-DAS technologies.

These datasets are leveraged to identify both behavioral (internal) and situational (external) indicators that unveil drivers' challenges, risky driving behaviors, or critical situations. These indicators are utilized for real-time monitoring of driving errors, facilitating the development of decision rules for an Advanced HMI. This HMI is designed to interact with the driver in an "adaptive" and "cooperative" manner, aligning with the driving context and situational risks.

Consequently, this Human-Centered Design (HCD) approach (illustrated in Figure 6.5) contributes to defining validation scenarios for the VI-DAS system and evaluating the effectiveness of the project's devices in addressing drivers' actual needs, potential driving errors, and real accident risks.

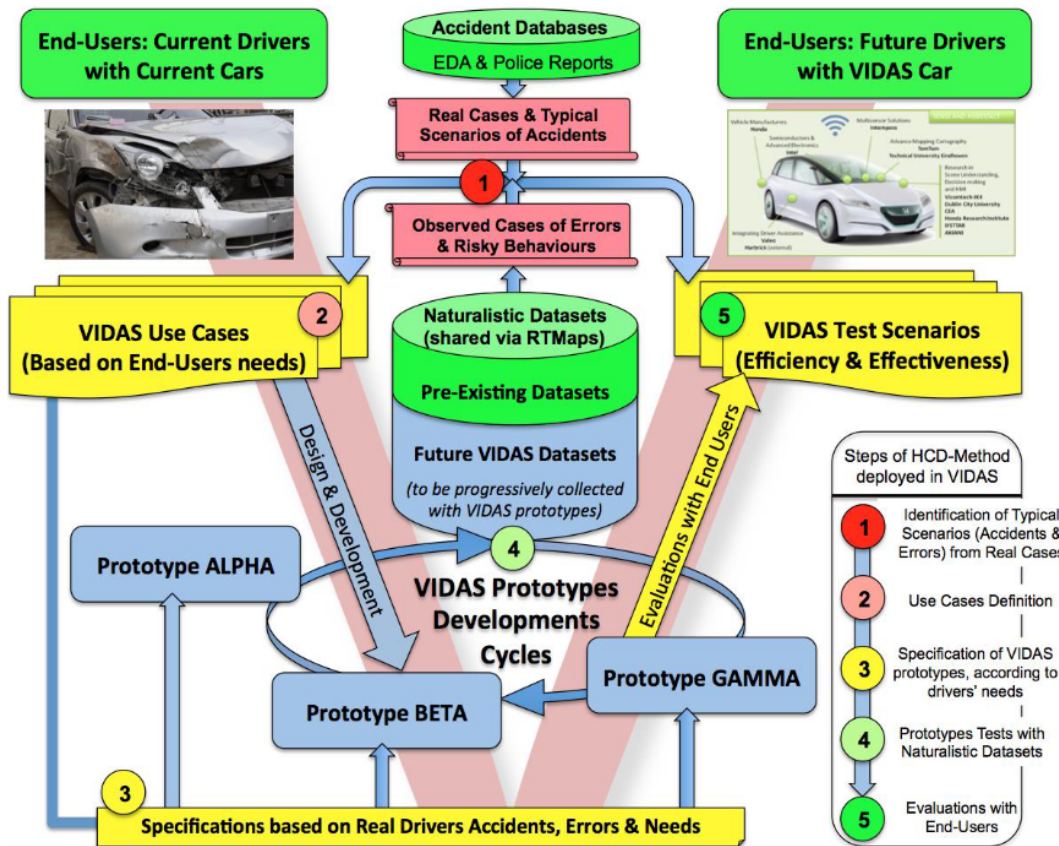


Figure 6.5: VI-DAS Human Centred design combined with development cycle V-model and cyclic approach of 3 prototypes.

The VI-DAS project proposed different use cases to tackle complex situations rather than simple actions and manoeuvre descriptions in different scenarios. One of the main objectives of VI-DAS is to address the hand-over and hand-back between manual and automated driving modes, focused on the driver's status and scene interpretation, always keeping the driver in the loop. Thus, the different scenarios and use cases selected in VI-DAS addressed new ways of including human factors for each of the performed maneuvers (regardless of the automated level) and continuously monitoring the driver's reactions to warnings and driving advice.

After conducting an initial study and proposing a wide array of potential use cases where VI-DAS could be impactful, a cross-check was performed using data from accident databases and naturalistic driving observations. Following the Human-Centric Design (HCD) methodology, two primary generic driving scenarios were pinpointed as especially pertinent for the VI-DAS project (illustrated in Figure 6.6): one centered around urban driving, and the other focusing on highway driving.

1. Generic Highway driving scenario: Lane Change (with a specific focus on Highway Insertion)

When driving on the highway, the primary challenges identified from the Naturalistic Driving Database and the Accident Databases arose during Lane Change maneuvers, regardless of the drivers' motivations. A key objective identified for VI-DAS is to facilitate Lane Changes within the context of complex road infrastructures, such as highway entrances, merges, or exits.

2. Generic Urban driving scenario: Intersection Crossing

In the urban context, intersection crossing emerged as the most critical situation and the most challenging scenario for VI-DAS, based on both the recorded accident risks in the EDA and the errors/difficulties observed in the Naturalistic Database.

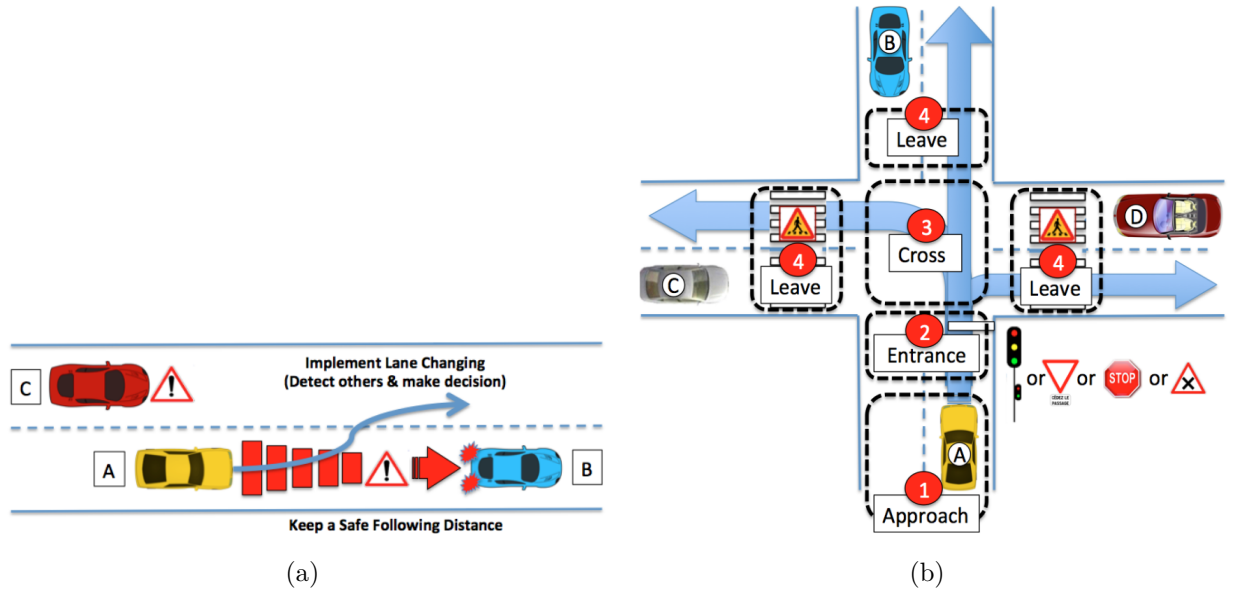


Figure 6.6: VI-DAS validation scenarios: (a) Generic Driving Scenario for Highway. (b) Generic Driving Scenario in Urban Area: Intersection Crossing.

To effectively address the validation activities conducted during the VI-DAS system integration process, it's crucial to differentiate between two significant concepts in the development of ADAS and AD systems: **verification** and **validation**.

1. **Verification** is the evaluation of a system to demonstrate that it satisfies all specified requirements at a specific stage of its development [21]. It addresses the developer's in-

quiries regarding the system's performance [98]. From a practical standpoint, verification is an *internal process* conducted by the component creator during the development phase to assess its precision, robustness, and other technical aspects. It primarily answers the developer's fundamental question: "Are we constructing this system correctly?" [56]

2. **Validation** is the process by which it is conclusively established that a final product aligns with its design objectives and stated performance criteria [21]. It addresses customer/user concerns about the system's validity [98]. In practice, validation is an *external procedure* carried out by the solution consumer during the assessment phase to gauge the solution's alignment with defined user requirements. It directly addresses the user's critical query: "Are you developing the correct system?" [56]

Therefore, the testing activities reported in this section primarily focused on validating the driving maneuvers executed by drivers, each with distinct profiles, and across multiple levels of automation. The verification stages of individual components are not detailed in this section, as they are the responsibility of each developer involved in component development. Concerning DMS systems, verification is conducted internally, with each DMS pipeline undergoing individual verification. In contrast, this section discusses the process and methodology of integrating various components into a unified system and proceeding with **validation**, facilitated by the HCD methodology to select validation scenarios and use cases.

6.5.2 Use case selection for validation

During the validation stage, the VI-DAS System under Test (SUT) and Functions under Test (FUT) were integrated into a functional prototype after three iterations. To validate the system, various use cases were chosen to implement these components in real vehicles and simulators. The following use cases, derived from the relevant scenario selection, were selected for the validation process:

1. **AHiDA-M** (Automatic Highway Driver Assistance - Manual): This use case demonstrates the assistance VI-DAS provides to the driver while driving manually on the highway, such as for lane changing or merging in risky situations.
2. **AHiDA-A** (Automatic Highway Driver Assistance - Autonomous): This use case illustrates the VI-DAS functionality to interact with the driver while the car drives autonomously, analyzing the driver's awareness of the situation.
3. **AICA** (Automatic Intersection Crossing Assistance): This use case defines a function of the VI-DAS vehicle which handles driving maneuvers at intersections, with automatic detection of other vehicles and pedestrians.

Moreover, to facilitate the validation process of various automated functionalities, the VI-DAS project established a set of context scenario-based tests. These test cases encompassed comprehensive environment descriptions, incorporating other road participants, road characteristics, and driver behaviors. The following lines describe the storyline of the various validation test cases conducted during the evaluation of the VI-DAS system, outlining the expected outcomes of the (SUT) for each scenario. Figure 6.7 illustrates the test cases 1 to 4.

1. **(AHiDA-M) test case 1:** The VI-DAS vehicle is being driven manually, and the driving lane ends ahead, requiring a lane change. Two cars are closely positioned in the left lane, leaving insufficient space to execute the lane change.

Expected result: The system assesses the surrounding environment and identifies the two vehicles and the limited gap between them. It notifies the driver, checks for signs of fatigue by monitoring eyelid aperture, confirms the driver's look at the recommended maneuver, and prompts the driver to execute a braking maneuver.

2. **(AHiDA-M) test case 2:** The VI-DAS vehicle is being driven manually, and the driving lane ends ahead, requiring a lane change. Two cars are closely positioned in the left lane, leaving sufficient space to execute the lane change.

Expected result: The system assesses the surrounding environment and identifies the two vehicles and the sufficient gap between them. It notifies the driver to change lane, checks for signs of fatigue by monitoring eyelid aperture, confirms the driver's look at the recommended maneuver, and the driver performs a safe lane change.

3. **(AHiDA-A) test case 3:** The driver engages the vehicle's autonomous mode while another vehicle is detected on the same lane (obstacle). Following, the driver is engaged with a smartphone and notices the system's warnings.

Expected result: The system evaluates the surrounding environment, detecting the presence of the approaching obstacle. It issues a notification to the driver, prompting them to resume control. The system verifies the driver's readiness to regain control, and the driver responds by assuming control of the vehicle and applying the brakes as necessary.

4. **(AHiDA-A) test case 4:** The driver engages the vehicle's autonomous mode while another vehicle is detected on the same lane (obstacle). Following, the driver is engaged with a smartphone and does not respond to the system's warnings.

Expected result: The system evaluates the surrounding environment, detecting the presence of the approaching obstacle. It issues a notification to the driver, prompting them to resume control. The system verifies the driver's readiness to regain control, where the driver is not responsive to the warnings, then the autonomous system performs an emergency break.

5. **(AICA) test case 5:** The VI-DAS vehicle is being driven manually as it approaches an intersection. The driver intends to make a left turn while a pedestrian is crossing at the zebra crossing.

Expected result: The system assesses the surroundings, recognizing the pedestrian's presence on the zebra crossing. It promptly alerts the driver. Furthermore, the system ensures that the driver has properly checked the left side before proceeding with the turn.

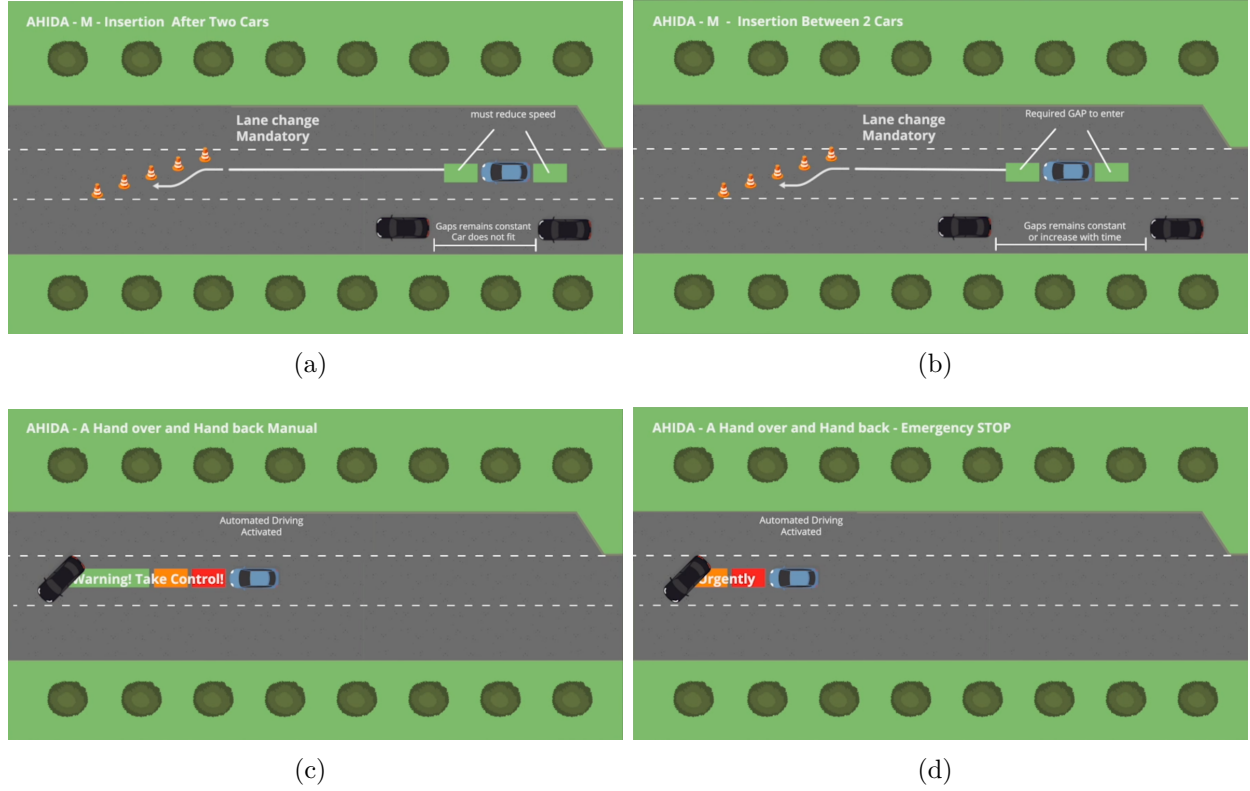


Figure 6.7: VI-DAS AHIDA-M and AHIDA-A Test Cases: (a) AHIDA-M Test Case 1 (b) AHIDA-M Test Case 2 (c) AHIDA-A Test Case 3 (d) AHIDA-A Test Case 4

6.5.3 Integration details

The previously described use cases require the integration of different DMS components such as eyelid aperture estimation (as described in Section 6.2), eye gaze estimation (as described in Section 6.3) and driver action recognition (as described in Section 6.4). Data coming from vehicle detectors, lane sensing, localization components and driver monitoring components is processed by a risk evaluation module which evaluates and assesses the current situation. When the external or internal conditions were negatively assessed by the risk assessing module, the vehicle's human-machine interface (HMI) advised the driver. Figure 6.10 showcases the different external sensing and internal monitoring components integrated into the architecture of the VI-DAS vehicle.

As part of the integration actions, a simulation set-up was used to gather ground truth material and to test the performance of the DMS C++ library in operational conditions before installing it into the instrumented vehicle. The simulator, as depicted in Figure 6.9 was set up to emulate the same camera locations of the test vehicle.

In addition, the simulator runs the PreScan simulator [192] to control scenarios and enable Driving-in-the-Loop test runs. A bridge between PreScan and RTMaps was built by the VI-DAS consortium to enable the utilization of the same RTMaps processing pipeline (as in Figure 6.8) but with simulated signals from PreScan.

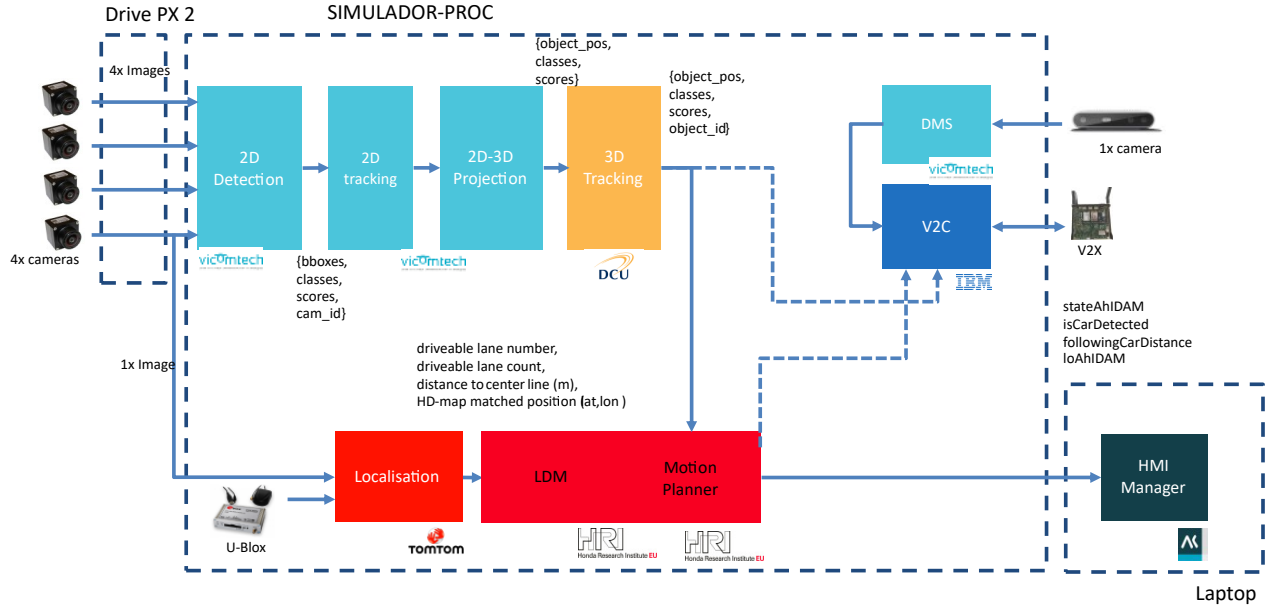


Figure 6.8: Component diagram of the VI-DAS instrumented vehicle

During the evaluation of the DMS pipelines, eyelid aperture estimation and gaze estimation were tested within a driving simulator using three different machines: a standard PC (Intel i5, 8GB RAM), an embedded platform (NXP i.MX6 ARM Cortex A9, 1GB RAM), and the NVIDIA Drive PX 2 platform (2XTegra X2 SoCs). The entire pipeline operates in real-time (>30 FPS) on both the PC and NVIDIA Drive PX 2, with computing times of 32.5 ms (30.8 FPS) and 23.2 ms (43.1 FPS) respectively. For the embedded platform, an operational speed of about 11 FPS (87.7 ms) is maintained, which is considered adequate for An embedded DMS system.

Moreover, during the VI-DAS integration activities, eyelid aperture estimation and gaze estimation components were integrated into the VI-DAS vehicles. The instrumented vehicle was equipped with interior cameras monitoring the face, hands and body of the driver (Intel RealSense D400 Series Depth), while 4 external fisheye cameras were used to create a 360° surround view system as shown in Figure 6.10. Lane detection and positioning systems along with risk estimation, IoT and HMI components completed the vehicle set-up.

The processing equipment integrated into the vehicle comprised a Master PC featuring an Intel i7-7740X CPU, 32 GB of RAM, an NVIDIA GTX Titan XP GPU card, and running Ubuntu 18.04. This setup facilitated the integration of various components, including multicamera object detection and 3D projection, object tracking, driver monitoring pipelines, Local Dynamic Map, Motion Planner, and Localization systems. Additionally, the vehicle was equipped with an APU2D4 device for V2C (Vehicle-to-Cloud) communications. For the HMI, a Windows laptop was utilized to integrate the HMI manager, responsible for outputting signals to the driver through the HMI devices. Furthermore, an NVIDIA Drive PX 2 platform was employed for capturing data from multiple external cameras and for seamless data streaming.



Figure 6.9: Integration of DMS pipelines in driving simulation: (a) Driver monitoring captures using driving simulator, (b) PreScan virtual world at AHiDA-M use case simulation.



Figure 6.10: Vehicle equipped with interior cameras (Intel RealSense D400) for face, body, and hand monitoring, alongside perception and localization devices (Velodyne HDL-32, 4 Sekonix 100° cameras, xNAV550, uBlox), and broadcasting mechanisms to display demo images outside the vehicle.

6.5.4 Testing campaigns and validation results

The testing campaigns involved executing the previously described test cases within the VI-DAS project. Real open road test runs were conducted in various locations to validate different use cases. Table 6.15 lists the specific locations where each test case (TC) was carried out as part of the VI-DAS validation activities.

Two cities were selected for the validation activities: San Sebastian, Spain, and Helmond, Netherlands. In San Sebastian, the focus was on integrating the components and executing specific TCs outlined by the project, such as AHIDA-M TC1, TC2, and AICA TC5. The primary demonstration occurred at the ITS Europe Congress 2019, held from 2nd to 6th June 2019 in Helmond, Netherlands (refer to Figure 6.11)². This demonstration focused on showcasing the seamless control transfer between human drivers and autonomous driving functions, with driver monitoring playing a crucial role in the system.

Tested Use Cases	Validation Locations
AHiDA-M TC1 & TC2	• AP1 Highway, San Sebastian, Spain • N270 Highway, Helmond, Netherlands
AHiDA-A TC3 & TC4	• N270 Highway, Helmond, Netherlands
AICA TC5	• Zuatzu, San Sebastian, Spain

Table 6.15: Locations where the validation campaigns were conducted for the VI-DAS project



Figure 6.11: DMS integration in VI-DAS lane change Use Case

To validate the integrated solution for each TC, validation checklists were created during test runs. These checklists aimed to evaluate the expected behavior outlined in the Human-centered Scenario definition phase and Use Case selection. A detailed description of the

² For a video summary of the demo, visit https://www.youtube.com/watch?v=8o3hT3H_gDU

expected results for each TC can be found in Section 6.5.2. These checklists were designed to verify the correct functioning of each VI-DAS macro-component during the conducted trials.

The validation results are presented in Table 6.16. This table outlines the various validation checks conducted during trials at different test locations (refer to Table 6.15). Some checks confirmed that the tested system components operated as expected, while others assessed not only the system’s expected performance but also its stability throughout the entire trial duration. For example, the Driver DMS functions were closely monitored to verify their robustness during the trials. It was observed, for instance, during AICA TC5 (urban scenario), that the DMS functions (eyelid aperture and gaze estimation) provided a valid measure *partially continuously*, indicating occasional periods where the system did not return valid measures. This behavior was attributed to significant head movements required by drivers in the Urban Scenario being tested.

Validation Check	AHiDA-M TC1	AHiDA-M TC2	AHiDA-A TC3	AHiDA-A TC4	AICA TC5
External perception detected vehicles?	✓	✓	✓	✓	✓
External perception detected pedestrians?	-	-	-	-	✓
Eyelid aperture estimation operation	continuous	continuous	-	-	partially continuous
Eye gaze estimation operation	continuous	continuous	-	-	partially continuous
DMS measures sent to IoT platform	continuous	continuous	continuous	continuous	partially continuous
Driver action recognition operation	-	-	continuous driver ready	continuous distraction	-
Risk situation assessment	continuous	continuous	continuous	continuous	continuous
HMI notified the driver	✓	✓	✓	✓	✓
Driver/AD performed maneuver	break and change lane	change lane	driver take control	emergency break	left turn with warnings
Validation result	expected	expected	expected	expected	partially expected

Table 6.16: VI-DAS validation checklist for each test case (described in Section 6.5.2). A checkmark indicates that the function operated as expected. A dash signifies that validation checking is not applicable for the testing case because the system function was not utilized during the trial test. Continuous and Partial Continuous assessment was conducted based on the robustness level of the validated function. Partial Continuous results imply instability of the function during testing.

Such insights provide integrators of these systems with valuable guidance for improving

system robustness, such as considering the use of additional cameras to enhance stability. In contrast, TCs conducted in highway scenarios, characterized by reduced head movements, demonstrated continuous system performance consistent with the expected outcomes defined during the use case definition phase.

To validate the system's real-time capabilities, the DMS signals were captured and seamlessly transmitted to an IoT platform for visualization in a web application. The data, including driver gaze direction and eyelid aperture level, was transmitted using IBM's Bluemix IoT platform and stored in georeferenced databases. Throughout the trials, it was consistently observed that signals were received by the IoT system at an average frequency of 10Hz, which proved adequate for real-time visualization on both the web application and map interface. Figure 6.12 illustrates the seamless performance of the integrated DMS and its efficient transmission of signals to the IoT platform, showcasing its real-time capabilities and Vehicle-to-Cloud (V2C) functionality [157], [158]. Operators accessing the web application were able to monitor, in real-time, the driver's position, eye movements, and head signals.

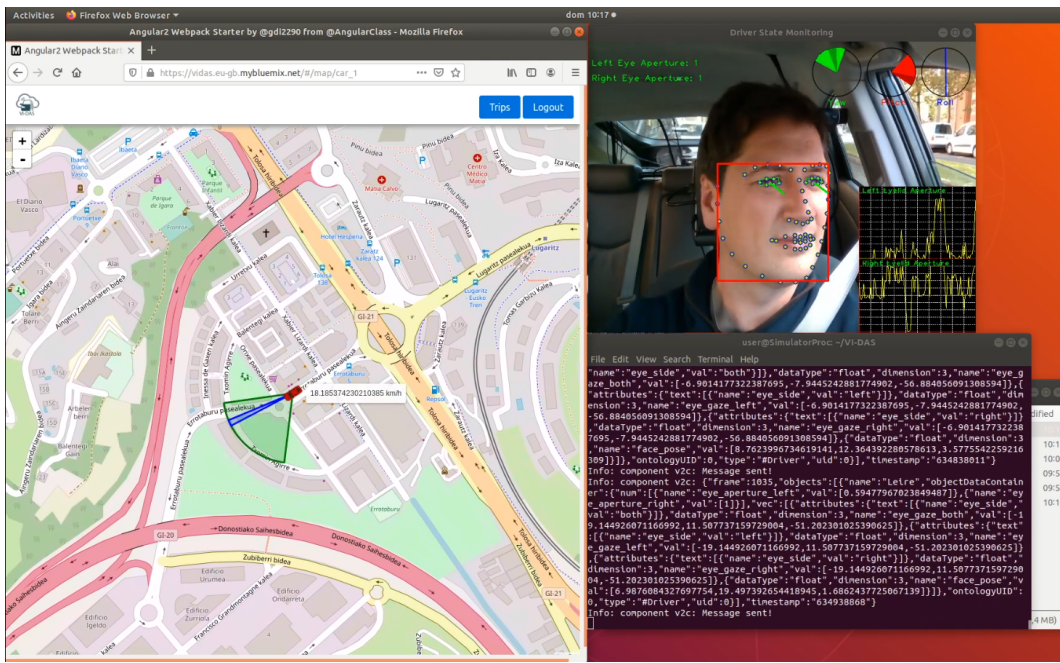


Figure 6.12: DMS signals (eyelid aperture and gaze estimation) transmitted to an IoT platform during the AICA TC5 trial. The accompanying image depicts a web application featuring an online map, showcasing the vehicle's position indicated by two semicircles: a larger green semicircle representing head movement and a smaller blue semicircle indicating the driver's gaze direction.

6.6 Conclusion

This chapter has outlined the essential elements required for implementing various use cases of driving monitoring. Specifically, three use cases have been covered: driver eyelid aperture estimation, driver gaze estimation, and driver action recognition. Each use case

has been accompanied by a comprehensive description of the underlying technologies and the components involved in their implementation. Furthermore, a detailed depiction of parameters and interfaces (inputs and outputs) for each component has been provided. All implementations of these use cases are compliant with the proposed DMSLib software framework (refer to Chapter 5). The modular nature of the proposed approach facilitates the rapid implementation of new functionalities and the integration of additional components.

The proposed pipelines were successfully implemented and tested in both a driving simulator and a real research car. To ensure accurate validation, the camera configuration for both setups remained consistent, enabling a thorough examination of the solution’s effectiveness. Notably, this camera setup aligns with the configuration chosen for the creation of the DMD (refer to Chapter 4). Consequently, the pipelines were evaluated using the extensive dataset available from the DMD, further enhancing the reliability and comprehensiveness of the evaluation process.

Incorporated into an operational test vehicle, two distinct pipelines were employed: one dedicated to eyelid aperture monitoring and another to gaze estimation. The data produced by these pipelines served as critical inputs for a range of exterior sensing components and risk assessment algorithms. These integrated components and algorithms were pivotal in discerning the driver’s state and evaluating the level of risk within the driving environment. Various scenarios and use cases were meticulously defined following a Human-Centered methodology. Carefully selected test cases were employed to validate the anticipated behavior of each predefined scenario across diverse locations. The outcomes of these trials provided invaluable insights into the performance of the integrated system, revealing instances of partially anticipated behavior, particularly notable in urban settings where significant driver head movements were observed. Consequently, these findings played a pivotal role in facilitating the smooth transition of control between the human driver and the AD function, enhancing overall system reliability and safety.

Chapter 7

Conclusions and future work

This thesis has explored the process to develop industrial DMS based on the analysis of human behavior to define and develop driver monitoring systems, including the creation of complex datasets for its training and validation. The following sections summarize and highlight the main contributions of the research work done in this thesis, and discuss the research questions which drive its work, addressing the main findings, outcomes and limitations. This discussion guides the future work section, in which the potential evolution of this work is presented, as well as the next steps for future developments in the field of driving monitoring.

7.1 Conclusions

This thesis begins with its **first contribution** in Chapter 2, which involves a comprehensive analysis of driver state definitions and taxonomy from a CV perspective. This process includes a thorough review of the existing literature on human factors related to driving state monitoring and their impact on the driver's physiological state. Through this analysis, various physiological mechanisms of driver inattention have been identified, which can guide the design and implementation of vision-based DMS. However, establishing a direct correlation between human attention behavior and the technological capabilities of state-of-the-art CV and AI methods is a challenging task.

The complexity of human behavior, especially when internal and external factors of human physiology are present can limit the application of automatic methods to sense driver's state. Many signs of altered driver state can be seen by the human eyes, which allows the design and development of automated systems to sense and assess driver state. Hence, this thesis provides a classification of driver state behavior attending to the visual characteristics of the phenomena. Two mechanisms of driver inattention were identified: fatigue and distraction which were the object of study of this thesis. It has been seen that other mechanisms of inattention such as emotions and cognitive workload could also affect the overall driver's inattention state. However, their effects on the driver's face and body are not always visible. For instance, there have been seen driver distraction moments which are produced by "lost in thoughts" episodes which is a form of cognitive workload behavior.

This analysis plays a crucial role in addressing RQ1 as it identifies a complete set of fatigue and distraction signs. These findings have been instrumental in designing other components that contributed to this thesis. Special emphasis was placed on visual cues observed in the human face, where fatigue and drowsiness driver states are prominently manifested. Conversely, distraction could be detected by analyzing other parts of the driver’s body. Consequently, when defining a DMS, these considerations should be taken into account, including the placement of sensors and cameras within the car cabin. The established classification of driver inattention symptoms serves as a valuable reference for other researchers in the field of driver monitoring. Moreover, this study has guided the work performed in Chapter 3 for creating novel vision-based eyelid aperture analysis, in Chapter 4 for creating the DMD and the DMS modular software framework in Chapter 5.

In Chapter 3, the **second contribution** of this work presents a novel user-adaptive eyelid aperture estimation method was introduced, which holds potential for advanced fatigue estimation systems. This method applies online adaptation of an individual’s eyelid basal state to obtain a normalized aperture level that remains invariant across different individuals’ physiognomies. The implementation of this method followed the principles of the DMSLib framework and was validated using data from the DMD.

The validation results on the DMD dataset demonstrated that the proposed method outperformed other state-of-the-art techniques in blink assessment. Its key advantage lies in its user adaptiveness feature, which showed improved blink detection results when analyzing individual data, which addresses RQ2. However, it is important to note that the method has limitations in accurately sensing eyelid aperture under certain conditions, such as when the subject wears sunglasses or when specular reflections in glasses obstruct the visibility of the eyes. These challenges become particularly prominent in setups employing IR illuminators to illuminate the scene at night.

One possible solution to address these limitations is to decouple the incidence angle of the IR illuminators from the camera’s main axis. However, it should be acknowledged that this approach may not entirely resolve the issue of specular reflections and could necessitate the use of alternative light sources installed in the vehicle.

The **third contribution** of this thesis, presented in Chapter 4, introduces a metadata taxonomy for driver monitoring datasets. During the course of this research, it was observed that there is a significant lack of publicly available and comprehensive datasets suitable for developing DMSs. In response to this gap, the proposed taxonomy addresses RQ3 by identifying the essential characteristics that a dataset should possess, including context, geometrical, and temporal metadata necessary to meet the requirements of DMS. Driver state is a complex behavior that evolves over time, and therefore, the labeling of driver monitoring datasets must incorporate this temporal dimension. Hence, the proposed taxonomy includes both instantaneous and temporal metadata. When designing DMS, researchers and practitioners can utilize any or all of these dimensions of the dataset to meet their specific needs.

In addition, Chapter 4 presents a **fourth contribution** of this thesis, describing the complete process of creating a large-scale dataset for vision-based DMS. The development of this thesis

revealed the great need for a comprehensive dataset with most of the characteristics identified by the dataset taxonomy. The methodology proposed facilitates the creation of large-scale, multi-purpose, multi-sensor datasets for the development of DMS and addresses RQ4 by defining the plan and steps needed to build the DMD, one of the largest open driver monitoring datasets known at the moment.

The creation of the DMD dataset involved three stages: preparation, recording, and post-processing. During the preparation stage, challenges arose in defining the relevant behaviors that needed to be captured by the sensors. Based on the types of driver inattention studied (fatigue and distraction), a scripted protocol was developed to capture a diverse set of driver states. The volunteers who participated in the creation of the DMD were asked to perform the actions as naturally as possible and without giving more details to interfere with their participation. However, the recordings captured similar behavior among the participants, which could introduce some bias when training algorithms with the DMD dataset.

Moreover, during the preparation stage, the selection of camera sensors and their positions in the recording setup were defined. The use of RGB-Depth cameras was preferred due to the requirement of capturing more than just RGB modality. Future DMSs are expected to operate under both day and night conditions, necessitating the use of modalities beyond RGB. Therefore, the DMD employed different multi-modal camera sensors to capture Depth, Infrared (IR), and RGB information.

The use of IR cameras has been proposed as a solution for low-light conditions in many works in the literature. However, during the DMD recording process, it was observed that IR would produce specular reflections on users wearing glasses, occluding the driver's eyes and limiting the analysis of eye-related DMS functionalities. This physical limitation of the cameras does not have an immediate solution through image processing methods. Therefore, the fusion of CV algorithms with the analysis of other driver parameters, such as driving patterns, is needed to improve the robustness of DMSs.

The second stage of the DMD creation involved recording the driving sessions. This stage led to important findings regarding the materials to be used. The proposed methodology utilized uncompressed data formats to preserve the original quality of the recordings. However, this could cause bandwidth issues in the recording equipment. Therefore, careful selection of camera framerate and image format is crucial to avoid limitations when recording data on regular data recording devices.

The third stage of the dataset creation process involved post-processing the captured data. For the DMD, a step of data synchronization between camera feeds was necessary due to small shifts. A method to align and calculate these small shifts was proposed, which was essential for annotating the sequences. Annotated multi-view sequences require synchronized videos to optimize the annotation effort. Additionally, compressed versions of the resulting sequences are needed for ingestion into annotation tools. An improvement for future dataset creation would be to include hardware synchronization between cameras to reduce errors in software synchronization.

Annotating large-scale datasets like the DMD can be time-consuming and challenging, blocking the further development of DMS. Two approaches were explored in this thesis. The first

approach involved manual annotation of the defined classes, which requires a large team of well-trained annotators to label the entire dataset. However, this may not always be feasible, and alternative strategies such as semi-automatic annotation can be employed. This thesis explored the use of semi-automatic labeling for annotating the DMD's temporal features for distraction detection. It was found that semi-automatic labeling significantly reduced annotation times with fewer human resources allocated.

In Chapter 5, the **fifth contribution** of the thesis is presented, motivated by the analysis of RQ5. This led to the classification of DMS functions based on complexity levels: low-level, physiological, and driver status features. Each level requires an increasing level of physiological analysis of the driver's visual cues.

Furthermore, Chapter 5 presents the **sixth contribution**, which is the architecture of a modular software framework for building industrial-grade DMS applications. This architecture enables the optimization of algorithms by reusing low-level modules in more complex methods. It allows for optimal implementation and meets real-time specific requirements. The framework's application was tested using the industrial-grade middleware application RTMaps, which facilitates the connection with other automobile automated systems.

The proposed DMS framework facilitates the deployment of various driver monitoring pipelines tailored to the specific requirements of the DMS function. Chapter 6 introduces the **seventh contribution**, which is the description of three pipelines: eyelid aperture estimation, eye gaze estimation, and action recognition. This work presents a novel approach to describe different DMS functions, taking into account the proposed DMSLib software framework. The framework has successfully addressed the RQ5 by defining and describing the parameters, inputs, and outputs of these DMS functions.

A significant achievement of this research is the practical implementation of the eyelid aperture estimation and gaze estimation pipelines in both real testing scenarios using a vehicle and a driving simulator. The outputs from these pipelines were utilized by advanced situation awareness algorithms to enable the integration of sophisticated autonomous features. The key takeaway from this contribution emphasizes the necessity of modular components within an architecture where DMS components can seamlessly integrate with other vehicle features, potentially within a common vehicle Electronic Control Unit (ECU). Furthermore, it highlights the importance of a standardized interface for communication with other components. This is precisely why the DMSLib framework was developed: to provide a common interface for effective communication and offer a modular architecture that simplifies the integration of new DMS functions.

7.2 Future work

Characterizing the state of a driver and, more generally, DM will remain important despite the progressive increase in vehicle automation. The automotive industry has realized the vision to have a fully automated vehicle is far away to be feasible. SAE Level 3 enables vehicles to drive by themselves under certain conditions such as on a highway and in sunny weather, but a driver must still be present and able to take back control of the vehicle at

any time and in a relatively short lapse of time. To ensure that the driver can take back control, technologies for monitoring the state of the driver will become even more critical. These technologies are also needed to monitor the driver during the time he/she is driving and to possibly allow the vehicle to take back control if necessary.

Currently, some vehicle manufacturers offer Driver Monitoring Systems (DMSs) based on driver behavior and vehicle behavior, such as detecting steering-wheel movements and lane deviations, respectively. While these systems are useful for vehicles with automation up to SAE Level 2, they become obsolete at higher levels of automation. When a vehicle operates autonomously, monitoring its behavior provides no information about the state of the driver. Therefore, technologies that directly monitor both the driver and the driving environment become necessary, as long as the driver is involved in the driving task, even partially.

In this direction, driver inattention is a complex state of human behavior that can be influenced by many factors. In this thesis, special attention was given to two mechanisms: fatigue and distraction. However, other mechanisms can also be factored into the equation such as emotions and cognitive workload. These last two factors sometimes are not easy to measure and assess using CV methods since it does not visually manifest in the driver's body. Therefore, this makes it very hard to visually distinguish their signs. The inclusion of other sources of data, hence, is a pending topic in this field. This thesis contributed with a multi-modal dataset for DM. Then, research should continue in the direction of incorporating these different modalities, besides RGB/IR, into the design of automated DMSs.

The field's future lies in developing more holistic in-cabin monitoring solutions. Driven by regulations and the need to enhance driver and passenger safety, prompted by new vehicle ratings like EuroNCAP, the automotive industry must develop occupant monitoring solutions that consider both the driver and their interaction with other occupants. Vision-based solutions for in-cabin monitoring show promise due to existing CV algorithms that can detect and analyze human features within the cabin. However, other data sources like radars or depth cameras could also complement visual information. The fusion of these diverse data sources will pose a challenge for future DMSs.

Future work should include the fusion of external and internal data sources to provide a more comprehensive view of the driver's state. External data sources include perception of the vehicle surrounding. Internal data sources include driver and occupant information. This fusion will enable a better situation awareness specially useful for L3+ vehicles.

The presented method for estimating the eyelid aperture level is a promising approach for fatigue detection. However, different aspects can be further studied to understand the driver's state better. For instance, in this work the eyelid aperture normalization was performed using a Bayesian approach. Where the prior probability was defined based on the assumption that there is not prior information of the driver's eyes being either open and closed. This assumption can be further studied to understand if there is a better way to define this prior probability.

The DMD is a live project aiming to collect a diverse and comprehensive set of data to model driver state and behavior. The future of the DMD should focus on expanding the dataset to include more diverse driving conditions. The dataset should include more naturalistic

recordings of different driving scenarios, such as driving in urban areas, highways, and rural roads. The dataset should also include recordings of different weather conditions, such as rain, snow, and fog. These recordings should be annotated with the driver's state, including fatigue and distraction, to enable the development of DMSs that can operate under different driving conditions. Finally, the dataset could be extended to include more participants with the objective of reducing the gender, age, driving experience and any other bias that could be present in the dataset.

Additionally, existing and new recordings can benefit from a semi-automatic annotation framework and incorporate active learning concepts to gradually reduce manual annotation efforts.

Moreover, the generation of the DMD could also benefit from modern techniques to generate synthetic data based on 3D modeling techniques. This could help to increase the diversity of the dataset and to reduce the bias that could be present in the dataset. The use of synthetic data could also help to reduce the annotation effort needed to generate the dataset.

While numerous methods exist for low-level features such as face detection and body pose, limited methods have been validated for measuring driver state. Therefore, a prominent future direction is the development and validation of methods that provide medically validated measures of driver state.

To date, the development of ADS has outpaced the development of DMSs. This can be attributed, in large part, to the belief held by some automotive industry players that they could easily bypass Levels 3 and 4 and directly transition to Level 5, where driver monitoring is unnecessary. However, the current evidence has shown that it will take decades, if ever, for most privately-owned vehicles to become fully automated. Along the challenging journey towards Level 5, the automotive industry must significantly boost research and development efforts for DMSs. Both ADSs and DMSs need to be developed synergistically. The future of the field of Driver Monitoring and DMSs could not be more promising.

7.3 Bibliographic contributions

The following publications have been produced or participated during the course of this thesis:

- **Journal papers and book chapters**

- **Title:** Challenges of Large-Scale Multi-Camera Datasets for Driver Monitoring Systems
Authors: Juan Diego Ortega, Paola Cañas, Marcos Nieto, Oihana Otaegui, Luis Salgado
Journal: Sensors
Year: 2022
Cite: [161]
- **Title:** Open your eyes: Eyelid aperture estimation in Driver Monitoring Systems
Authors: Juan Diego Ortega, Paola Cañas, Marcos Nieto, Oihana Otaegui, Luis Salgado

- Book:** Communications in Computer and Information Science
Year: 2021
Cite: [160]
- **Title:** On creating vision-based advanced driver assistance systems
Authors: Marcos Nieto, Oihana Otaegui, Gorka Vélez, **Juan Diego Ortega**, Andoni Cortés
Journal: IET Intelligent Transport Systems
Year: 2015
Cite: [148]
 - **Conference papers**
 - **Title:** A real-time software framework for driver monitoring systems: software architecture and use cases
Authors: **Juan Diego Ortega**, Marcos Nieto, Paola Cañas, Oihana Otaegui, Luis Salgado
Conference: Real-Time Image Processing and Deep Learning
Year: 2021
Cite: [163]
 - **Title:** Detection of Distraction-related Actions on DMD: An Image and a Video-based Approach Comparison
Authors: Paola Cañas, **Juan Diego Ortega**, Marcos Nieto, Oihana Otaegui
Conference: 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)
Year: 2021
Cite: [25]
 - **Title:** User-adaptive Eyelid Aperture Estimation for Blink Detection in Driver Monitoring Systems
Authors: **Juan Diego Ortega**, Marcos Nieto, Luis Salgado, Oihana Otaegui
Conference: 6th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS)
Year: 2020
Cite: [164]
 - **Title:** DMD: A Large-Scale Multi-Modal Driver Monitoring Dataset for Attention and Alertness Analysis
Authors: **Juan Diego Ortega**, Neslihan Kose, Paola Cañas, Min-An Chao, Alexander Unnervik, Marcos Nieto, Oihana Otaegui, Luis Salgado
Conference: Proceedings of the European Conference on Computer Vision (ECCV) Workshops
Year: 2020
Cite: [162]
 - **Title:** A Cloud-Based AI Framework for Machine Learning Orchestration: A "Driving or Not-Driving" Case-Study for Self-Driving Cars

- Authors:** Cristian Olariu, Haytham Assem, **Juan Diego Ortega**, Marcos Nieto
Conference: IEEE Intelligent Vehicles Symposium, Proceedings
Year: 2019
Cite: [157]
- **Title:** Efficient monocular point-of-gaze estimation on multiple screens and 3D face tracking for driver behaviour analysis
Authors: Jon Goenetxea, Luis Unzueta, Unai Elordi, **Juan Diego Ortega**, Oihana Otaegui
Conference: 6th Driver Distraction and Inattention Conference
Year: 2018
Cite: [77]
 - **Title:** The role of cloud-computing in the development and application of ADAS
Authors: Cristian Olariu, **Juan Diego Ortega**, Javier Yebes
Conference: European Signal Processing Conference
Year: 2018
Cite: [158]
- **Public Datasets and Dissemination Videos**
 - **Title:** Driver Monitoring Dataset
URL: <https://dmd.vicomtech.org/>
 - **Title:** VI-DAS Demonstration Video
URL: https://www.youtube.com/watch?v=8o3hT3H_gDU

References

- [1] H. Abraham, B. Reimer, and B. Mehler, “Advanced driver assistance systems (adas): A consideration of driver perceptions on training, usage & implementation”, in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 61, 2017, pp. 1954–1958. DOI: [10.1177/1541931213601967](https://doi.org/10.1177/1541931213601967).
- [2] Affectiva. “Interior sensing ai”. (2023), [Online]. Available: <https://go.affectiva.com/auto> (visited on 05/17/2023).
- [3] A. S. Aghaei, B. Donmez, C. C. Liu, *et al.*, “Smart driver monitoring: When signal processing meets human factors: In the driver’s seat”, *IEEE Signal Processing Magazine*, vol. 33, pp. 35–48, 2016. DOI: [10.1109/MSP.2016.2602379](https://doi.org/10.1109/MSP.2016.2602379).
- [4] C. Ahlström and K. Kircher, “Review of real-time visual driver distraction detection algorithms”, in *7th International Conference on Methods and Techniques in Behavioral Research*, ACM, 2010. DOI: [10.1145/1931344.1931346](https://doi.org/10.1145/1931344.1931346).
- [5] L. S. Angell, J. L. Aufflick, P. A. Austria, *et al.* “Driver workload metrics project: Task 2 final report”. (2006), [Online]. Available: https://www.nhtsa.gov/sites/nhtsa.gov/files/documents/driver_workload_metrics_appendices.pdf (visited on 07/09/2023).
- [6] ASAM. “Openlabel concept paper, version 1.0”. (Jul. 2020), [Online]. Available: <https://www.asam.net/index.php?eID=dumpFile&t=f&f=3876&token=413e8c85031ae64cc35cf42d0768627514868b2f> (visited on 05/17/2023).
- [7] The Associated Press. “Congress’ new mandate to carmakers: Figure out a way to stop drunk driving”. (2021), [Online]. Available: <https://www.npr.org/2021/11/09/1053847935/congress-cars-drunk-driving-auto> (visited on 07/09/2023).
- [8] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic, “Incremental face alignment in the wild”, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2014, pp. 1859–1866. DOI: [10.1109/CVPR.2014.240](https://doi.org/10.1109/CVPR.2014.240).
- [9] M. H. Baccour, F. Driewer, E. Kasneci, and W. Rosenstiel, “Camera-based eye blink detection algorithm for assessing driver drowsiness”, in *IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2019, pp. 987–993. DOI: [10.1109/IVS.2019.8813871](https://doi.org/10.1109/IVS.2019.8813871).
- [10] T. Baltrusaitis, P. Robinson, and L.-P. Morency, “Openface: An open source facial behavior analysis toolkit”, in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2016, pp. 1–10. DOI: [10.1109/WACV.2016.7477553](https://doi.org/10.1109/WACV.2016.7477553).
- [11] L. Barr, S. Popkin, and H. Howarth, “An evaluation of emerging driver fatigue detection measures and technologies-final report”, U.S. Department of Transportation, Federal Motor Carrier Safety Administration, Jun. 2009, pp. 1–41.

- [12] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, and M. Grundmann, *Blazepose: On-device real-time body pose tracking*, 2020. arXiv: [2006.10204](https://arxiv.org/abs/2006.10204) [cs.CV].
- [13] S. Begum, “Intelligent driver monitoring systems based on physiological sensor signals: A review”, in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2013, pp. 282–289. DOI: [10.1109/ITSC.2013.6728246](https://doi.org/10.1109/ITSC.2013.6728246).
- [14] T. Bellet, M. Cunneen, M. Mullins, *et al.*, “From semi to fully autonomous vehicles: New emerging risks and ethico-legal challenges for human-machine interactions”, *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 63, pp. 153–164, 2019. DOI: [10.1016/j.trf.2019.04.004](https://doi.org/10.1016/j.trf.2019.04.004).
- [15] L. M. Bergasa, J. Nuevo, M. A. Sotelo, R. Barea, and M. E. Lopez, “Real-time system for monitoring driver vigilance”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, pp. 63–77, 2006. DOI: [10.1109/TITS.2006.869598](https://doi.org/10.1109/TITS.2006.869598).
- [16] C. Berka, D. J. Levendowski, M. N. Lumicao, *et al.*, “Eeg correlates of task engagement and mental workload in vigilance, learning, and memory tasks.”, *Aviation, space, and environmental medicine*, vol. 78, B231–44, 2007.
- [17] G. Borghi, M. Fabbri, R. Vezzani, S. Calderara, and R. Cucchiara, “Face-from-depth for head pose estimation on depth images”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 596–609, 2020. DOI: [10.1109/TPAMI.2018.2885472](https://doi.org/10.1109/TPAMI.2018.2885472).
- [18] G. Borghi, M. Venturelli, R. Vezzani, and R. Cucchiara, “Poseidon: Face-from-depth for driver pose estimation”, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2017, pp. 5494–5503. DOI: [10.1109/CVPR.2017.583](https://doi.org/10.1109/CVPR.2017.583).
- [19] G. Borghini, L. Astolfi, G. Vecchiato, D. Mattia, and F. Babiloni, “Measuring neurophysiological signals in aircraft pilots and car drivers for the assessment of mental workload, fatigue and drowsiness”, *Neuroscience and Biobehavioral Reviews*, vol. 44, pp. 58–75, 2014. DOI: [10.1016/j.neubiorev.2012.10.003](https://doi.org/10.1016/j.neubiorev.2012.10.003).
- [20] L. N. Boyle, J. Tippin, A. Paul, and M. Rizzo, “Driver performance in the moments surrounding a microsleep”, *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 11, pp. 126–136, 2008. DOI: [10.1016/j.trf.2007.08.001](https://doi.org/10.1016/j.trf.2007.08.001).
- [21] British Standards Institution (BSI). “Connected and automated mobility - vocabulary, version 5.0”. (Apr. 2023), [Online]. Available: https://www.bsigroup.com/globalassets/localfiles/aaa/bsi_flex_1890_v5.pdf (visited on 03/18/2024).
- [22] Cadillac. “Cadillac super cruise system”. (2023), [Online]. Available: <https://www.cadillac.com/technology/super-cruise> (visited on 05/17/2023).
- [23] H. Caesar, V. Bankiti, A. H. Lang, *et al.*, “Nuscenes: A multimodal dataset for autonomous driving”, in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2020, pp. 11 618–11 628. DOI: [10.1109/CVPR42600.2020.01164](https://doi.org/10.1109/CVPR42600.2020.01164).
- [24] J. L. Campbell, J. L. Brown, J. S. Graving, *et al.* “Human factors design principles for level 2 and level 3 automated driving concepts”. (2018), [Online]. Available: https://www.nhtsa.gov/sites/nhtsa.gov/files/documents/13494_812555_1213automationhfguidance.pdf (visited on 05/17/2023).
- [25] P. Cañas, J. D. Ortega, M. Nieto, and O. Otaegui, “Detection of distraction-related actions on dmd: An image and a video-based approach comparison”, in *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Com-*

- puter Graphics Theory and Applications*, vol. 5, 2021, pp. 458–465. DOI: [10.5220/0010244504580465](https://doi.org/10.5220/0010244504580465).
- [26] Cardio-ID. “Cardiowheel”. (2023), [Online]. Available: <https://www.cardio-id.com> (visited on 07/03/2023).
 - [27] M. I. Chacon-Murguia and C. Prieto-Resendiz, “Detecting driver drowsiness: A survey of system designs and technology”, *IEEE Consumer Electronics Magazine*, vol. 4, pp. 107–119, 2015. DOI: [10.1109/MCE.2015.2463373](https://doi.org/10.1109/MCE.2015.2463373).
 - [28] A. Chand, S. Jayesh, and A. Bhasi, “Road traffic accidents: An overview of data sources, analysis techniques and contributing factors”, *Materials Today: Proceedings*, vol. 47, pp. 5135–5141, 2021, International Conference on Sustainable materials, Manufacturing and Renewable Technologies 2021. DOI: <https://doi.org/10.1016/j.matpr.2021.05.415>.
 - [29] A. Chowdhury, R. Shankaran, M. Kavakli, and M. M. Haque, “Sensor applications and physiological features in drivers’ drowsiness detection: A review”, *IEEE Sensors Journal*, vol. 18, pp. 3055–3067, 2018. DOI: [10.1109/JSEN.2018.2807245](https://doi.org/10.1109/JSEN.2018.2807245).
 - [30] Consumer Reports. “Tesla’s camera-based driver monitoring still fails to keep driver attention on the road”. (2021), [Online]. Available: <https://goo.by/T5BtJ> (visited on 07/03/2023).
 - [31] M. Cordts, M. Omran, S. Ramos, *et al.*, “The cityscapes dataset for semantic urban scene understanding”, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2016, pp. 3213–3223. DOI: [10.1109/CVPR.2016.350](https://doi.org/10.1109/CVPR.2016.350).
 - [32] M. L. Cunningham and M. A. Regan, “Chapter 4. driver distraction and inattention”, in *Transport and Sustainability*. 2018, vol. 11, pp. 57–82. DOI: [10.1108/S2044-994120180000011004](https://doi.org/10.1108/S2044-994120180000011004).
 - [33] B. Cyganek and S. Gruszczyński, “Hybrid computer vision system for drivers’ eye recognition and fatigue monitoring”, *Neurocomputing*, vol. 126, pp. 78–94, 2014. DOI: [10.1016/j.neucom.2013.01.048](https://doi.org/10.1016/j.neucom.2013.01.048).
 - [34] T. D’orazio, M. Leo, C. Guaragnella, and A. Distanto, “A visual approach for driver inattention detection”, *Pattern Recognition*, vol. 40, pp. 2341–2355, 2007. DOI: [10.1016/j.patcog.2007.01.018](https://doi.org/10.1016/j.patcog.2007.01.018).
 - [35] M. Daniluk, M. Rezaei, R. Nicolescu, and R. Klette, “Eye status based on eyelid detection: A driver assistance system”, in *International Conference on Computer Vision and Graphics*, 2014, pp. 171–178. DOI: [10.1007/978-3-319-11331-9_21](https://doi.org/10.1007/978-3-319-11331-9_21).
 - [36] T. Danisman, I. M. Bilasco, C. Djeraba, and N. Ihaddadene, “Drowsy driver detection system using eye blink patterns”, in *International Conference on Machine and Web Intelligence*, IEEE, 2010, pp. 230–233. DOI: [10.1109/ICMWI.2010.5648121](https://doi.org/10.1109/ICMWI.2010.5648121).
 - [37] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database”, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2009, pp. 248–255. DOI: [10.1109/CVPRW.2009.5206848](https://doi.org/10.1109/CVPRW.2009.5206848).
 - [38] N. Deo and M. M. Trivedi, “Looking at the driver/rider in autonomous vehicles to predict take-over readiness”, *IEEE Transactions on Intelligent Vehicles*, vol. 5, pp. 41–52, 2020. DOI: [10.1109/TIV.2019.2955364](https://doi.org/10.1109/TIV.2019.2955364).
 - [39] K. Diaz-Chito, A. Hernández-Sabaté, and A. M. López, “A reduced feature set for driver head pose estimation”, *Applied Soft Computing*, vol. 45, pp. 98–107, 2016. DOI: [10.1016/j.asoc.2016.04.027](https://doi.org/10.1016/j.asoc.2016.04.027).

- [40] W. Dong, J. Li, R. Yao, C. Li, T. Yuan, and L. Wang, *Characterizing driving styles with deep learning*, 2016. arXiv: [1607.03611 \[cs.AI\]](#).
- [41] Y. Dong, Z. Hu, K. Uchimura, and N. Murayama, “Driver inattention monitoring system for intelligent vehicles: A review”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, pp. 596–614, 2011. DOI: [10.1109/TITS.2010.2092770](#).
- [42] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, “Centernet: Keypoint triplets for object detection”, in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 6568–6577. DOI: [10.1109/ICCV.2019.00667](#).
- [43] P. Ebrahim, W. Stolzmann, and B. Yang, “Eye movement detection for assessing driver drowsiness by electrooculography”, in *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, 2013, pp. 4142–4148. DOI: [10.1109/SMC.2013.706](#).
- [44] T. Edwards, L. Whichello, and A. Yee, “Do not believe your eyes: The use of synthetic data for driver monitoring product development”, Seeing Machines, 2023.
- [45] A. El Basiouni El Masri, H. Artail, and H. Akkary, “Toward self-policing: Detecting drunk driving behaviors through sampling can bus data”, in *International Conference on Electrical and Computing Technologies and Applications (ICECTA)*, IEEE, 2017, pp. 1–5. DOI: [10.1109/ICECTA.2017.8252037](#).
- [46] M. R. Endsley, “Toward a theory of situation awareness in dynamic systems”, in *Human Error in Aviation*. 2017, pp. 217–249. DOI: [10.4324/9781315092898-13](#).
- [47] H. M. Eraqi, Y. Abouelnaga, M. H. Saad, and M. N. Moustafa, “Driver distraction identification with an ensemble of convolutional neural networks”, *Journal of Advanced Transportation*, pp. 1–12, 2019. DOI: [10.1155/2019/4125865](#).
- [48] European Road Transport Research Advisory Council (ERTRAC). “Report on the h2020rtr conference”. (2018), [Online]. Available: <https://www.2zeroemission.eu/wp-content/uploads/2019/02/Report-on-H2020RTR18-final.pdf> (visited on 03/25/2024).
- [49] European Road Transport Research Advisory Council (ERTRAC). “Connected, co-operative and automated mobility roadmap”. (2022), [Online]. Available: <https://www.ertrac.org/wp-content/uploads/2022/07/ERTRAC-CCAM-Roadmap-V10.pdf> (visited on 03/25/2024).
- [50] European Transport Safety Council (ETSC), *The Role of Driver Fatigue in Commercial Road Transport Crashes*. Brussels, 2001, p. 33.
- [51] EuroNCAP. “Euroncap vision 2030, a safer future for mobility”. (2022), [Online]. Available: <https://cdn.euroncap.com/media/74468/euro-ncap-roadmap-vision-2030.pdf> (visited on 03/25/2024).
- [52] European Parliament, “Report on the eu road safety policy framework 2021-2030, recommendations on next steps towards ‘vision zero’ (2021/2014(ini))”, 2021.
- [53] European Parliament. Council of the European Union. “Regulation on type-approval requirements for motor vehicles and their trailers, and systems, components and separate technical units intended for such vehicles, as regards their general safety and the protection of vehicle occupants and vulnerable road users, (eu) 2019/2144”. (2019), [Online]. Available: <https://www.sae.org/news/2020/12/rise-of-sae-level-2> (visited on 03/25/2024).
- [54] M. Everingham, S. M. A. Eslami, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes challenge: A retrospective”, *International*

- Journal of Computer Vision*, vol. 111, pp. 98–136, 2015. DOI: [10.1007/s11263-014-0733-5](https://doi.org/10.1007/s11263-014-0733-5).
- [55] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (voc) challenge”, *International Journal of Computer Vision*, vol. 88, pp. 303–338, 2010. DOI: [10.1007/s11263-009-0275-4](https://doi.org/10.1007/s11263-009-0275-4).
 - [56] FAME Project. “Common evaluation methodology”. (Q1-2024), [Online]. Available: <https://www.connectedautomateddriving.eu/methodology/common-evaluation-methodology/> (visited on 03/18/2024).
 - [57] X. Fan, Y. Sun, B. Yin, and X. Guo, “Gabor-based dynamic representation for human fatigue monitoring in facial image sequences”, *Pattern Recognition Letters*, vol. 31, pp. 234–243, 2010. DOI: [10.1016/j.patrec.2009.08.014](https://doi.org/10.1016/j.patrec.2009.08.014).
 - [58] A. Fernández, R. Usamentiaga, J. Carús, and R. Casado, “Driver distraction using visual-based sensors and algorithms”, *Sensors*, vol. 16, p. 1805, 2016. DOI: [10.3390/s16111805](https://doi.org/10.3390/s16111805).
 - [59] ffmpeg.org. “FFMPEG tool”. (2023), [Online]. Available: <https://www.ffmpeg.org>.
 - [60] A. Fogelton and W. Benesova, “Eye blink detection based on motion vectors analysis”, *Computer Vision and Image Understanding*, vol. 148, pp. 23–33, 2016. DOI: [10.1016/j.cviu.2016.03.011](https://doi.org/10.1016/j.cviu.2016.03.011).
 - [61] Y. Freund and R. E. Schapire, “A short introduction to boosting”, *Journal of Japanese Society for Artificial Intelligence*, vol. 14, pp. 771–780, 1999. DOI: [10.5555/1624312.1624417](https://doi.org/10.5555/1624312.1624417).
 - [62] L. Fridman, D. E. Brown, M. Glazer, *et al.*, “Mit advanced vehicle technology study: Large-scale naturalistic driving study of driver behavior and interaction with automation”, *IEEE Access*, vol. 7, pp. 102 021–102 038, 2019. DOI: [10.1109/ACCESS.2019.2926040](https://doi.org/10.1109/ACCESS.2019.2926040).
 - [63] L. Fridman, J. Lee, B. Reimer, and T. Victor, “‘owl’ and ‘lizard’: Patterns of head pose and eye pose in driver gaze classification”, *IET Computer Vision*, vol. 10, pp. 308–314, 2016. DOI: [10.1049/iet-cvi.2015.0296](https://doi.org/10.1049/iet-cvi.2015.0296).
 - [64] L. Fridman, B. Reimer, B. Mehler, and W. T. Freeman, “Cognitive load estimation in the wild”, in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, ACM, 2018, pp. 1–9. DOI: [10.1145/3173574.3174226](https://doi.org/10.1145/3173574.3174226).
 - [65] L. Fridman, H. Toyoda, S. Seaman, *et al.*, “What can be predicted from six seconds of driver glances?”, in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, ACM, 2017, pp. 2805–2813. DOI: [10.1145/3025453.3025929](https://doi.org/10.1145/3025453.3025929).
 - [66] F. Friedrichs and B. Yang, “Camera-based drowsiness reference for driver state classification under real driving conditions”, in *IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2010, pp. 101–106. DOI: [10.1109/IVS.2010.5548039](https://doi.org/10.1109/IVS.2010.5548039).
 - [67] W. Fuhl, T. Santini, and E. Kasneci, “Fast & robust eyelid outline & aperture detection in real-world scenarios”, in *IEEE Winter Conference on Applications of Computer Vision*, 2017. DOI: [10.1109/WACV.2017.126](https://doi.org/10.1109/WACV.2017.126).
 - [68] Fujitsu. “FUJITSU Vehicle ICT FEELytm”. (2015), [Online]. Available: <https://www.fujitsu.com/global/about/resources/news/press-releases/2015/0119-02.html> (visited on 07/09/2023).

- [69] I. García, S. Bronte, L. M. Bergasa, J. Almazán, and J. J. Y. Torres, “Vision-based drowsiness detector for real driving conditions”, in *IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2012, pp. 618–623. DOI: [10.1109/IVS.2012.6232222](https://doi.org/10.1109/IVS.2012.6232222).
- [70] J. Garney. “An analysis of throughput characteristics of universal serial bus”. (1996), [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.163.7407> (visited on 03/25/2024).
- [71] M. Gavrilescu and N. Vizireanu, “Feedforward neural network-based architecture for predicting emotions from speech”, *Data*, vol. 4, p. 101, 2019. DOI: [10.3390/data4030101](https://doi.org/10.3390/data4030101).
- [72] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets robotics: The kitti dataset”, *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013. DOI: [10.1177/0278364913491297](https://doi.org/10.1177/0278364913491297).
- [73] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? the kitti vision benchmark suite”, in *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2012, pp. 3354–3361. DOI: [10.1109/CVPR.2012.6248074](https://doi.org/10.1109/CVPR.2012.6248074).
- [74] R. J. Gerrig, P. G. Zimbardo, A. Campbell, and S. Cumming, *Psychology and life*, 2nd Austr. Pearson, 2011.
- [75] R. Ghoddoosian, M. Galib, and V. Athitsos, “A realistic dataset and baseline temporal model for early drowsiness detection”, in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, vol. 2019-June, IEEE, 2019, pp. 178–187. DOI: [10.1109/CVPRW.2019.00027](https://doi.org/10.1109/CVPRW.2019.00027).
- [76] S. Gill. “Volvo and vtt develop new method for monitoring driver status”. (2008), [Online]. Available: <https://www.tu-auto.com/volvo-and-vtt-develop-new-method-for-monitoring-driver-status> (visited on 07/09/2023).
- [77] J. Goenetxea, L. Unzueta, U. Elordi, J. D. Ortega, and O. Otaegui, “Efficient monocular point-of-gaze estimation on multiple screens and 3D face tracking for driver behaviour analysis”, in *Proceedings of the 6th Driver Distraction and Inattention conference*, 2018, pp. 118–125.
- [78] D. González Ortega, F. J. Díaz Pernas, M. Antón-Rodríguez, M. Martínez-Zarzuela, and J. F. Díez Higuera, “Real-time vision-based eye state detection for driver alertness monitoring”, *Pattern Analysis and Applications*, vol. 16, no. 3, pp. 285–306, 2013. DOI: [10.1007/s10044-013-0331-0](https://doi.org/10.1007/s10044-013-0331-0).
- [79] C. Gou, Y. Wu, K. Wang, K. Wang, F.-Y. Wang, and Q. Ji, “A joint cascaded framework for simultaneous eye detection and eye state estimation”, *Pattern Recognition*, vol. 67, pp. 23–31, 2017. DOI: [10.1016/j.patcog.2017.01.023](https://doi.org/10.1016/j.patcog.2017.01.023).
- [80] I. Grishchenko, A. Ablavatski, Y. Kartynnik, K. Raveendran, and M. Grundmann, *Attention mesh: High-fidelity face mesh prediction in real-time*, 2020. arXiv: [2006.10962 \[cs.CV\]](https://arxiv.org/abs/2006.10962).
- [81] R. A. Guler, G. Trigeorgis, E. Antonakos, P. Snape, S. Zafeiriou, and I. Kokkinos, “Densereg: Fully convolutional dense shape regression in-the-wild”, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2017, pp. 2614–2623. DOI: [10.1109/CVPR.2017.280](https://doi.org/10.1109/CVPR.2017.280).
- [82] A. Halin, J. G. Verly, and M. van Droogenbroeck, “Survey and synthesis of state of the art in driver monitoring”, *Sensors*, vol. 21, no. 16, 2021. DOI: [10.3390/s21165558](https://doi.org/10.3390/s21165558).

- [83] W. Han, Y. Yang, G.-B. Huang, O. Sourina, F. Klanner, and C. Denk, “Driver drowsiness detection based on novel eye openness recognition method and unsupervised feature learning”, in *IEEE International Conference on Systems, Man, and Cybernetics*, IEEE, 2015, pp. 1470–1475. DOI: [10.1109/SMC.2015.260](https://doi.org/10.1109/SMC.2015.260).
- [84] P. A. Hancock, L. A. Simmons, L. Hashemi, H. D. Howarth, and T. A. Ranney, “The effects of in-vehicle distraction on driver response during a crucial driving maneuver”, *Transportation Human Factors*, vol. 1, pp. 295–309, 1999. DOI: [10.1207/sthf0104_1](https://doi.org/10.1207/sthf0104_1).
- [85] J. L. Harbluk, Y. I. Noy, P. L. Trbovich, and M. Eizenman, “An on-road assessment of cognitive distraction: Impacts on drivers’ visual behavior and braking performance”, *Accident Analysis and Prevention*, vol. 39, no. 2, pp. 372–379, 2007. DOI: [10.1016/j.aap.2006.08.013](https://doi.org/10.1016/j.aap.2006.08.013).
- [86] M. M. Hayhoe, “Advances in relating eye movements and cognition”, *Infancy*, vol. 6, no. 2, pp. 267–274, 2004. DOI: [10.1207/s15327078in0602_7](https://doi.org/10.1207/s15327078in0602_7).
- [87] M. A. Hearst, S. Dumais, E. Osuna, J. Platt, and B. Scholkopf, “Support vector machines”, *IEEE Intelligent Systems and their Applications*, vol. 13, pp. 18–28, 1998. DOI: [10.1109/5254.708428](https://doi.org/10.1109/5254.708428).
- [88] S. Hoch, M. Schweigert, F. Althoff, and G. Rigoll, “The bmw surf project: A contribution to the research on cognitive vehicles”, in *IEEE Intelligent Vehicles Symposium, Proceedings*, IEEE, 2007, pp. 692–697. DOI: [10.1109/ivs.2007.4290197](https://doi.org/10.1109/ivs.2007.4290197).
- [89] Honda. “Honda sensing elite system”. (2022), [Online]. Available: <https://global.honda/newsroom/news/2022/c221201eng.html> (visited on 05/17/2023).
- [90] J. A. Horne and L. A. Reyner, “Sleep related vehicle accidents”, *BMJ*, vol. 310, pp. 565–567, 1995. DOI: [10.1136/bmj.310.6979.565](https://doi.org/10.1136/bmj.310.6979.565).
- [91] J. A. Horne and L. A. Reyner, “Vehicle accidents related to sleep: A review”, *Occupational and Environmental Medicine*, vol. 56, pp. 289–294, 1999. DOI: [10.1136/oem.56.5.289](https://doi.org/10.1136/oem.56.5.289).
- [92] J. L. Houston, G. C. A. Zuidhof, L. Bergamini, *et al.*, “One thousand and one hours: Self-driving motion prediction dataset”, in *Conference on Robot Learning (CoRL)*, 2020. arXiv: [2006.14480 \[cs.CV\]](https://arxiv.org/abs/2006.14480).
- [93] A. G. Howard, M. Zhu, B. Chen, *et al.*, *Mobilenets: Efficient convolutional neural networks for mobile vision applications*, 2017. arXiv: [1704.04861 \[cs.CV\]](https://arxiv.org/abs/1704.04861).
- [94] H. Ichihashi, T. Shirai, K. Nagasaka, and T. Miyoshi, “Neuro-fuzzy id3: A method of inducing fuzzy decision trees with linear programming for maximizing entropy and an algebraic method for incremental learning”, *Fuzzy Sets and Systems*, vol. 81, no. 1, pp. 157–167, 1996. DOI: [10.1016/0165-0114\(95\)00247-2](https://doi.org/10.1016/0165-0114(95)00247-2).
- [95] M. Ingre, T. Åkerstedt, B. Peters, A. Anund, and G. Kecklund, “Subjective sleepiness, simulated driving performance and blink duration: Examining individual differences”, *Journal of Sleep Research*, vol. 15, no. 1, pp. 47–53, 2006. DOI: [10.1111/j.1365-2869.2006.00504.x](https://doi.org/10.1111/j.1365-2869.2006.00504.x).
- [96] Intel Corp. “Intel realsense sdk”. (2023), [Online]. Available: <https://intelrealsense.com/sdk-2/> (visited on 05/17/2023).
- [97] Intempora. “Rtmaps middleware”. (2023), [Online]. Available: <https://intempora.com/products/rtmaps/> (visited on 07/09/2023).
- [98] International Organization for Standardization (ISO). “Iso 26262-1:2018(en) road vehicles — functional safety — part 1: Vocabulary”. (Dec. 2018), [Online]. Available:

- <https://www.iso.org/obp/ui/en/#iso:std:iso:26262:-1:ed-2:v1:en> (visited on 03/18/2024).
- [99] S. Ishimaru, K. Kunze, K. Kise, *et al.*, “In the blink of an eye: Combining head motion and eye blink frequency for activity recognition with google glass”, in *5th Augmented Human International Conference*, Association for Computing Machinery, 2014. DOI: [10.1145/2582051.2582066](https://doi.org/10.1145/2582051.2582066).
- [100] M. Itoh, “Individual differences in effects of secondary cognitive activity during driving on temperature at the nose tip”, in *IEEE International Conference on Mechatronics and Automation (ICMA)*, IEEE, 2009, pp. 7–11. DOI: [10.1109/ICMA.2009.5246188](https://doi.org/10.1109/ICMA.2009.5246188).
- [101] M. Jabon, J. Bailenson, E. Pontikakis, L. Takayama, and C. Nass, “Facial-expression analysis for predicting unsafe driving behavior”, *IEEE Pervasive Computing*, vol. 10, no. 4, pp. 84–95, 2011. DOI: [10.1109/MPRV.2010.46](https://doi.org/10.1109/MPRV.2010.46).
- [102] C. Jacobé de Naurois, C. Bourdin, C. Bougard, and J. L. Vercher, “Adapting artificial neural networks to a specific driver enhances detection and prediction of drowsiness”, *Accident Analysis and Prevention*, vol. 121, pp. 118–128, 2018. DOI: [10.1016/j.aap.2018.08.017](https://doi.org/10.1016/j.aap.2018.08.017).
- [103] C. Jacobé de Naurois, C. Bourdin, A. Stratulat, E. Diaz, and J. L. Vercher, “Detection and prediction of driver drowsiness using artificial neural network models”, *Accident Analysis and Prevention*, vol. 126, pp. 95–104, 2019. DOI: [10.1016/j.aap.2017.11.038](https://doi.org/10.1016/j.aap.2017.11.038).
- [104] Jaguar Land Rover. “Jaguar land rover road safety research includes brain wave monitoring to improve driver concentration and reduce accidents”. (2015), [Online]. Available: <https://media.jaguar.com/news/2015/06/jaguar-land-rover-road-safety-research-includes-brain-wave-monitoring> (visited on 06/26/2023).
- [105] J. Janai, F. Güney, A. Behl, and A. Geiger, “Computer vision for autonomous vehicles: Problems, datasets and state of the art”, *Foundations and Trends in Computer Graphics and Vision*, vol. 12, no. 1-3, pp. 1–308, 2020. DOI: [10.1561/06000000079](https://doi.org/10.1561/06000000079).
- [106] Y. Ji, S. Wang, Y. Lu, J. Wei, and Y. Zhao, “Eye and mouth state detection algorithm based on contour feature extraction”, *Journal of Electronic Imaging*, vol. 27, no. 5, p. 1, 2018. DOI: [10.1117/1.jei.27.5.051205](https://doi.org/10.1117/1.jei.27.5.051205).
- [107] K. Kaida, M. Takahashi, T. Åkerstedt, *et al.*, “Validation of the karolinska sleepiness scale against performance and eeg variables”, *Clinical Neurophysiology*, vol. 117, no. 7, pp. 1574–1581, 2006. DOI: [10.1016/j.clinph.2006.03.011](https://doi.org/10.1016/j.clinph.2006.03.011).
- [108] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt, “Driver behavior analysis for safe driving: A survey”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3017–3032, 2015. DOI: [10.1109/TITS.2015.2462084](https://doi.org/10.1109/TITS.2015.2462084).
- [109] A. Kashevnik, I. Lashkov, and A. Gurtov, “Methodology and mobile application for driver behavior analysis and accident prevention”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2427–2436, 2020. DOI: [10.1109/TITS.2019.2918328](https://doi.org/10.1109/TITS.2019.2918328).
- [110] I. Kayadibi, G. E. Güraksin, U. Ergün, and N. Ö. Süzme, “An eye state recognition system using transfer learning: Alexnet-based deep convolutional neural network”, *International Journal of Computational Intelligence Systems*, vol. 15, 2022. DOI: [10.1007/s44196-022-00108-2](https://doi.org/10.1007/s44196-022-00108-2).

- [111] V. Kazemi and J. Sullivan, “One millisecond face alignment with an ensemble of regression trees”, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1867–1874. DOI: [10.1109/CVPR.2014.241](https://doi.org/10.1109/CVPR.2014.241).
- [112] M. Q. Khan and S. Lee, “Gaze and eye tracking: Techniques and applications in adas”, *Sensors*, vol. 19, no. 24, 2019. DOI: [10.3390/s19245540](https://doi.org/10.3390/s19245540).
- [113] K. W. Kim, H. G. Hong, G. P. Nam, and K. R. Park, “A study of deep cnn-based classification of open and closed eyes using a visible light camera sensor”, *Sensors*, vol. 17, no. 7, 2017. DOI: [10.3390/s17071534](https://doi.org/10.3390/s17071534).
- [114] W. Kim, W. S. Jung, and H. K. Choi, “Lightweight driver monitoring system based on multi-task mobilenets”, *Sensors*, vol. 19, no. 14, 2019. DOI: [10.3390/s19143200](https://doi.org/10.3390/s19143200).
- [115] D. E. King, “Dlib-ml: A machine learning toolkit”, *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009. DOI: [10.1145/1577069.1755843](https://doi.org/10.1145/1577069.1755843).
- [116] K. Kircher, C. Ahlstrom, and A. Kircher, “Comparison of two eye-gaze based real-time driver distraction detection algorithms in a small-scale field operational test”, in *5th International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*, 2009, pp. 16–23. DOI: [10.17077/drivingassessment.1297](https://doi.org/10.17077/drivingassessment.1297).
- [117] R. Knippling and P. Rau. “Perclos : A valid psychophysiological measure of alertness as assessed by psychomotor vigilance”. (1998), [Online]. Available: <https://rosap.ntl.bts.gov/view/dot/113> (visited on 03/25/2024).
- [118] N. Kose, O. Kopuklu, A. Unnervik, and G. Rigoll, “Real-time driver state monitoring using a cnn based spatio-temporal approach”, in *IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 3236–3242. DOI: [10.1109/ITSC.2019.8917460](https://doi.org/10.1109/ITSC.2019.8917460).
- [119] T. Kunderinger, N. Sofra, and A. Riener, “Assessment of the potential of wrist-worn wearable sensors for driver drowsiness detection”, *Sensors*, vol. 20, p. 1029, 2020. DOI: [10.3390/s20041029](https://doi.org/10.3390/s20041029).
- [120] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, “A survey of deep learning applications to autonomous vehicle control”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 712–733, 2021. DOI: [10.1109/TITS.2019.2962338](https://doi.org/10.1109/TITS.2019.2962338).
- [121] S. K. Lal and A. Craig, “A critical review of the psychophysiology of driver fatigue”, *Biological Psychology*, vol. 55, no. 3, pp. 173–194, 2001. DOI: [10.1016/S0301-0511\(00\)00085-5](https://doi.org/10.1016/S0301-0511(00)00085-5).
- [122] T. C. Lansdown, N. Brook-Carter, and T. Kersloot, “Distraction from multiple in-vehicle secondary tasks: Vehicle performance and mental workload implications”, *Ergonomics*, vol. 47, pp. 91–104, 2004. DOI: [10.1080/00140130310001629775](https://doi.org/10.1080/00140130310001629775).
- [123] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning”, *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. DOI: [10.1038/nature14539](https://doi.org/10.1038/nature14539).
- [124] J. D. Lee, K. L. Young, and M. A. Regan, “Defining driver distraction”, in *Driver Distraction: Theory, Effects, and Mitigation*, 2008, pp. 31–40. DOI: [10.1201/9781420007497.ch3](https://doi.org/10.1201/9781420007497.ch3).
- [125] K. W. Lee, H. S. Yoon, J. M. Song, and K. R. Park, “Convolutional neural network-based classification of driver’s emotion during aggressive and smooth driving using multi-modal camera sensors”, *Sensors*, vol. 18, no. 4, pp. 14–16, 2018. DOI: [10.3390/s18040957](https://doi.org/10.3390/s18040957).

- [126] Y. Liang and J. D. Lee, “Combining cognitive and visual distraction: Less than the sum of its parts”, *Accident Analysis and Prevention*, vol. 42, no. 3, pp. 881–890, 2010. DOI: [10.1016/j.aap.2009.05.001](https://doi.org/10.1016/j.aap.2009.05.001).
- [127] Y. Liang and J. D. Lee, “A hybrid bayesian network approach to detect driver cognitive distraction”, *Transportation Research Part C: Emerging Technologies*, vol. 38, pp. 146–155, 2014. DOI: [10.1016/j.trc.2013.10.004](https://doi.org/10.1016/j.trc.2013.10.004).
- [128] Y. Liang, M. L. Reyes, and J. D. Lee, “Real-time detection of driver cognitive distraction using support vector machines”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 340–350, 2007. DOI: [10.1109/TITS.2007.895298](https://doi.org/10.1109/TITS.2007.895298).
- [129] Y. Liao, S. E. Li, W. Wang, Y. Wang, G. Li, and B. Cheng, “Detection of driver cognitive distraction: A comparison study of stop-controlled intersection and speed-limited highway”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1628–1637, 2016. DOI: [10.1109/TITS.2015.2506602](https://doi.org/10.1109/TITS.2015.2506602).
- [130] T.-Y. Lin, M. Maire, S. Belongie, *et al.*, “Microsoft coco: Common objects in context”, in *European Conference on Computer Vision (ECCV)*. Springer International Publishing, 2014, pp. 740–755. DOI: [10.1007/978-3-319-10602-1_48](https://doi.org/10.1007/978-3-319-10602-1_48).
- [131] W. Liu, D. Anguelov, D. Erhan, *et al.*, “Ssd: Single shot multibox detector”, in *European Conference on Computer Vision (ECCV)*. 2016, vol. 32, pp. 21–37. DOI: [10.1007/978-3-319-46448-0_2](https://doi.org/10.1007/978-3-319-46448-0_2).
- [132] A. M. Malla, P. R. Davidson, P. J. Bones, R. Green, and R. D. Jones, “Automated video-based measurement of eye closure for detecting behavioral microsleep.”, in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, 2010, pp. 6741–4. DOI: [10.1109/IEMBS.2010.5626013](https://doi.org/10.1109/IEMBS.2010.5626013).
- [133] B. Mandal, L. Li, G. S. Wang, and J. Lin, “Towards detection of bus driver fatigue based on robust visual analysis of eye state”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 545–557, 2017. DOI: [10.1109/TITS.2016.2582900](https://doi.org/10.1109/TITS.2016.2582900).
- [134] M. Martin, J. Popp, M. Anneken, M. Voit, and R. Stiefelhagen, “Body pose and context information for driver secondary task detection”, in *IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 2015–2021. DOI: [10.1109/IVS.2018.8500523](https://doi.org/10.1109/IVS.2018.8500523).
- [135] M. Martin, A. Roitberg, M. Haurilet, *et al.*, “Drive & act: A multi-modal dataset for fine-grained driver behavior recognition in autonomous vehicles”, in *IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 2801–2810.
- [136] S. Martin, A. Tawari, and M. M. Trivedi, “Monitoring head dynamics for driver assistance systems: A multi-perspective approach”, in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2013, pp. 2286–2291. DOI: [10.1109/ITSC.2013.6728568](https://doi.org/10.1109/ITSC.2013.6728568).
- [137] Q. Massoz, T. Langohr, C. Francois, and J. G. Verly, “The ulg multimodality drowsiness database (called drozy) and examples of use”, in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016. DOI: [10.1109/WACV.2016.7477715](https://doi.org/10.1109/WACV.2016.7477715).
- [138] A. D. McDonald, T. K. Ferris, and T. A. Wiener, “Classification of driver distraction: A comprehensive analysis of feature generation, machine learning, and input measures”, *Human Factors*, vol. 62, no. 6, pp. 1019–1035, 2020. DOI: [10.1177/0018720819856454](https://doi.org/10.1177/0018720819856454).
- [139] V. Melnicuk, S. Birrell, E. Crundall, and P. Jennings, “Towards hybrid driver state monitoring: Review, future perspectives and the role of consumer electronics”, in *IEEE Intelligent Vehicle Symposium (IV)*, IEEE, 2016, pp. 1392–1397.

-
- [140] Mercedes-Benz. “Mercedes-benz drive pilot system”. (2022), [Online]. Available: <https://goo.by/pwBdn> (visited on 05/17/2023).
 - [141] Mitsubishi Electric Corporation. “Mitsubishi Electric Develops Machine learning Technology That Detects Cognitive Distractions in Drivers”. (2015), [Online]. Available: <https://goo.gl/mKyrpE> (visited on 07/09/2024).
 - [142] M. Miyaji, H. Kawanaka, and K. Oguri, “Driver’s cognitive distraction detection using physiological features by the AdaBoost”, in *IEEE Conference on Intelligent Transportation Systems, Proceedings (ITSC)*, IEEE, 2009, pp. 90–95. DOI: [10.1109/ITSC.2009.5309881](https://doi.org/10.1109/ITSC.2009.5309881).
 - [143] C. Miyajima and K. Takeda, “Driver-Behavior Modeling Using On-Road Driving Data: A new application for behavior signal processing”, *IEEE Signal Processing Magazine*, vol. 33, no. 6, pp. 14–21, 2016. DOI: [10.1109/MSP.2016.2602377](https://doi.org/10.1109/MSP.2016.2602377).
 - [144] N. Moslemi, M. Soryani, and R. Azmi, “Computer vision-based recognition of driver distraction: A review”, *Concurrency and Computation: Practice and Experience*, vol. 33, pp. 1–25, 2021. DOI: [10.1002/cpe.6475](https://doi.org/10.1002/cpe.6475).
 - [145] E. Murphy-Chutorian, S. Member, and M. M. Trivedi, “Head Pose Estimation in Computer Vision : A Survey”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 607–626, 2009.
 - [146] R. A. Naqvi, M. Arsalan, G. Batchuluun, H. S. Yoon, and K. R. Park, “Deep learning-based gaze detection system for automobile drivers using a NIR camera sensor”, *Sensors*, vol. 18, no. 2, 2018. DOI: [10.3390/s18020456](https://doi.org/10.3390/s18020456).
 - [147] M. Nieto, A. Cortés, O. Otaegui, J. Arróspide, and L. Salgado, “Real-time lane tracking using rao-blackwellized particle filter”, *Journal of Real-Time Image Processing*, vol. 11, no. 1, pp. 179–191, 2016. DOI: [10.1007/s11554-012-0315-0](https://doi.org/10.1007/s11554-012-0315-0).
 - [148] M. Nieto, O. Otaegui, G. Vélez, J. D. Ortega, and A. Cortés, “On creating vision-based advanced driver assistance systems”, *IET Intelligent Transport Systems*, vol. 9, no. 1, pp. 59–66, 2015. DOI: [10.1049/iet-its.2013.0167](https://doi.org/10.1049/iet-its.2013.0167).
 - [149] M. Nieto, O. Senderos, and O. Otaegui, “Boosting ai applications: Labeling format for complex datasets”, *SoftwareX*, vol. 13, p. 100653, 2021. DOI: [10.1016/j.softx.2020.100653](https://doi.org/10.1016/j.softx.2020.100653).
 - [150] T. Nilsson, T. M. Nelson, and D. Carlson, “Development of fatigue symptoms during simulated driving”, *Accident Analysis and Prevention*, vol. 29, no. 4 SPEC. ISS. Pp. 479–488, 1997. DOI: [10.1016/s0001-4575\(97\)00027-4](https://doi.org/10.1016/s0001-4575(97)00027-4).
 - [151] Y. Noguchi, K. Shimada, M. Ohsuga, Y. Kamakura, and Y. Inoue, “The assessment of driver’s arousal states from the classification of eye-blink patterns”, in *Engineering Psychology and Cognitive Ergonomics. EPCE. Lecture Notes in Computer Science*, D. Harris, Ed. Springer, Berlin, Heidelberg, 2009, pp. 414–423. DOI: [10.1007/978-3-642-02728-4_44](https://doi.org/10.1007/978-3-642-02728-4_44).
 - [152] E. Ntaryamira, C. Maxim, and L. Cucu-Grosjean, “Data consistency and temporal validity under the circular buffer communication paradigm”, in *Proceedings of the 2019 Research in Adaptive and Convergent Systems, RACS*, 2019, pp. 51–56. DOI: [10.1145/3338840.3355682](https://doi.org/10.1145/3338840.3355682).
 - [153] E. Ntaryamira, C. Maxim, and L. Cucu-grosjean, “Ensuring data freshness for periodic real-time tasks”, in *12th Junior Researcher Workshop on Real-Time Computing - JRWRTC*, 2018.

- [154] E. Ohn-Bar and M. M. Trivedi, “The power is in your hands: 3d analysis of hand gestures in naturalistic video”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 912–917, 2013. DOI: [10.1109/CVPRW.2013.134](https://doi.org/10.1109/CVPRW.2013.134).
- [155] E. Ohn-Bar and M. M. Trivedi, “Looking at humans in the age of self-driving and highly automated vehicles”, *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 90–104, 2016. DOI: [10.1109/TIV.2016.2571067](https://doi.org/10.1109/TIV.2016.2571067).
- [156] E. Ohn-Bar and M. M. Trivedi, “Are all objects equal? deep spatio-temporal importance prediction in driving videos”, *Pattern Recognition*, vol. 64, pp. 425–436, 2017. DOI: [10.1016/j.patcog.2016.08.029](https://doi.org/10.1016/j.patcog.2016.08.029).
- [157] C. Olariu, H. Assem, J. D. Ortega, and M. Nieto, “A cloud-based ai framework for machine learning orchestration: A ‘driving or not-driving’ case-study for self-driving cars”, in *IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2019, pp. 1715–1722. DOI: [10.1109/IVS.2019.8813870](https://doi.org/10.1109/IVS.2019.8813870).
- [158] C. Olariu, J. D. Ortega, and J. J. Yebes, “The role of cloud-computing in the development and application of adas”, in *European Signal Processing Conference*, 2018, pp. 1037–1041. DOI: [10.23919/EUSIPCO.2018.8553029](https://doi.org/10.23919/EUSIPCO.2018.8553029).
- [159] M. Omidyeganeh, S. Shirmohammadi, S. Abtahi, *et al.*, “Yawning detection using embedded smart cameras”, *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 3, pp. 570–582, 2016. DOI: [10.1109/TIM.2015.2507378](https://doi.org/10.1109/TIM.2015.2507378).
- [160] J. D. Ortega, P. Cañas, M. Nieto, O. Otaegui, and L. Salgado, “Open your eyes: Eyelid aperture estimation in driver monitoring systems”, in C. Klein, M. Helfert, K. Berns, and G. O., Eds. Springer, Cham, 2021, vol. 1475, pp. 165–189. DOI: [10.1007/978-3-030-89170-1_9](https://doi.org/10.1007/978-3-030-89170-1_9).
- [161] J. D. Ortega, P. N. Cañas, M. Nieto, O. Otaegui, and L. Salgado, “Challenges of large-scale multi-camera datasets for driver monitoring systems”, *Sensors*, vol. 22, no. 7, pp. 42–45, 2022. DOI: [10.3390/s22072554](https://doi.org/10.3390/s22072554).
- [162] J. D. Ortega, N. Kose, P. Cañas, *et al.*, “Dmd: A large-scale multi-modal driver monitoring dataset for attention and alertness analysis”, in *ECCV Workshops, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12538 LNCS, 2020, pp. 387–405. DOI: [10.1007/978-3-030-66823-5_23](https://doi.org/10.1007/978-3-030-66823-5_23).
- [163] J. D. Ortega, M. Nieto, P. Cañas, O. Otaegui, and L. Salgado, “A real-time software framework for driver monitoring systems: software architecture and use cases”, in *Conference on Real-Time Image Processing and Deep Learning*, International Society for Optics and Photonics, vol. 11736, SPIE, 2021, p. 13. DOI: [10.1117/12.2588712](https://doi.org/10.1117/12.2588712).
- [164] J. D. Ortega, M. Nieto, L. Salgado, and O. Otaegui, “User-adaptive eyelid aperture estimation for blink detection in driver monitoring systems”, in *6th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS)*, 2020, pp. 342–352. DOI: [10.5220/0009369003420352](https://doi.org/10.5220/0009369003420352).
- [165] J. Paxion, E. Galy, and C. Berthelon, “Mental workload and driving”, *Frontiers in Psychology*, vol. 5, pp. 1–11, 2014. DOI: [10.3389/fpsyg.2014.01344](https://doi.org/10.3389/fpsyg.2014.01344).
- [166] J. Pohl, W. M. Birk, and L. Westervall, “A driver-distraction-based lane-keeping assistance system”, *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 221, pp. 541–552, 2007. DOI: [10.1243/09596518JSCE218](https://doi.org/10.1243/09596518JSCE218).

- [167] H. Qin, J. Liu, and T. Hong, “An eye state identification method based on the embedded hidden markov model”, in *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, IEEE, 2012, pp. 255–260. DOI: [10.1109/ICVES.2012.6294293](https://doi.org/10.1109/ICVES.2012.6294293).
- [168] T. A. Ranney, E. Mazzae, R. Garrett, and M. J. Goodman, “Nhtsa driver distraction research: Past, present, and future”, *Driver distraction Internet forum*, p. 9, 2000.
- [169] E. M. Rantanen and J. H. Goldberg, “The effect of mental workload on the visual field size and shape”, *Ergonomics*, vol. 42, no. 6, pp. 816–834, 1999. DOI: [10.1080/001401399185315](https://doi.org/10.1080/001401399185315).
- [170] A. Rasouli and J. K. Tsotsos, “Autonomous vehicles that interact with pedestrians: A survey of theory and practice”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 900–918, 2020. DOI: [10.1109/TITS.2019.2901817](https://doi.org/10.1109/TITS.2019.2901817).
- [171] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection”, in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788. DOI: [10.1109/CVPR.2016.91](https://doi.org/10.1109/CVPR.2016.91).
- [172] M. A. Regan, C. Hallett, and C. P. Gordon, “Driver distraction and driver inattention: Definition, relationship and taxonomy”, *Accident Analysis and Prevention*, vol. 43, pp. 1771–1781, 2011. DOI: [10.1016/j.aap.2011.04.008](https://doi.org/10.1016/j.aap.2011.04.008).
- [173] M. A. Regan and D. L. Strayer, “Towards an understanding of driver inattention: Taxonomy and theory”, in *Annals of Advances in Automotive Medicine. Association for the Advancement of Automotive Medicine.*, vol. 58, 2014, pp. 5–14.
- [174] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017. DOI: [10.1109/TPAMI.2016.2577031](https://doi.org/10.1109/TPAMI.2016.2577031).
- [175] E. Rogado, J. L. García, R. Barea, L. M. Bergasa, and M. E. L. Guillén, “Driver fatigue detection system”, in *IEEE International Conference on Robotics and Biomimetics*, IEEE, 2008, pp. 1105–1110. DOI: [10.1109/ROBIO.2009.4913155](https://doi.org/10.1109/ROBIO.2009.4913155).
- [176] E. Roidl, S. F. Wilhelm, O. Michael, and H. Rainer, “Introducing a multivariate model for predicting driving performance: The role of driving anger and personal characteristics”, *Journal of Safety Research*, vol. 47, pp. 47–56, 2013.
- [177] ROS.org. “Rosbag format”. (2023), [Online]. Available: <http://wiki.ros.org/rosbag/Format> (visited on 04/26/2023).
- [178] S. Rostaminia, A. Mayberry, D. Ganesan, B. Marlin, and J. Gummeson, “Ilid: Low-power sensing of fatigue and drowsiness measures on a computational eyeglass”, in *ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, 2017, pp. 1–26. DOI: [10.1145/3090088](https://doi.org/10.1145/3090088).
- [179] M. Roth and D. M. Gavrila, “Dd-pose - a large-scale driver head pose benchmark”, in *IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 927–934. DOI: [10.1109/IVS.2019.8814103](https://doi.org/10.1109/IVS.2019.8814103).
- [180] Saab. “Saab Driver Attention Warning System”. (2007), [Online]. Available: <http://www.saabnet.com/tsn/press/071102.html> (visited on 05/03/2023).
- [181] SAE International. “’level 2+’: Making automated driving profitable, mainstream”. (2020), [Online]. Available: <https://www.sae.org/news/2020/12/rise-of-sae-level-2> (visited on 07/03/2023).

- [182] SAE International. “J3016 - Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles”. (2021), [Online]. Available: https://www.sae.org/standards/content/j3016_202104/.
- [183] A. Sahayadhas, K. Sundaraj, and M. Murugappan, “Detecting driver drowsiness based on sensors: A review”, *Sensors*, vol. 12, no. 12, pp. 16 937–16 953, 2012. DOI: [10.3390/s121216937](https://doi.org/10.3390/s121216937).
- [184] S. Sánchez-Carballido, O. Senderos, M. Nieto, and O. Otaegui, “Semi-automatic cloud-native video annotation for autonomous driving”, *Applied Sciences*, vol. 10, p. 4301, 2020. DOI: [10.3390/app10124301](https://doi.org/10.3390/app10124301).
- [185] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks”, in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4510–4520. DOI: [10.1109/CVPR.2018.00474](https://doi.org/10.1109/CVPR.2018.00474).
- [186] J. Schmidt, R. Laarousi, W. Stolzmann, and K. Karrer-Gauß, “Eye blink detection for different driver states in conditionally automated driving and manual driving using eeg and a driver camera”, *Behavior Research Methods*, vol. 50, no. 3, pp. 1088–1101, 2018. DOI: [10.3758/s13428-017-0928-0](https://doi.org/10.3758/s13428-017-0928-0).
- [187] A. Schwarz, M. Haurilet, M. Martinez, and R. Stiefelhagen, “Driveahead - a large-scale driver head pose dataset”, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 1165–1174. DOI: [10.1109/CVPRW.2017.155](https://doi.org/10.1109/CVPRW.2017.155).
- [188] Seeing Machines. “Automotive dms”. (2023), [Online]. Available: <https://seeingmachines.com/products/automotive/> (visited on 05/03/2023).
- [189] B. Seppelt, B. Reimer, L. Angell, and S. Seaman, “Considering the human across levels of automation: Implications for reliance”, in *9th International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*, 2017, pp. 228–234. DOI: [10.17077/drivingassessment.1640](https://doi.org/10.17077/drivingassessment.1640).
- [190] M. M. Shirazi and A. B. Rad, “Detection of intoxicated drivers using online system identification of steering behavior”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 4, pp. 1738–1747, 2014. DOI: [10.1109/TITS.2014.2307891](https://doi.org/10.1109/TITS.2014.2307891).
- [191] N. Shiwakoti, P. Stasinopoulos, and F. Fedele, “Investigating the state of connected and autonomous vehicles: A literature review”, *Transportation Research Procedia*, vol. 48, no. 2018, pp. 870–882, 2020. DOI: [10.1016/j.trpro.2020.08.101](https://doi.org/10.1016/j.trpro.2020.08.101).
- [192] Siemens. “Simcenter prescan software”. (2024), [Online]. Available: <https://plm.sw.siemens.com/en-US/simcenter/autonomous-vehicle-solutions/prescan/> (visited on 03/19/2024).
- [193] G. Sikander and S. Anwar, “Driver fatigue detection systems: A review”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2339–2352, 2019. DOI: [10.1109/TITS.2018.2868499](https://doi.org/10.1109/TITS.2018.2868499).
- [194] S. Silke. “Ford biometric system can track your breathing, heartbeat and sweaty palms, huffington post”. (2012), [Online]. Available: http://huffingtonpost.com/2012/06/27/ford-biometric-system-breathing-heartbeat_n_1631993.html (visited on 07/04/2023).
- [195] S. Singh, “Critical reasons for crashes investigated in the national motor vehicle crash causation survey”, National Highway Traffic Safety Administration (NHTSA), 2015, pp. 1–3. DOI: [10.1016/j.trf.2015.04.014](https://doi.org/10.1016/j.trf.2015.04.014).

- [196] S. Sivaraman and M. M. Trivedi, “Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013. DOI: [10.1109/TITS.2013.2266661](https://doi.org/10.1109/TITS.2013.2266661).
- [197] SmartEye. “Driver monitoring system”. (2023), [Online]. Available: <https://smarteeye.se/solutions/automotive/driver-monitoring-system> (visited on 05/03/2023).
- [198] F. Song, X. Tan, X. Liu, and S. Chen, “Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients”, *Pattern Recognition*, vol. 47, no. 9, pp. 2825–2838, 2014. DOI: [10.1016/j.patcog.2014.03.024](https://doi.org/10.1016/j.patcog.2014.03.024).
- [199] T. Soukupová and J. Cech, “Real-time eye blink detection using facial landmarks”, in *21th Computer Vision Winter Workshop*, 2016.
- [200] Statistica. “Passenger and commercial vehicle sales in china from 2010 to 2021”. (2022), [Online]. Available: <https://statista.com/statistics/233743/vehicle-sales-in-china/> (visited on 05/17/2023).
- [201] F. M. Sukno, S. K. Pavani, C. Butakoff, and A. F. Frangi, “Automatic assessment of eye blinking patterns through statistical shape models”, in *International Conference on Computer Vision Systems (ICVS)*, vol. 5815 LNCS, 2009, pp. 33–42. DOI: [10.1007/978-3-642-04667-4_4](https://doi.org/10.1007/978-3-642-04667-4_4).
- [202] P. Sun, H. Kretschmar, X. Dotiwalla, *et al.*, “Scalability in perception for autonomous driving: Waymo open dataset”, in *Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2443–2451. DOI: [10.1109/CVPR42600.2020.00252](https://doi.org/10.1109/CVPR42600.2020.00252).
- [203] M. Suzuki, N. Yamamoto, O. Yamamoto, T. Nakano, and S. Yamamoto, “Measurement of driver’s consciousness by image processing: A method for presuming driver’s drowsiness by eye-blinks coping with individual differences”, in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, IEEE, 2006, pp. 2891–2896. DOI: [10.1109/ICSMC.2006.385313](https://doi.org/10.1109/ICSMC.2006.385313).
- [204] A. Tawari, S. Sivaraman, M. M. Trivedi, T. Shannon, and M. Toppelhofer, “Looking-in and looking-out vision for urban intelligent assistance: Estimation of driver attentive state and dynamic surround for safe merging and braking”, in *IEEE Intelligent Vehicles Symposium (IV)*, 2014, pp. 115–120. DOI: [10.1109/IVS.2014.6856600](https://doi.org/10.1109/IVS.2014.6856600).
- [205] R. Terzi, S. Sagiroglu, and M. U. Demirezen, “Big data perspective for driver/driving behavior”, *IEEE Intelligent Transportation Systems Magazine*, vol. 12, no. 2, pp. 20–35, 2020. DOI: [10.1109/MITS.2018.2879220](https://doi.org/10.1109/MITS.2018.2879220).
- [206] Tesla. “Tesla autopilot system”. (2023), [Online]. Available: <https://www.tesla.com/support/autopilot> (visited on 05/17/2023).
- [207] Toyota Motor Corporation. “Toyota enhances pre-crash safety system with eye monitor”. (2008), [Online]. Available: <https://newsroom.toyota.eu/2008-toyota-enhances-pre-crash-safety-system-with-eye-monitor>.
- [208] M. M. Trivedi, T. Gandhi, and J. McCall, “Looking-in and looking-out of a vehicle: Computer-vision-based enhanced vehicle safety”, *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, pp. 108–120, 2007. DOI: [10.1109/TITS.2006.889442](https://doi.org/10.1109/TITS.2006.889442).
- [209] T. Urselmann, P. N. Cañas, J. D. Ortega, and M. Nieto, “Semi-automatic pipeline for large-scale dataset annotation task: A dmd application”, in *European Conference on Computer Vision (ECCV) Workshops 2022*, vol. 1, 2023, pp. 560–574. DOI: [10.1007/978-3-031-25075-0_38](https://doi.org/10.1007/978-3-031-25075-0_38).

- [210] VI-DAS Consortium. “Vi-das demonstration at the its european congress”. (2018), [Online]. Available: <http://www.vi-das.eu/new/vi-das-demonstration-its-european-congress> (visited on 07/03/2023).
- [211] P. Viola and M. J. Jones, “Robust Real-Time Face Detection”, *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004. DOI: [10.1023/B:VISI.0000013087.49260.fb](https://doi.org/10.1023/B:VISI.0000013087.49260.fb).
- [212] B. Visnic. “Hopeful but closer scrutiny for sae level 4 automation”. (2022), [Online]. Available: <https://www.sae.org/news/2022/10/the-autonomous-level-4-horizon> (visited on 06/17/2023).
- [213] Volvo Car Group. “Volvo Cars conducts research into driver sensors in order to create cars that get to know their drivers”. (2014), [Online]. Available: <https://www.media.volvocars.com/global/en-gb/media/pressreleases/140898/volvo-cars-conducts-research-into-driver-sensors-in-order-to-create-cars-that-get-to-know-their-driv> (visited on 06/26/2023).
- [214] S. Vora, A. Rangesh, and M. M. Trivedi, “On generalizing driver gaze zone estimation using convolutional neural networks”, in *IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2017, pp. 849–854. DOI: [10.1109/IVS.2017.7995822](https://doi.org/10.1109/IVS.2017.7995822).
- [215] D. de Waard, “The measurement of drivers’ mental workload”, Ph.D. dissertation, Groningen University, Traffic Research Center, 1996.
- [216] L. Wang, X. Ding, C. Fang, C. Liu, and K. Wang, “Eye blink detection based on eye contour extraction”, in *Image Processing: Algorithms and Systems VII*, J. T. Astola, K. O. Egiazarian, N. M. Nasrabadi, and S. A. Rizvi, Eds., vol. 7245, 2009. DOI: [10.1117/12.804916](https://doi.org/10.1117/12.804916).
- [217] W. Wang, C. Liu, and D. Zhao, “How much data are enough? a statistical approach with case study on longitudinal driving behavior”, *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 2, pp. 85–98, 2017. DOI: [10.1109/TIV.2017.2720459](https://doi.org/10.1109/TIV.2017.2720459).
- [218] C.-H. Weng, Y.-H. Lai, and S.-H. Lai, “Driver Drowsiness Detection via a Hierarchical Temporal Deep Belief Network”, in *Computer Vision - ACCV 2016 Workshops. Lecture Notes in Computer Science*, vol. 10118, 2017. DOI: [10.1007/978-3-319-54526-4](https://doi.org/10.1007/978-3-319-54526-4).
- [219] A. Wesley, D. Shastri, and I. Pavlidis, “A novel method to monitor driver’s distractions”, in *Conference on Human Factors in Computing Systems - Proceedings*, 2010, pp. 4273–4278. DOI: [10.1145/1753846.1754138](https://doi.org/10.1145/1753846.1754138).
- [220] B. Whitworth. “Mercedes launches attention assist”. (2009), [Online]. Available: <https://www.carmagazine.co.uk/car-news/industry-news/mercedes-benz/mercedes-launches-attention-assist/> (visited on 07/09/2023).
- [221] W. G. Wong. “Affectiva brings emotion to driver monitoring systems”. (2018), [Online]. Available: <https://www.electronicdesign.com/markets/automotive/article/21806288/affectiva-brings-emotion-to-driver-monitoring-systems> (visited on 05/03/2023).
- [222] F. Yang, X. Yu, J. Huang, P. Yang, and D. Metaxas, “Robust eyelid tracking for fatigue detection”, in *19th IEEE International Conference on Image Processing (ICIP)*, IEEE, Ed., 2012, pp. 1829–1832.
- [223] C.-W. You, N. D. Lane, F. Chen, *et al.*, “Carsafe app: Alerting drowsy and distracted drivers using dual cameras on smartphones”, in *11th Annual International Conference*

- on Mobile Systems, Applications, and Services*, Association for Computing Machinery, 2013, pp. 13–26. DOI: [10.1145/2462456.2465428](https://doi.org/10.1145/2462456.2465428).
- [224] K. Yuen, S. Martin, and M. M. Trivedi, “On looking at faces in an automobile: Issues, algorithms and evaluation on naturalistic driving dataset”, in *International Conference on Pattern Recognition (ICPR)*, 2016, pp. 2777–2782. DOI: [10.1109/ICPR.2016.7900056](https://doi.org/10.1109/ICPR.2016.7900056).
- [225] K. Yuen and M. M. Trivedi, “An occluded stacked hourglass approach to facial landmark localization and occlusion estimation”, *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 4, pp. 321–331, 2017. DOI: [10.1109/TIV.2017.2772889](https://doi.org/10.1109/TIV.2017.2772889).
- [226] K. Yuen and M. M. Trivedi, “Looking at hands in autonomous vehicles: A convnet approach using part affinity fields”, *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 3, pp. 361–371, 2019. DOI: [10.1109/tiv.2019.2955369](https://doi.org/10.1109/tiv.2019.2955369).
- [227] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, “Appearance-based gaze estimation in the wild”, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2015, pp. 4511–4520. DOI: [10.1109/CVPR.2015.7299081](https://doi.org/10.1109/CVPR.2015.7299081).
- [228] Y. J. Zhong, L. P. Du, K. Zhang, and X. H. Sun, “Localized energy study for analyzing driver fatigue state based on wavelet analysis”, in *International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR)*, vol. 4, IEEE, 2007, pp. 1843–1846. DOI: [10.1109/ICWAPR.2007.4421754](https://doi.org/10.1109/ICWAPR.2007.4421754).