D2.5: Sensor Platform and Models including V&V results from 3rd cycle	
Project Number:	690705
Classification **Deliverable No.:**	Public
Work Package(s): **Document Version:** **Issue Date:** **Document Timescale:** Start of the Document: Final version due:	WP2 Vs. 3.0 31.12.2018. Project Start Date: September 1, 2016 Month 26 Month 28
Compiled by:	F. Tango, CRF
Authors:	D. Käthner, DLR A. Giralt, CAF M. Eilers, HMT E. Fathiazar, HMT S. Suck, OFF P. P. Fouopi, DLR F. Tango, CRF M. Graf, ULM A. Knapp, BIT Z. Jakó, BIT
Technical Approval:	Fabio Tango, CRF
Issue Authorisation:	Andreas Lüdtke, OFF

DISTRIBUTION LIST		
Copy type[1]	Company and Location	Recipient
T	AutoMate Consortium	all AutoMate Partners

[1] Copy types: E=Email, C=Controlled copy (paper), D=electronic copy on Disk or other medium, T=Team site (Sharepoint)

RECORD OF REVISION		
Date	Status Description	Author
20.11.2018	Preparation of initial document	F. Tango (CRF)
14.12.2018	V2X communication – implementation part	A. Knapp, Z. Jakó
17.12.2018	V2X communication – test results	A. Knapp, Z. Jakó
19.12.2018	Contributions from partners	All involved partners
19.12.2018	Contributions for E2.1 and E3.1	M. Eilers, E. Fathiazar, S. Suck
20.12.2018	Integration of review comments	F. Tango (CRF)
27.12.2018	Final version ready	F. Tango (CRF)

Table of Contents

List of Figures **6**
List of Tables **10**
Executive Summary **11**
1 Introduction **12**
2 How the WP2 enablers contribute to the implementation **15**
3 Status of WP2 enablers in cycle 2 ..... **19**
3.1 E1.1 – Driver monitoring system with driver state model for distraction and drowsiness 19
3.1.1 Scenario and uses case where E1.1 is relevant 19
3.1.2 DMS Integration in VEDECOM Car 21
3.1.3 System Calibration 22
3.2 E1.2 – V2x communication 26
3.2.1 Scenario and uses case where E1.2 is relevant 26
3.2.2 Implementation 26
3.3 E2.1 – Driver Intention Recognition 32
3.3.1 Scenario and uses case where E2.1 is relevant 34
3.3.2 Driver Intention Recognition in rural road scenarios 35
3.3.3 Driver Intention Recognition in roundabout scenarios 55
3.4 E3.1 – Situation and vehicle model 62
3.4.1 Semantic enrichment of the situation model 62
3.4.2 Predicting the future evolution of the traffic scene 64
3.4.3 Concept 64
3.4.4 Improvements 67
3.4.5 Implementation 68
3.4.6 Environmental context information 69
3.5 Driving Task Model 72

3.5.1 Scenario and use cases where the driving task model is relevant 72
3.5.2 Improvements 72
3.5.3 Comparison with similar State-of-the-Art approaches..... 72
4 Validation of enablers.....76
4.1 E1.1 – Driver monitoring system with driver state model for distraction and drowsiness 76
4.1.1 Drowsiness experiment 76
4.1.2 Test protocol..... 77
4.1.3 Results..... 81
4.2 E1.2 – V2x communication 85
4.3 E2.1 – Driver Intention Recognition 91
4.3.1 Driver intention recognition in rural road scenarios 91
4.3.2 Driver intention recognition in roundabout scenarios..... 101
4.4 Driving Task Model Results 110
4.5 E3.1 – Situation and vehicle model 114
4.5.1 Predicting the future evolution of the traffic scene 114
4.5.2 Results: Prediction performance 117
4.5.3 Results: Runtime performance 121
4.5.4 Privacy..... 122
5 Conclusions and Outlook.....123

List of Figures

Figure 1: Project cycles, milestones and link between enablers (WP2, WP3 and WP4) and demonstrators (WP5 and WP6) 13
Figure 2: DMS visualisation interface of a VEDECOM driver looking at the smart phone on passenger seat. 20
Figure 3: DMS visualisation interface of a VEDECOM driver looking at the smart phone on his legs..... 21
Figure 4: Camera integrated in the Vedecom demo car..... 22
Figure 5: creation of the 3D world..... 23
Figure 6: ICP calibration. 24
Figure 7: calibration of "Driver world" to "Cockpit world". 25
Figure 8: ICP demo graphic user interface. 25
Figure 9: The architecture of extended system of Cohda MK5 unit using Raspberry PI 3B 29
Figure 10: The web user interface of RaspAP..... 30
Figure 11: The web based GUI of visualization framework. Elements: map (top left), instruments (bottom left), 3D visualization of proximity (right). 31
Figure 12: Conceptional graph structures of the initial BN (left) and the 2TBN (right) of the model for driver intention recognition on rural road during the second cycle. All shaded nodes are assumed to be observed during inference. Darker shaded nodes do not have a probability resp. density distribution associated within the model. 37
Figure 13: Conceptional graph structures of the initial BN (left) and the 2TBN (right) of the model for driver intention recognition on rural road during the third cycle..... 40
Figure 14: Assignment of roles to vehicles (dark) in the vicinity of the TeamMate vehicle (light) 43
Figure 15: Learned graph structure of the embedded Bayesian classifiers realizing $\boldsymbol{pDt|Dt-1,lRt,Ct}$ in $\boldsymbol{M1}$..... 47

Figure 16: Learned graph structures of the embedded Bayesian classifiers realizing $\boldsymbol{pOt|Ot-1,lLt,dRt,Ct}$ (left) and $\boldsymbol{pOt|Ot-1,lRt,dLt,Ct}$ (right) in $\boldsymbol{M1}$. 48
Figure 17: Learned graph structures of $\boldsymbol{pEt|Bt,llt}$ (left) and $\boldsymbol{pEt|Bt,lrt}$ (right) in $\boldsymbol{M1}$. 48
Figure 18: Learned graph structure of the embedded Bayesian classifiers realizing $\boldsymbol{pDt|Dt-1,lrt,Ct}$ in $\boldsymbol{M2}$ 49
Figure 19: Learned graph structures of the embedded Bayesian classifiers realizing $\boldsymbol{pOt|Ot-1,llt,drt,Ct}$ (left) and $\boldsymbol{pOt|Ot-1,lrt,dlt,Ct}$ (right) in $\boldsymbol{M2}$. . 49
Figure 20: Learned graph structures of $\boldsymbol{pEt|Bt,llt}$ (left) and $\boldsymbol{pEt|Bt,lrt}$ (right) in $\boldsymbol{M2}$. 50
Figure 21: Exemplary visualization of $\boldsymbol{polt|lRt,dLt,DANLt,EANLt} = \text{true} = \boldsymbol{pDANLt|olt,lRt,dLt,EANLt} = \text{true}\boldsymbol{o} \in \boldsymbol{OpDANLt|ot,lRt,dLt,EANLt} = \text{true}$ (i.e., assuming $\boldsymbol{POt}$ to be uniform). The dotted line represents $\boldsymbol{polt|lRt,dLt,DANLt,EANLt} = \text{true} > \boldsymbol{port|lRt,dLt,DANLt,EANLt} = \text{true}$, the dashed line represents $\boldsymbol{DANLt} = \boldsymbol{520m}$. 54
Figure 22: Schematic scheme of a roundabout. Contact point is marked with a red cross which depict the first entrance location of the TeamMate vehicle (shown in white) to the roundabout. The role of vehicles inside the roundabouts were defined relative to this point: AN (ahead next vehicle), BN (behind next vehicles) and BS (behind second vehicle) relative to the contact point. 57
Figure 23: Graph structure learned for big roundabouts using structure learning method. 61
Figure 24: Graph structure for small roundabout. 61
Figure 25: DMS video outputs (considering glasses as well). 83
Figure 26: Communication distance measured in Satory test track..... 86
Figure 27: Antenna placement and cars used for the field test in Óceánárok utca 87

Figure 28: Measured communication distances from the position of RSU (it is located in the same place): 634m northward (left) and 526m southward (right) 88
Figure 29: Communication distance measured in urban environment 89
Figure 30: Part of the visualization framework. The upper section shows the road works warning sign with some additional information. The lower section includes the instruments with speed limit sign in the road work area. 90
Figure 31: Binary confusion matrix. 93
Figure 32:Screenshot of an internal tool used for the annotation of experimental data. 95
Figure 33: Confusion matrix and corresponding metrics of interest for $\boldsymbol{M1}$, assuming automatic mode. 97
Figure 34: Confusion matrix and corresponding metrics of interest for $\boldsymbol{M2}$, assuming automatic mode. 98
Figure 35: Confusion matrix and corresponding metrics of interest for $\boldsymbol{M1}$, assuming manual mode. 98
Figure 36: Confusion matrix and corresponding metrics of interest for $\boldsymbol{M2}$, assuming manual mode. 99
Figure 37: Confusion matrix and corresponding metrics of interest for the model based on the traffic input from other vehicles in the small roundabouts. 107
Figure 38: Confusion matrix and corresponding metrics of interest for the model based on the traffic input of other vehicles in the small roundabouts and additional input speed input of the TeamMate vehicle. 107
Figure 39: Confusion matrix and corresponding metrics of interest for the model based on the traffic input from other vehicles in the big roundabouts. 108
Figure 40: Confusion matrix and corresponding metrics of interest for the model based on the traffic input from other vehicles in the big roundabouts and additional speed input of the TeamMate vehicle. 109

Figure 41: Basic model for the Peter scenario..... 113

List of Tables

Table 1: Role and relevance of the WP2 enablers for the cooperation 17
Table 2: Overview of the observable variables considered as causes $\boldsymbol{C}$ and effects $\boldsymbol{E}$ for intention recognition on rural roads during the third cycle. 42
Table 3: Summary of inputs from vehicles inside the roundabouts considered for modelling intention recognition in entering to the roundabouts. 57
Table 4: DMS performances with respect to the different areas inside the vehicle. 82
Table 5: Requirements and metrics used for the technical validation of E2.1 91
Table 6: Average computation time for the prediction of the temporal and spatial evolution for a single object for different number of vertices per corner polygon. 100
Table 7: Requirements and metrics used for the technical validation of E3.1 114
Table 8: Ratio of successes #$\boldsymbol{s}$ and the sum of successes #$\boldsymbol{s}$ and failures #$\boldsymbol{f}$ for the prediction of the temporal and spatial evolution for human participants, for different prediction horizons $\boldsymbol{i}$ (in seconds) and different levels of $\boldsymbol{\delta}$. Ratios limited to the location are shown in brackets, $\boldsymbol{AED}$ denotes the average Euclidean distance between the true location and the mean of the prediction. Bold values denote that the result is above the required $\mathbf{9\,010} - \boldsymbol{\delta}\%$. 118
Table 9: Approximated mean area $\boldsymbol{A\delta i}$ of the belief state over the location, for different prediction horizons $\boldsymbol{i}$ (in seconds) and different levels of $\boldsymbol{\delta}$, as the average of # measurements. 119
Table 10: Ratio of successes #$\boldsymbol{s}$ and the sum of successes #$\boldsymbol{s}$ and failures #$\boldsymbol{f}$ for automatically controlled traffic participants. 120
Table 11: Approximated mean area $\boldsymbol{A\delta i}$ for automatically controlled traffic participants. 121

Executive Summary

In the 3rd cycle, WP2, WP3 and WP4 are fed with the results of the evaluation process from the 2nd cycle, to deliver the final version of the enablers. Thus, the 3rd cycle ends with the evaluation of the final version of the demonstrators. This deliverable describes exactly the final state of the enablers developed in WP2, as well as the experiments conducted and proposed to technically validate them according to the validation plan and the requirements and metrics defined in D2.5 "Metrics & Experiments for V&V of the models in the 3rd cycle".

Now, these enablers are ready to be integrated in the demonstrators (both vehicles and driving simulators).

In particular, the enabler E1.1 (the DMS of CONTI partner) will be installed in the VED and CRF cars, as well as in the ULM driving simulator. Then, the enabler E1.2 (V2X communication from BIT partner) will be integrated into the VED vehicle. Also enablers E2.1 (Driver Intention Recognition – DIR in short – from OFF partner) and E3.1 (Situation and Vehicle Model, from OFF and DLR partners) are implemented in VED demo (and in ULM driving simulator as well). More details will be found in the deliverable of WP5 D5.6 "TeamMate Car Demonstrator after 3rd Cycle".

1 Introduction

The activities in the Automate project have been organized in 3 cycles to guarantee that the maturity of the technologies developed in the project is iteratively increased while assessing that the progresses are consistent with the needs of the demonstrators and, in turn, with the overall concept and objectives of the project.

As shown in Figure 1, the first 2 cycles are focused on the development and technical validation of the components (i.e. the enablers) performed in WP2, WP3 and WP4. The experience acquired in the 1st cycle (lesson learnt) has been used at the beginning of the 2nd cycle to review the requirements and metrics for the design and development of the enablers and, as a consequence, to improve them. As described in the document D2.4 "Sensor Platform and Models including V&V results from 2nd cycle", the enablers have been integrated into the demonstrators in WP5 (end of the 2nd cycle,) and the performances of the 1st version of the demonstrators are evaluated against their baseline in WP6.

Now, in the 3rd cycle, WP2, WP3 and WP4 are fed with the results of this evaluation process to deliver the final version of the enablers. Thus, the 3rd cycle ends with the evaluation of the final version of the demonstrators. This deliverable describes exactly the final state of the enablers developed in WP2, as well as the experiments conducted and proposed to technically validate them according to the validation plan and the requirements and metrics defined in D2.5 "Metrics & Experiments for V&V of the models in the 3rd cycle".

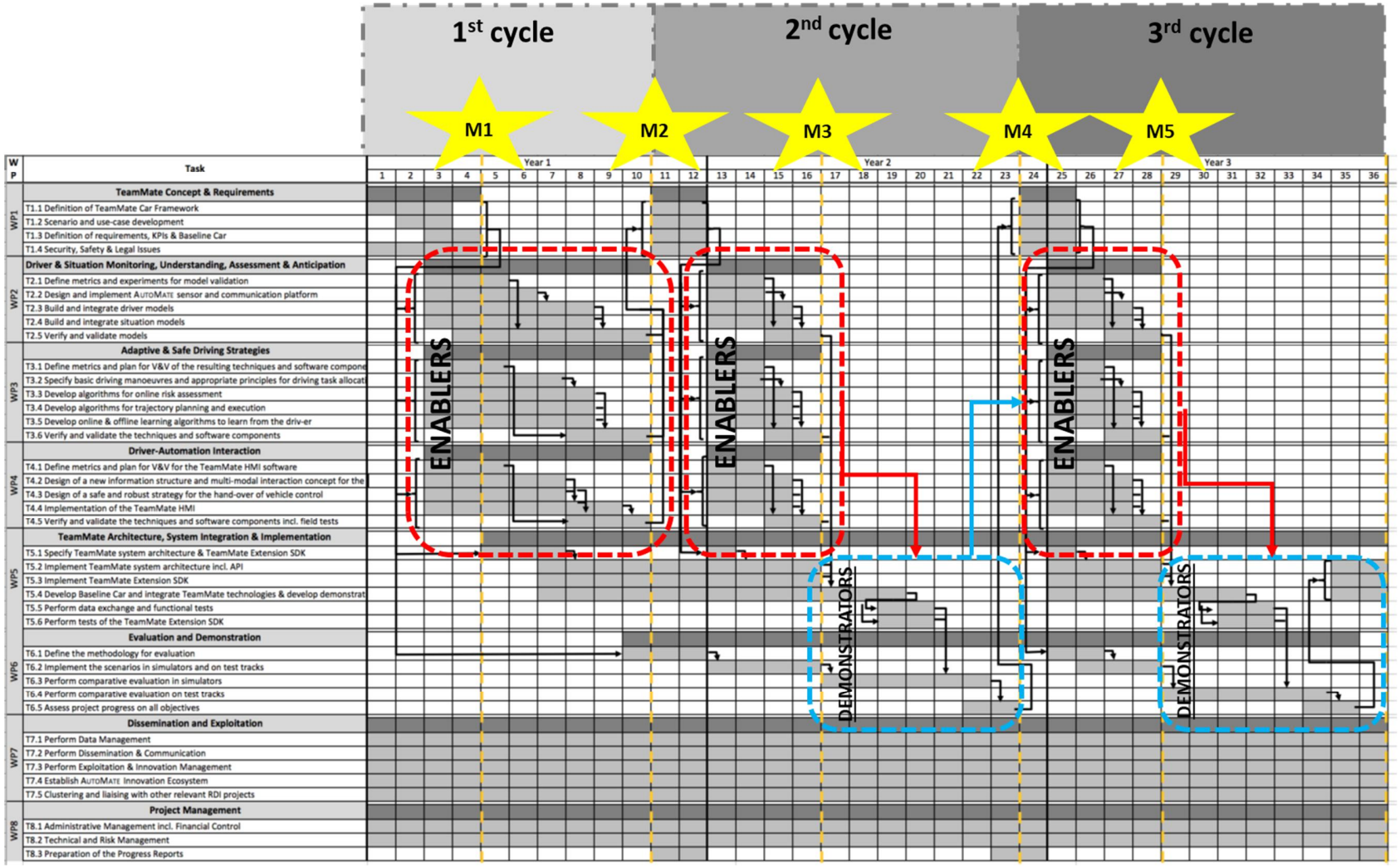


Figure 1: Project cycles, milestones and link between enablers (WP2, WP3 and WP4) and demonstrators (WP5 and WP6)

The development of all enablers follows the same process for WP2, WP3 and WP4 and, therefore, the deliverable D2.6, D3.7 and D4.6 have been structured with the same chapters to reflect this common (parallel) process.

The current document is structured as follows (with a similar structure of deliverable D2.4, where the development of the enablers in 2nd cycle is described). After this introduction, the general approach of the project regarding WP2 is described in Chapter 2. Then, the status of the enablers is presented in Chapter 3 including the latest improvements and their final developments. Next, Chapter 4 describes the validation of enablers along with validation methodologies and the results. Finally, Chapter 5 concludes the document.

2 How the WP2 enablers contribute to the implementation

The top-level objective of AutoMate is to develop, evaluate and demonstrate the "TeamMate Car" concept as a major enabler of highly automated vehicles. This concept consists of considering the driver and the automation as members of one team that understand and support each other in pursuing cooperatively the goal of driving safely, efficiently and comfortably from A to B. As a consequence, in order to show how the enablers contribute to the implementation of this concept, it is important to briefly explain why the cooperation is needed, and how the human and the automation can support each other to create a safe, efficient and comfortable driving experience.

As described in previous documents (see, for example, D2.4, D3.5 and D4.4), the scenarios and use cases selected to demonstrate the relevance of each enabler are representative and consistent with the direction of cooperation implemented by that enabler, as well as the modality of support (i.e. either in action or perception). Since the cooperation is implemented through the enablers developed in the project, Table 1 shows the role and relevance of each enabler in the cooperation.

The ultimate goal of D2.6 is to describe all results of T2.2-T2.5 at the end of the 3rd cycle. We remind here that: Task 2.2 "Design and implement AUTOMATE sensor and communication platform" aims at designing and implementing the AUTOMATE sensor and communication platform (Enabler 1); Task 2.3 "Build and integrate driver models" has the objective to develop the driver models (Enabler 2); finally, Task 2.4 "Build and integrate vehicle & situation models" aims at developing the situation models (Enabler 3) by combining data- and sensor-fusion (from T2.2) with probabilistic modelling techniques, to represent and estimate the spatial relations and physical states

of the vehicle and all objects in the environment.

Table 1: Role and relevance of the WP2 enablers for the cooperation

WP	ID	Enabler	Enabler Owner	Aim of the enabler	Direction of the support	
					Automation to Human	Human to Automation
WP2	**Enabler 1: Sensor and communication platform**					
	E1.1	Driver monitoring system with driver state model for distraction and drowsiness	CAF	Sensors and models for driver's visual distraction and drowsiness detection and classification	Enabler E1.1 is needed to implement a **support in perception** to complement the perception of the driver about the his/her state	
	E1.2	V2X communication	BIT	Allow the communication between the vehicle and everything.	Enabler E1.2 is needed to implement a **support in perception** to complement the perception of the driver about the environment	
	Enabler 2: Probabilistic Driver Modelling and Learning					
	E2.1	Driver intention recognition	OFF	Classify the current driver state, describe the interdependencies between the driver's state, type, behaviour and environment and predict the driver intention	Enabler E2.1 is needed to implement a **support in perception** to complement the perception of the driver about his/her state	

	Enabler 3: Probabilistic Vehicle and Situation Modelling					
	E3.1	Situation and vehicle model	DLR OFF	Estimate the dynamic vehicle and object state and position	Enabler E3.1 is needed to implement a **support in perception** to complement the perception of the driver about the situation and the vehicle	
	E3.2	Driving task Model	DLR	Define the driver's tasks to understand the expected behaviour (Paper Enabler)	Enabler E3.2 is needed to implement a **support in action along with E6.1 (Interaction Strategy)** to provide the driver with an effective means to interact with the automation in case of need.	Enabler E3.2 is needed to implement a **support in action along with E6.1 (Interaction Strategy)** to provide the driver with an effective means to answer and give feedback to the automation.

3 Status of WP2 enablers in cycle 2

This section describes the detailed results achieved during the 3rd cycle of the project. Modified assumptions and new approaches based on the knowledge acquired from the 1st and 2nd cycles are written along with the improvements and final state of the developments (and related results). Furthermore, the testing methodologies are presented, in order to validate the enablers.

3.1 E1.1 – Driver monitoring system with driver state model for distraction and drowsiness

This section describes the status of the enabler E1.1, that is the Driver Monitoring System (DMS), as provided by CAF partner.

3.1.1 Scenario and uses case where E1.1 is relevant

As shown in Table 1, Enabler E1.1 is needed to implement a support in perception from the automation to the human (A2H) to complement the perception of the driver about his/her state.

One of the use cases of MARTHA scenario has been revised to highlight and clarify the role of E1.1 to implement this cooperation.

Martha is driving in an extra-urban road in Manual Mode. She receives an incoming call: the car detects that she is distracted and this could lead to an unsafe behaviour. The TeamMate car offers a cooperation in action, suggesting a handover in order to shift to Automatic Mode. Martha accepts the suggestion and cedes the control.

The DMS will contribute to this scenario tested in the Vedecom demonstrator car by providing a Visual Distraction information through 2 output:

- eyes off the road
- ID of the instrument Martha is looking at

The Martha scenario was tested in the Vedecom demonstration car in static conditions. The driver had to look at the smart phone located on the passenger seat. He will take it, place it on his left leg and read the message.
The images below show first the DMS visual interface of the driver looking at the smart phone on the passenger seat, the second one shows the driver reading the message. We can see on the left part of the image that the correct areas are detected.

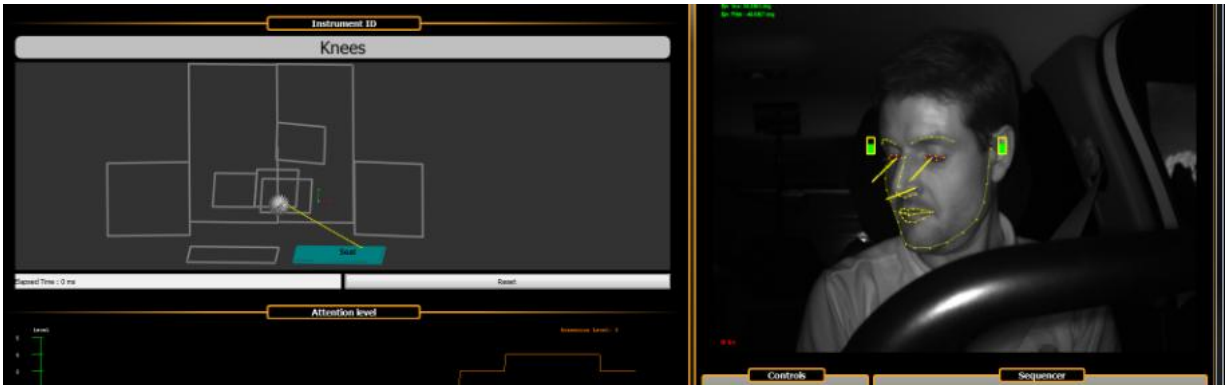


Figure 2: DMS visualisation interface of a VEDECOM driver looking at the smart phone on passenger seat.

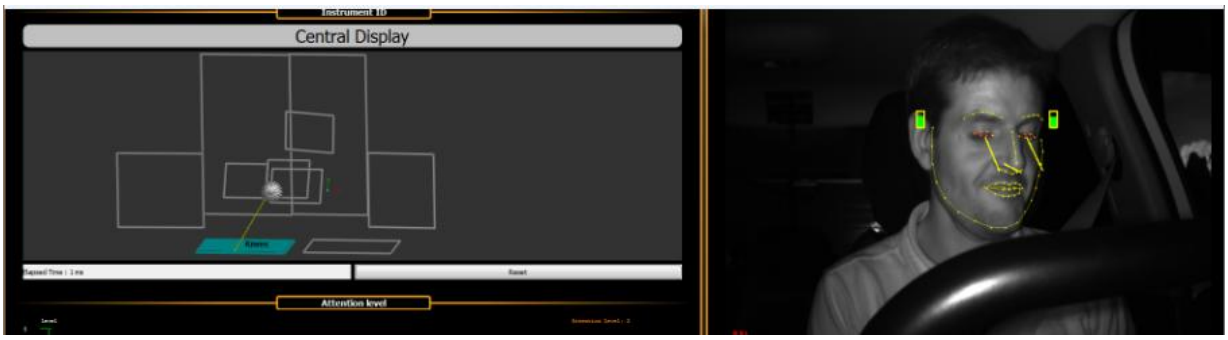


Figure 3: DMS visualisation interface of a VEDECOM driver looking at the smart phone on his legs.

3.1.2 DMS Integration in VEDECOM Car

Within the 2 first cycles the driver state system was validated in laboratory and in car condition considering an optimal position of the camera placed It is then necessary to perform validation tests of the generic scenarios and specific to the demonstration vehicle in the demonstration vehicle.

The nominal position of the camera is behind the steering looking at the driver face through the steering wheel. This optimal camera position couldn't be achieved in the Vedecom demonstration car because it would have been too intrusive by occluding some mandatory information displayed on vehicle screens.

The camera was placed in between the two frontal displays about 15 cm to the right and above of about 12 centimetres compared to the nominal position. Consequently, it is expected some performances degradation when the driver turns at the opposite of the camera position, that is left or downward.

The left image of Figure 4 shows the position of the camera (blue rectangle) integrated in the Vedecom demonstration car. The right image of Figure 1 shows the camera image and the tracking markers in green overlay.

Figure 4: Camera integrated in the Vedecom demo car

3.1.3 System Calibration

To obtain all the information concerning gazes, instruments observed, etc. It is necessary to first go through a calibration step which will be carried out in 3 steps:

- Creation of the 3D world in a "World Target Coordinate System"
- Extrinsic calibration of the ICP camera
- Calibration of the "Driver World" in the "Cockpit World"

3.1.3.1 Creation of the 3D World

The goal of this step is to create the 3D world of the cockpit in the coordinate system of a target "world".

To do this a Kinect is placed between the two front seats so you can see all the instruments we want to list. A calibration chessboard is placed at the Central Display (if possible with no angle value yaw, pitch and roll) to allow the algorithms to obtain an origin.

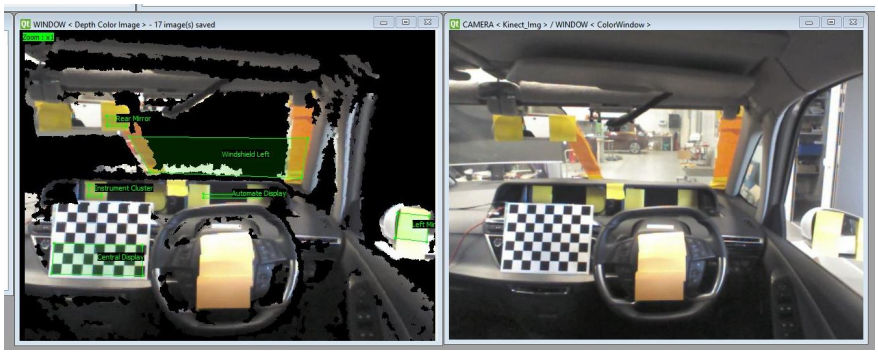


Figure 5: creation of the 3D world.

Once the recording is done, we must list the different instruments to create the 3D world.

The instrument of interest are tagged with yellow paper for providing accurate 3D data.

3.1.3.2 Extrinsic Calibration of the ICP Camera

To calibrate the ICP camera which is in our case located to the right of the steering wheel, we must place a small calibration chessboard at the location of the driver's head (this calibration chessboard must, like the previous one, have no angle yaw, pitch and roll).

Using the record made by this camera, the algorithm allows you to perform the calibration.

Figure 6: ICP calibration.

3.1.3.3 Calibration of the « Driver World » in the « Cockpit World »

After having obtained the various worlds which interests us, it is necessary to create the matrix of passage making it possible to pass from "Driver World" to "Cockpit World".

To do this, we now use a µEye camera to make a record to obtain the 2 previously placed targets, on both sides of the record.

Our algorithm then allows us to obtain the transition matrix.

Figure 7: calibration of "Driver world" to "Cockpit world".

Once done, we get a display like the one we can see in the next image.

In the left upper quadrant, the cockpit world is visualized. The instruments are represented by white rectangles which are filled with a green color when the estimated driver eye gaze intersect it.

Figure 8: ICP demo graphic user interface.

3.2 E1.2 – V2x communication

The Enabler E1.2, on V2X communication, is described in the following sections.

3.2.1 Scenario and uses case where E1.2 is relevant

V2x is essential part of future ITS systems. It directly improves the security of transportation, and it can improve the traffic flow and transport efficiency.
In AutoMate project, there are two scenarios in which V2X has a crucial role, even if in these scenarios the utilization of V2X differs.
All in all, the scenarios and use cases did not change with respect to cycle 2, except that V2V will be not demonstrated in EVA scenario
Thus, as reminder, in the MARTHA scenario V2I acts as an additional sensor, meaning that the TeamMate car is able to receive information about the environment (i.e. road works ahead), which would not be available in time for a safe cooperation. This is based on the simple information sharing concept of V2x.
The use case of MARTHA scenario has been revised to highlight and clarify the role of E1.2 to implement this cooperation. *Martha is driving in an extra-urban road in Manual Mode. She receives an incoming call: the car detects that she is distracted, and this could lead to an unsafe behaviour. The TeamMate car offers a cooperation in action, suggesting a handover in order to shift to Automatic Mode. Martha accepts the suggestion and cedes the control.*

3.2.2 Implementation

The 3rd cycle was about preparing the V2X equipment, i.e. Cohda MK5 units for field tests, conformance tests and deployment. That required to get familiar

with the device inner operations and the structure of the related Linux based applications and scripts. Using this knowledge we were able to customize its behaviour and set up them according to the needs of the different scenarios.
To achieve the high reliability that is required for an embedded device and to have automated, repeatable functions, different configuration files and scripts were developed during the 3rd cycle. A short overview of these are presented in this section.
As mentioned in Deliverable 2.4[2], in the previous cycle the specific standardized V2X messages were identified that are relevant for the AutoMate project scenarios. Furthermore, the compliance of the generated messages by the Cohda MK5 units were successfully tested. In this cycle, one of these specific messages, the Decentralized Environmental Message (DENM) was filled up with actual information instead of dummy data using configuration files. Different configuration files were prepared for each location of field tests and for the different test cases as well as for conformance testing.
The following scripts were developed for the V2X equipment to maintain high reliability, while its functions are customised:

- `rsu_denm-rww.sh` – automatically starts the V2X device in RSU mode after boot up to broadcast Road Works Warning (RWW) message with given configuration; the script is able to stop and/or restart this procedure according to the operator needs without restarting the device
- `obu_cam.sh` – same behaviour for Cooperative Awareness Message (CAM)

[2] AutoMate Deliverable 2.4: "Sensor Platform and Models including V&V results from 2nd cycle"

By some other settings and these scripts the field tests and deployment of RSU are basically solved, however, the devices are still black boxes. It is hard to access them physically (UTP/Ethernet cable is required) as well as complicated to enter in the operation system (configuring IP address and connection via SSH is also necessary). That is natural for an embedded system. To overcome these issues BIT extended the V2X device with an additional mini computer (with a Raspberry PI 3 Model B in this case) that has Ethernet and Wi-Fi interfaces as well. With proper applications and configuration the mini PC behaves as a wireless access point (AP), and provides limited access to the V2X device through a web based interface. This functionality simplifies the supervision of the Cohda MK5 unit, furthermore, remote access is possible (from a limited range of course, due to wireless nature) that allows to check the log files and adjust the configuration if necessary. For more complicated tasks an operator is still able to connect via SSH, but configuring the local network connections is unnecessary, since the AP has DHCP server too. This is also a nice advantage that makes the V2X device easier to use. The architecture of this extended system is depicted in Figure 9, while the web based interface is visible in Figure 10.

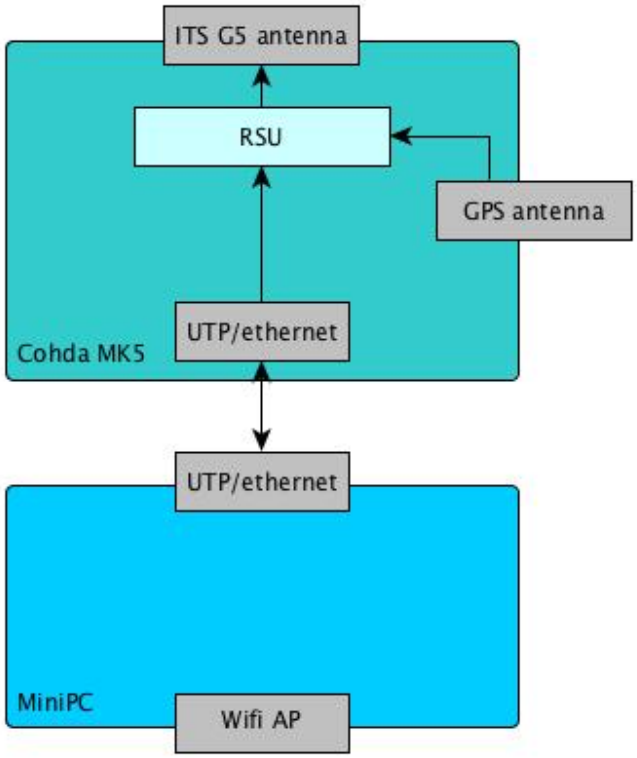


Figure 9: The architecture of extended system of Cohda MK5 unit using Raspberry PI 3B

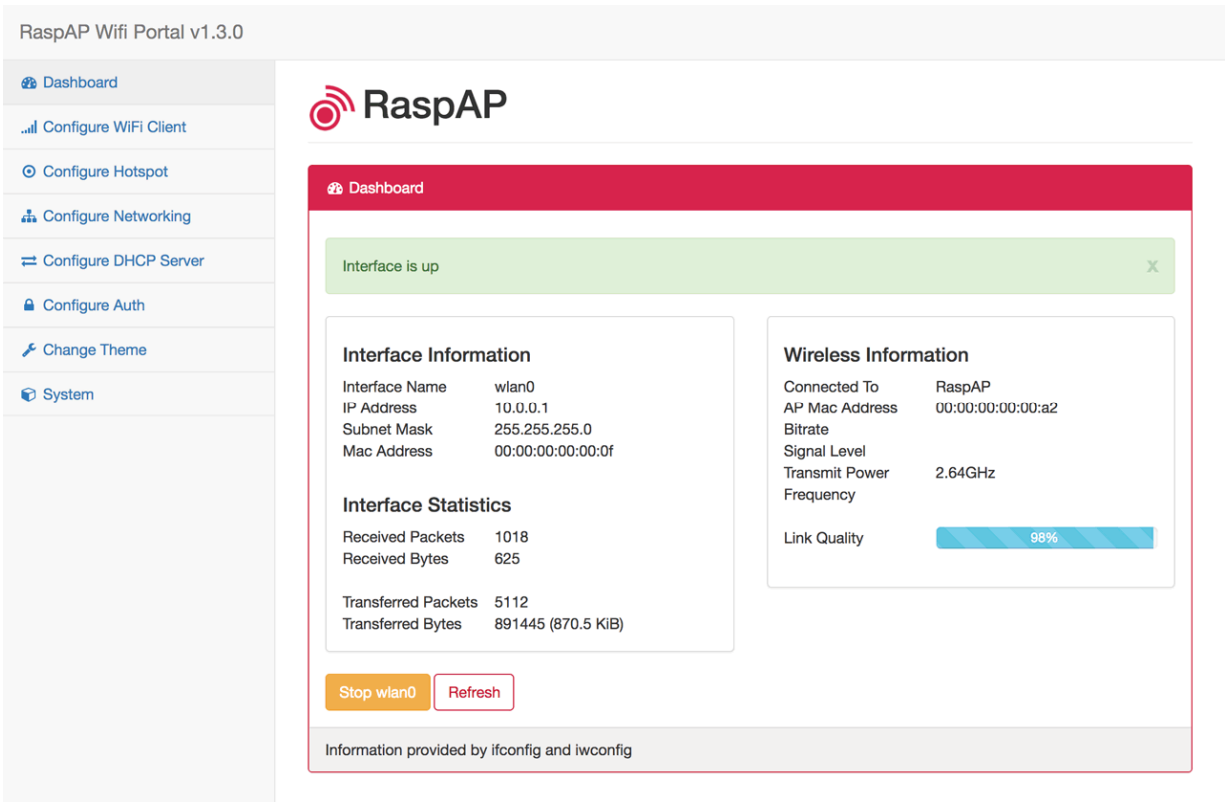


Figure 10: The web user interface of RaspAP[3]

Finally, BIT developed a web based visualization framework for the test results. The design of this framework started in the 2nd cycle, but the implementation was not finished, therefore it was not reported in D2.4. The visualization framework that is called Virtual HMI or VHMI depict some basic attributes of a given car (like speed, direction of movement etc.) using graphical elements. The parameters are originates from the GNSS based positioning system of the V2X device. Furthermore, a map is shown on the interface that illustrates nearby cars equipped with V2X unit and warns of road works in the proximity.

[3] https://github.com/billz/raspap-webgui

The significance of the web based interface is that these features are available from web browser, i.e. from a smart phone or tab as well without any specific application. The framework is able to work in real time by instantly visualizing the data coming from a vehicle as well as it is able to read log files stored in a specific format, and replay the recorded scenario. To transmit and log the GNSS data in specific format coming from the Cohda MK5 unit, the gateway application is used that was developed in the previous cycle and reported in D2.4. The VHMI was applied during the evaluation of the field tests and to validate the V2X equipment. Figure 11 shows graphical interface of the visualization framework.

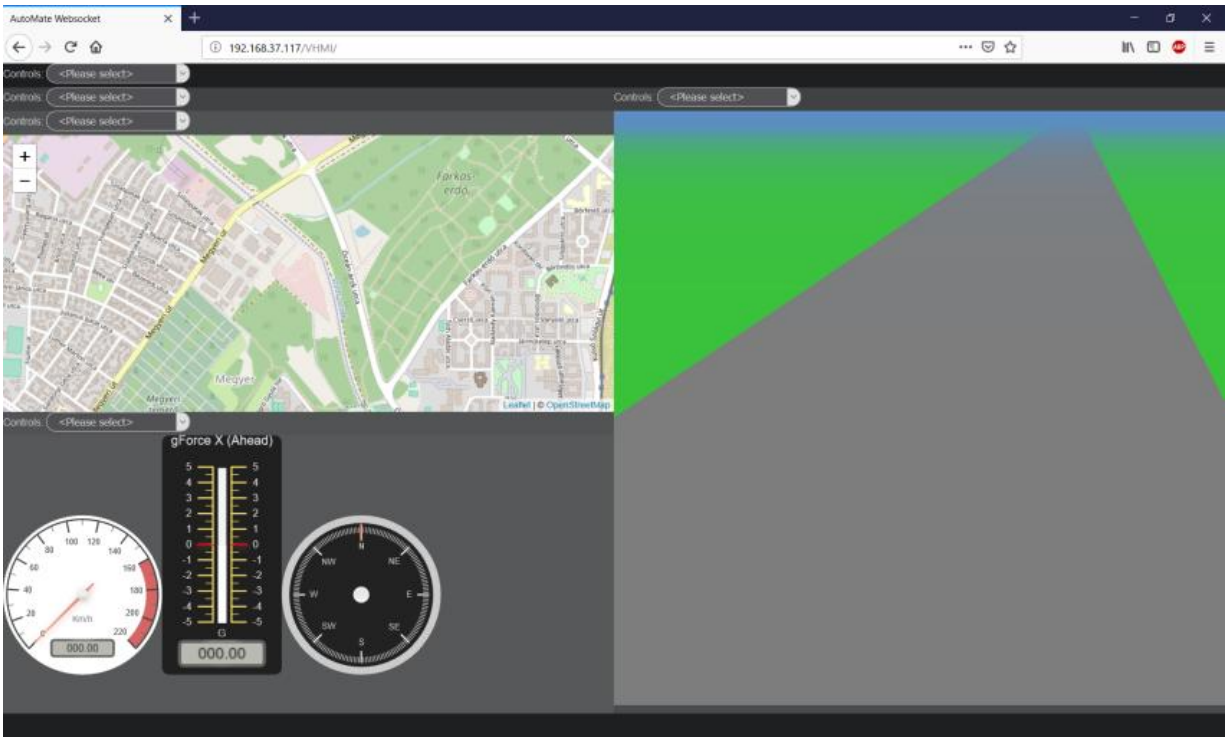


Figure 11: The web based GUI of visualization framework. Elements: map (top left), instruments (bottom left), 3D visualization of proximity (right).

3.3 E2.1 – Driver Intention Recognition

Driver intention recognition most commonly addresses the problem of anticipating driving manoeuvres a driver is likely to perform in the next few seconds. As early knowledge about such manoeuvre intentions may serve as a potential enabler to generate adaptive warnings and early interventions prior to the initiation of potential dangerous manoeuvres, driver intention recognition is of ever increasing importance for the development of advanced driver assistance systems and has become a popular research topic in recent years. Approaches reported in the literature (some comparative reviews are provided e.g., by Doshi and Trivedi [4] and Lefèvre et al. [5]) mainly differ in respect to the selected scenarios and addressed manoeuvres, modelling techniques used, and the sensor input considered.

In AutoMate, the development of models for driver intention recognition primarily focussed on the recognition of overtaking intentions on rural roads, akin to the Peter scenario. The purpose of intention recognition in the Peter scenario is to constantly provide the TeamMate vehicle with an online recognition of the current intentions of the driver (to be used for the cooperation). In addition, data are collected, to be used by E4.2 (Learning of intention from the driver) to learn when to trigger the overtaking in Automated

---

[4] Doshi, A. and M. M. Trivedi (2011), "Tactical Driver Behavior Prediction and Intent Inference: A Review", in *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems*, pp. 1892-1897.

[5] Lefèvre, S., D. Vasquez, and Ch. Laugier (2014), "A Survey on Motion Prediction and Risk Assessment for Intelligent Vehicles", in *Robomech Journal*, 1, 1, pp. 1-14.

Mode and to perform it in a human-like style. Within the TeamMate vehicle, the information provided can be used as follows:

- If the driver is in control of the TeamMate vehicle (manual driving), the information provided by the model can be used to assess the safety of the intended driving manoeuvre.
- If the automation is in control of the TeamMate vehicle (autonomous driving), the information provided by the model can serve as a mechanism to learn and trigger the most appropriate manoeuvres to the automation.

In addition to rural road scenarios, we extended the focus in the third cycle of AutoMate to cover a variant scenario (roundabouts) compared to overtaking scenario. The extension of the model was focused on Eva, dealing with intention recognition in entering to roundabouts.

For the development of the models for driver intention recognition in AutoMate, we started with a pre-existing framework, consisting of libraries and algorithms for the creation and utilization of (Dynamic) Bayesian Networks, originally developed during the former EU project HoliDes. Within AutoMate, this framework was significantly updated and extended, e.g., to allow for the learning and utilization of more complex model structures and parametric distributions, enabling the update of model parameters during runtime (as required by Enabler 4.2), and enabling the use in rural road and roundabout scenarios. To face the "cold start" problem during the first cycle of AutoMate, i.e., model development prior to the conduction of any data collection experiments, we made use of experimental data obtained during the former EU project HoliDes.

3.3.1 Scenario and uses case where E2.1 is relevant

The relevant use cases for the Peter scenario have already been reported in deliverable D2.4 "Sensor Platform and Models including V&V results from 2nd cycle" and remain valid. As an update, in the third cycle of AutoMate, the model was extended to cover a variant scenario (roundabouts) compared to overtaking scenario. The extension of the model was focused on Eva, dealing with intention recognition in entering to roundabouts.

In the Eva scenario, the TeamMate vehicle is approaching to a roundabout in automated mode. Due to a high traffic flow in the roundabout, the TeamMate vehicle hesitates to enter the roundabout and takes a long time to evaluate the situation and to enter the roundabout, resulting in frustration and reducing the acceptance of the automation.

To increase the efficiency in entering the roundabout, the TeamMate vehicle asks Eva to enter to the roundabout in the manual mode, and learns about the proper situation to enter the roundabout from Eva's manual driving. Later the TeamMate vehicle triggers entering to the roundabouts maneuver according to Eva's preferences.

In this use case, E2.1 provides a strategy to trigger the decisions of the automation, by introducing the right moment regarding to the traffic situation in the roundabout to enter to the roundabouts. Here, the intention recognition model learns the proper traffic situation to enter the roundabout from human perspective and triggers entrance maneuver regarding to the preference of a human driver.

Now, we consider two applicative areas for the DIR: the rural roads and the roundabouts.

3.3.2 Driver Intention Recognition in rural road scenarios

Next paragraphs describe the concept, the improvements and the implementation activities for the rural road scenario.

3.3.2.1 Concept

The general concept of the probabilistic model for driver intention recognition on rural roads (for the remainder of this section simply referred to as model) has been introduced in deliverable D2.4 "Sensor Platform and Models including V&V results from 2nd cycle". Based on the validation results obtained at the end of the second cycle, we made some adjustments to this concept for the third cycle. As such, we will shortly recap the concept of the driver intention recognition on rural roads.

The model is conceptualized a Dynamic Bayesian Network that represents the relations between the driver's intentions, driving manoeuvres resp. behaviors, and the situational context, as observable by the TeamMate vehicle's senor and communication platform.

For rural road scenarios with one lane in each direction, and assuming right-hand traffic, we considered three primary driving manoeuvres resp. behaviours: performing lane changes from the right to the left lane (LCL), from the left to the right lane (LCR), and general lane-keeping behaviour (LK), in the following represented by a discrete variable $B, Val(B) = \{b_{LCL}, b_{LCR}, b_{LK}\}$. Corresponding to these behaviours, we considered three potential intentions: the intention to change to the left lane (i.e., to initiate an overtaking), to return to the right lane (in order to complete an overtaking manoeuvre), and the absence of a lane change intention. To provide a more precise definition of intentions within the context of driver intention recognition in AutoMate, we

specify that *the presence of lane change intention implies that the driver is currently performing such lane change or will initiate such lane within the duration of one second*.

Although arguably less intuitive, for modelling purposes, it is more convenient to replace lane change intentions with *target lane intentions*, i.e., whether the driver intends to drive on the left or on the right lane, represented by a binary variable $I, Val(I) = \{i_L, i_R\}$. Correspondingly, let $L, Val(L) = \{l_L, l_R\}$ denote a binary variable that represents whether the TeamMate vehicle is located on the left or right lane. Following the definition of lane change intentions, we define that a discrepancy between a target lane intention and the lane in which the TeamMate vehicle in located, implies *that the driver will initiate a lane change to the target lane within the next second*. We will discuss the drawbacks of this definition in the next section.

Lastly, let $\boldsymbol{C}$ denote a set of discrete and continuous variables representing the observable *causes* for the formation of intentions and let $\boldsymbol{E}$ denote a set of discrete and continuous variables representing the observable *effects* of the intentions in terms of the resulting driving behaviour.

During the second cycle, the model was then based on the assumption that the temporal evolution of intentions and behaviours can be expressed as two hidden first-order Markov processes. More specifically we assumed that for any number of time steps $T \geq 1$, the conditional joint distribution $p(I^{1:T}, B^{1:T}, \boldsymbol{E}^{1:T} | L^{1:T}, \boldsymbol{C}^{1:T})$ can be factorized, according to the (conceptual) graph structure shown in Figure 12, as:

$$p(I^{1:T}, B^{1:T}, \boldsymbol{E}^{1:T} | L^{1:T}, \boldsymbol{C}^{1:T}) = p(I^{1:T} | L^{1:T}, \boldsymbol{C}^{1:T}) p(B^{1:T}, \boldsymbol{E}^{1:T} | I^{1:T}, L^{1:T})$$

$$= p(I^1 | L^1, \boldsymbol{C}^1) p(B^1 | I^1, L^1) p(\boldsymbol{E}^1 | B^1, L^1) \prod_{t=2}^{T} p(I^t | I^{t-1}, L^t, \boldsymbol{C}^t) p(B^t | B^{t-1}, I^t, L^t) p(\boldsymbol{E}^t | B^t, L^t).$$

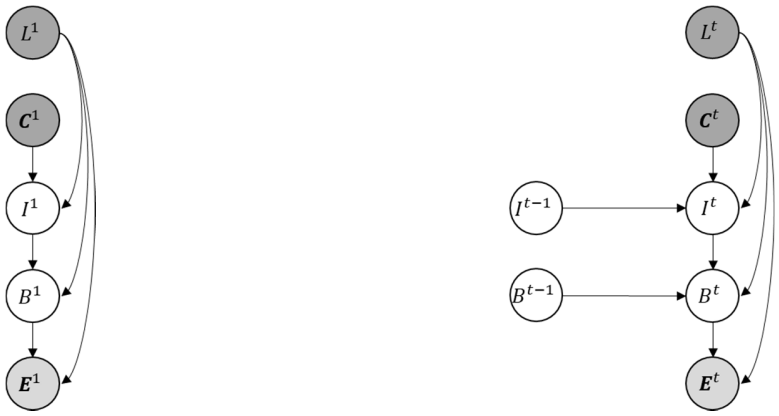


Figure 12: Conceptional graph structures of the initial BN (left) and the 2TBN (right) of the model for driver intention recognition on rural road during the second cycle. All shaded nodes are assumed to be observed during inference. Darker shaded nodes do not have a probability resp. density distribution associated within the model.

As such, the model was defined in terms of two components, a component for *intention recognition*, realized akin to a Maximum-entropy Markov Model, where for any number of time steps T the (conditional) joint distribution $p(I^{1:T}|L^{1:T}, \boldsymbol{C}^{1:T})$ is defined as

$$p(I^{1:T}|L^{1:T}, \boldsymbol{C}^{1:T}) = p(I^1|L^1, \boldsymbol{C}^1) \prod_{t=2}^{T} p(I^t|I^{t-1}, L^t, \boldsymbol{C}^t),$$

and a component for *behaviour recognition*, realized akin to an (input-dependent) Hidden Markov Model with factorized observation model, where for any number of time steps T the (conditional) joint probability distribution $p(B^{1:T}, \boldsymbol{E}^{1:T}|I^{1:T}, L^{1:T})$ is defined as

$$p(B^{1:T}, \boldsymbol{E}^{1:T}|I^{1:T}, L^{1:T}) = p(B^1|I^1, L^1)p(\boldsymbol{E}^1|B^1, L^1) \prod_{t=2}^{T} p(B^t|B^{t-1}, I^t, L^t)p(\boldsymbol{E}^t|B^t, L^t).$$

These components can be interpreted as follows: we assume that intentions evolve based on the situational context encountered. Intentions then manifest

themselves by the execution of driving manoeuvres whose effects can be observed.

3.3.2.2 Improvements

The proposed definition has the important drawback in that it does not allow to distinguish between the "clear" absence of a lane change intention, e.g., due to the absence of a lead vehicle, and situations in which the driver is closely following a lead vehicle, occasionally checking whether opposing traffic allows for a safe overtaking, a situation quite commonly encountered within the experimental data used during the second cycle.

To allow for such a distinction, for the third cycle, we extend the notion of intention by the concepts of *desire* and *opportunity* to perform a lane change. To provide a clear definition, we specify that:

1. The presence of a desire to change the lane implies that the driver desires to change the lane, but may lack the opportunity to do so, such that the presence of a desire does not imply the presence of an intention, as previously defined, and does not imply the initiation of a lane change.
2. The presence of an opportunity implies that the driver has decided that a given situation (in his subjective opinion) provides the opportunity to perform a lane change within the next second.

Keeping the original definition of intentions, we can then deterministically derive the presence of a lane change intention based on the presence of a desire and an opportunity. As with intentions, for modelling purposes, it is convenient to define desires and opportunities in terms of target lanes. As such, let the desire be represented by the binary variable $D, Val(D) = \{d_L, d_R\}$

and let the opportunity be represented by the binary variable $O, Val(O) = \{o_L, o_R\}$.

Incorporating these considerations to update the conceptual model structure for driver intention recognition on rural roads is achieved as follows. The component for behaviour recognition remains unchanged, while the component for intention recognition is replaced by a conceptual mechanisms of desires and opportunities, where for any number of time steps T the (conditional) joint distribution $p(D^{1:T}, O^{1:T}, I^{1:T} | L^{1:T}, \boldsymbol{C}^{1:T})$ is defined as:

$$p(D^{1:T}, O^{1:T}, I^{1:T} | L^{1:T}, \boldsymbol{C}^{1:T})$$
$$= p(D^1 | L^1, \boldsymbol{C}^1) p(O^1 | L^1, D^1, \boldsymbol{C}^1) p(I^1 | L^1, D^1, O^1) \prod_{t=2}^{T} p(D^t | D^{t-1}, L^t, \boldsymbol{C}^t) p(O^t | O^{t-1}, D^t, L^t, \boldsymbol{C}^t) p(I^t | L^t, D^t, O^t).$$

Given these conceptual considerations, we assume that for any number of time steps $T \geq 1$, the conditional joint distribution $p(D^{1:T}, O^{1:T}, I^{1:T}, B^{1:T}, \boldsymbol{E}^{1:T} | L^{1:T}, \boldsymbol{C}^{1:T})$ can be factorized, according to the (conditional) graph structure shown in Figure 13, as:

$$p(D^{1:T}, O^{1:T}, I^{1:T}, B^{1:T}, \boldsymbol{E}^{1:T} | L^{1:T}, \boldsymbol{C}^{1:T})$$
$$= p(D^{1:T} | L^{1:T}, \boldsymbol{C}^{1:T}) p(O^{1:T} | L^{1:T}, D^{1:T}, \boldsymbol{C}^{1:T}) p(I^{1:T} | L^{1:T}, D^{1:T}, O^{1:T}) p(B^{1:T}, \boldsymbol{E}^{1:T} | I^{1:T}, L^{1:T})$$
$$p(D^1 | L^1, \boldsymbol{C}^1) p(O^1 | L^1, D^1, \boldsymbol{C}^1) p(I^1 | L^1, D^1, O^1) p(B^1 | I^1, L^1) p(\boldsymbol{E}^1 | B^1, L^1)$$
$$= \prod_{t=2}^{T} p(D^t | D^{t-1}, L^t, \boldsymbol{C}^t) p(O^t | O^{t-1}, L^t, D^t, \boldsymbol{C}^t) p(I^t | L^t, D^t, O^t) p(B^t | B^{t-1}, I^t, L^t) p(\boldsymbol{E}^t | B^t, L^t)$$

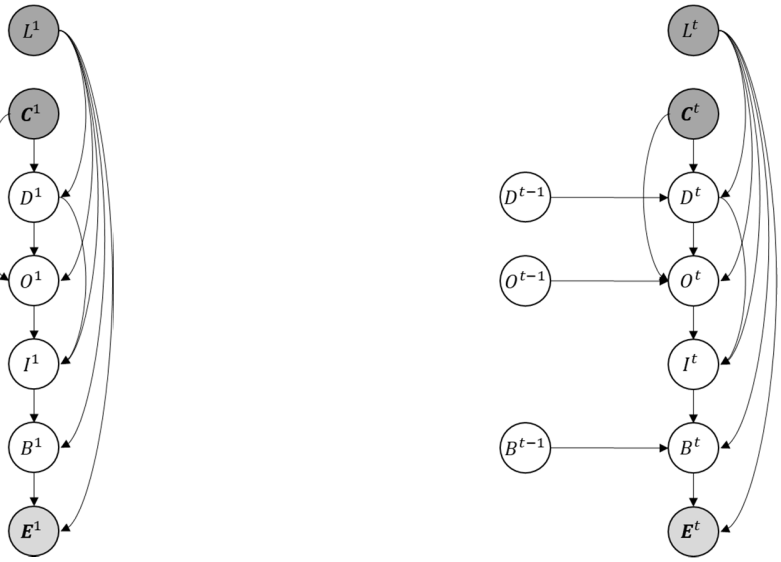


Figure 13: Conceptional graph structures of the initial BN (left) and the 2TBN (right) of the model for driver intention recognition on rural road during the third cycle.

When fully specified, the model can be utilized as follows: Let $\boldsymbol{X} = \{I, D, O, B\}$ denote the set over variables representing the current target lane intentions I, desires D, opportunities O, and manoeuvres B, during runtime, the model is used to at each time step t infer and maintain a joint belief state $p(\boldsymbol{X}^t | l^{1:t}, \boldsymbol{c}^{1:t}, \boldsymbol{e}^{1:t})$ given all available input obtained thus far, via recursive Bayesian filtering adapted to the structure of the model:

$$p(\boldsymbol{X}^t | l^{1:t}, \boldsymbol{c}^{1:t}, \boldsymbol{e}^{1:t}) \propto \sum_{i \in I} p(\boldsymbol{e}^t | B^t, l^t) \sum_{d \in D} \sum_{o \in O} \sum_{b \in B} p(\boldsymbol{X}^t | \boldsymbol{x}^{t-1}, l^t, \boldsymbol{c}^t, \boldsymbol{e}^t) p(\boldsymbol{x}^{t-1} | l^{1:t-1}, \boldsymbol{c}^{1:t-1}, \boldsymbol{e}^{1:t-1}),$$

where

$$p(\boldsymbol{X}^t | \boldsymbol{x}^{t-1}, l^t, \boldsymbol{c}^t, \boldsymbol{e}^t) = p(I^t | D^t, O^t, l^t) p(D^t | D^{t-1}, l^t, \boldsymbol{c}^t) p(O^t | O^{t-1}, l^t, D^t, \boldsymbol{c}^t) p(B^t | B^{t-1}, I^t, l^t).$$

From this joint belief state, separate belief states over intentions, $p(I^t | l^{1:t}, \boldsymbol{c}^{1:t}, \boldsymbol{e}^{1:t})$, desires, $p(D^t | l^{1:t}, \boldsymbol{c}^{1:t}, \boldsymbol{e}^{1:t})$, opportunities, $p(O^t | l^{1:t}, \boldsymbol{c}^{1:t}, \boldsymbol{e}^{1:t})$, and behaviours, $p(B^t | l^{1:t}, \boldsymbol{c}^{1:t}, \boldsymbol{e}^{1:t})$, can easily be derived via marginalization with the

distributions or derived measures, e.g., the most probable assignments, provided to other components of the TeamMate vehicle. If in autonomous mode, we can simply limit the model to the sub-component for intention recognition to derive $p(I^t|l^{1:t}, \boldsymbol{c}^{1:t})$, $p(D^t|l^{1:t}, \boldsymbol{c}^{1:t})$, and $p(O^t|l^{1:t}, \boldsymbol{c}^{1:t})$.

As provided, the model should be understood as conceptional, in that the parameters and finer structure, e.g., which variables constitute causes $\boldsymbol{C}$ and effects $\boldsymbol{E}$, must be provided based on prior expert knowledge and/or derived from multivariate time-series of human behaviour data via the use of machine-learning methods.

3.3.2.3 Implementation

For the development and validation of the probabilistic models for driver intention recognition on rural roads, OFF, ULM, and HMT conducted a dedicated simulator study in the OFF driving simulator, as described in detail in deliverable D2.5 "Metrics and Experiments for V&V of the driver, vehicle and situation models in the 3rd cycle". The experimental data obtained was split into a training set D_{Train}, including approx. 67.5% of the experimental data (2138134 samples or approx. 594 minutes), and a test set D_{Test}, including the remaining experimental data (1029216 samples or approx. 286 minutes). Within the third cycle, the training set has been used to learn the graph structures and parameters of two models M_1 and M_2, which were then subsequently validated on the test set D_{Test} (as described in section 4.3.1).

For potential causes and effects for intention recognition on rural roads, we focused on a subset of the totally available input obtained in the simulator experiment, for which we considered a set of 30 observable variables, as described in the next table.

Table 2: Overview of the observable variables considered as causes C and effects E for intention recognition on rural roads during the third cycle.

Variable	Type	Description
E_X	Binary	Represents, whether there exists a vehicle X in the traffic situation. $X \in \{BNL, BNR, ANL, ANR, ASL\}$.
T_X	Binary	Represents the type (PKW or LKW) of a vehicle X on the right lane. $X \in \{BNR, ANR\}$.
D_X	Continuous	Represents the distance between the TeamMate vehicle and a vehicle X along the course of the road. $X \in \{BNL, BNR, ANL, ANR, ASL\}$.
S_X	Continuous	Represents the difference between the longitudinal velocity of a vehicle X and the TeamMate vehicle, measured from the vehicle behind to the vehicle in front (in respect to the travelling direction of the TeamMate vehicle). $X \in \{BNL, BNR, ANL, ANR, ASL\}$.
W_X	Continuous	Represents the time headway between two vehicles, measured from the vehicle behind to the vehicle in front (in respect to the travelling direction of the TeamMate vehicle). $X \in \{BNL, BNR, ANL, ANR, ASL\}$
C_X	Continuous	Represents the inverse time to collision between to vehicles, measured from the vehicle behind to the vehicle in front (in respect to the travelling direction of the TeamMate vehicle). $X \in \{ANL, ANR\}$
V	Continuous	Represents a hypothetical viewing distance (up to a maximum of 700m) for the TeamMate vehicle, when located on the right lane.

F	Binary	Represents whether the TeamMate vehicle has a hypothetical free view, i.e. a viewing distance over 700m, when located on the right lane.
Y	Continuous	Represents the heading angle, resp. yaw angle of the TeamMate vehicle in respect to the course of the road.
P	Continuous	Represents the lateral position of the TeamMate vehicle within the road, measured as the deviation from the centreline.
A_{Lat}	Continuous	Represents the yaw rate of the TeamMate vehicle.
A_{Lon}	Continuous	Represents the acceleration of the TeamMate vehicle.

As apparent from the following figure and owed to the simulator scenario being limited to a single lead vehicle. We considered up to five vehicles in the vicinity of the TeamMate vehicle. The vehicles are assigned to fixed "roles" based on their relative positions to the TeamMate vehicles and each other**Errore. L'origine riferimento non è stata trovata.**. To improve the robustness to noise, we excluded the BNR vehicle for cases where the TeamMate vehicle was located on the right lane, and the BNL and ASL vehicles for cases where the TeamMate vehicle was located on the left lane.

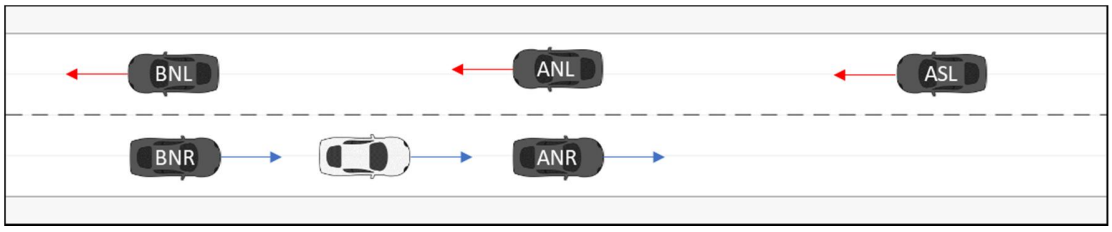


Figure 14: Assignment of roles to vehicles (dark) in the vicinity of the TeamMate vehicle (light)

Based on the results of the second cycle showing a preference for embedded Bayesian classifiers, we modelled the CPDs $p(D^t|D^{t-1}, L^t, \boldsymbol{C}^t)$, resp. $p(D^1|L^1, \boldsymbol{C}^1)$, and $p(O^t|O^{t-1}, L^t, \boldsymbol{C}^t)$, resp. $p(O^1|L^1, \boldsymbol{C}^1)$, in terms of embedded Bayesian classifiers, such that

$$p(D^t|d^{t-1}, l^t, \boldsymbol{c}^t) = \frac{1}{Z(d^{t-1}, l^t, \boldsymbol{c}^t)} p(D^t|d^{t-1}, l^t) p(\boldsymbol{c}^t|D^t, l^t)$$

and

$$p(O^t|o^{t-1}, l^t, d^t, \boldsymbol{c}^t) = \frac{1}{Z(o^{t-1}, l^t, d^t, \boldsymbol{c}^t)} p(O^t|o^{t-1}, l^t, d^t) p(\boldsymbol{c}^t|O^t, l^t, d^t)$$

with $Z(d^{t-1}, l^t, \boldsymbol{c}^t) = \sum_{d \in D} p(d^t|d^{t-1}, l^t) p(\boldsymbol{c}^t|d^t, l^t)$ resp. $Z(o^{t-1}, l^t, d^t, \boldsymbol{c}^t) = \sum_{o \in O} p(o^t|o^{t-1}, l^t, d^t) p(\boldsymbol{c}^t|o^t, l^t, d^t)$ representing normalization constants and $p(\boldsymbol{C}^t|D^t, L^t)$ resp. $p(\boldsymbol{C}^t|O^t, L^t, D^t)$ factorizing according to a Bayesian network structure. Lastly, the distribution $p(\boldsymbol{E}^t|B^t, L^t)$ was factorized based on a Bayesian network. Based on the intuition that information required for intention and behaviour recognition is strongly influenced by specific context, like e.g., the lane L^t, the TeamMate vehicle is located in, we allowed for context-specific independencies by considering distinct factorizations of $p(\boldsymbol{C}^t|D^t, L^t)$, $p(\boldsymbol{C}^t|O^t, L^t, D^t)$, and $p(\boldsymbol{E}^t|B^t, L^t)$, to be used in different context, e.g., when travelling on the left or on the right lane of the road.

We compared two different conceptual realizations for the factorizations of $p(\boldsymbol{C}^t|D^t, L^t)$, $p(\boldsymbol{C}^t|O^t, L^t, D^t)$, and $p(\boldsymbol{E}^t|B^t, L^t)$ resulting in two different models, in the following denoted as M_1 and M_2. For M_1, $p(\boldsymbol{C}^t|D^t, L^t)$, $p(\boldsymbol{C}^t|O^t, L^t, D^t)$, and $p(\boldsymbol{E}^t|B^t, L^t)$ were factorized in a way that continuous variables were conditionally independent given any discrete parents, akin to an (augmented) naïve Bayesian classifier. For M_2, $p(\boldsymbol{C}^t|D^t, L^t)$, $p(\boldsymbol{C}^t|O^t, L^t, D^t)$, and $p(\boldsymbol{E}^t|B^t, L^t)$ were factorized in a way that allowed the modelling of interdependencies between continuous variables, where we limited the number of connected continuous

variables to a maximum of two. For both M_1 and M_2, following the results during the second cycle, distributions over continuous variables were approximated by mixture of Gaussian distributions with the number of mixture component derived automatically during parameter estimation.

For deriving the exact graph structures and corresponding parameters of the different realizations for M_1 and M_2, we relied on machine-learning methods based on discriminative structure learning techniques using the training set D_{Train}. Depending on the resulting graph structure, observable variables that are not conditioned by any hidden variable will cancel out during the inference and were thus excluded from the corresponding graph structures.

For each M_1 and M_2, the following sub-models had to be learned:

- $p(\boldsymbol{D^t}|\boldsymbol{D^{t-1}}, \boldsymbol{l_R^t}, C^t)$: For recognizing the desire to change to the *left* lane when located on the *right* lane, we a priori limited the set of considered observable variables to the subset $\{E_{ANR}, T_{ANR}, D_{ANR}, S_{ANR}, W_{ANR}, V, F\}$ such as to force the recognition of the desires to be based on a potential lead vehicle. Figure 15 shows the learned graph structure, omitting the conditioning $\boldsymbol{l_R^t}$, for the embedded classifier realizing $\boldsymbol{p}(\boldsymbol{D^t}|\boldsymbol{D^{t-1}}, \boldsymbol{l_R^t}, C^t)$ for M_1, Figure 18 shows the learned graph structure for M_2.
- $p(\boldsymbol{D^t}|\boldsymbol{D^{t-1}}, \boldsymbol{l_L^t}, \boldsymbol{C^t})$: Under the assumption that, in rural road scenarios, a driver located on the *left* lane will return to the *right* lane as soon as the opportunity arises, we a priori defined $\boldsymbol{p}(\boldsymbol{D^t}|\boldsymbol{D^{t-1}}, \boldsymbol{l_L^t}, \boldsymbol{C^t}) = \boldsymbol{p}(\boldsymbol{D^t}|\boldsymbol{l_L^t})$, with $\boldsymbol{p}(d_R^t|\boldsymbol{l_L^t}) \cong 1.0$ and $\boldsymbol{p}(d_L^t|\boldsymbol{l_L^t}) \cong 0.0$.
- $p(O^t|O^{t-1}, \boldsymbol{l_R^t}, d_L^t, C^t)$: For recognizing the subjective opportunity to change to the *left* lane when located on the *right* lane, and having the desire to change to the *left* lane, we a priori limited the set of considered observable variables to the subset

$$\{E_{ANL}, D_{ANL}, S_{ANL}, W_{ANL}, C_{ANL}\} \cup \{E_{ASL}, D_{ASL}, S_{ASL}, W_{ASL}, C_{ASL}\}$$
$$\cup \{E_{ANR}, T_{ANR}, D_{ANR}, S_{ANR}, W_{ANR}\} \cup \{E_{BNL}, D_{BNL}, S_{BNL}, W_{BNL}\} \cup \{V, F\}.$$

Errore. L'origine riferimento non è stata trovata. (right) shows the learned graph structure, omitting the conditioning $\boldsymbol{l}_R^t$ and d_L^t, for the embedded classifier realizing $p(O^t|O^{t-1}, \boldsymbol{l}_R^t, d_L^t, \boldsymbol{C^t})$ for M_1, Figure 19 (right) shows the learned graph structure for M_2.

- $p(O^t|O^{t-1}, \boldsymbol{l}_L^t, d_R^t, C^t)$: For recognizing the subjective opportunity to change to the *right* lane when located on the *left* lane, and having the desire to change to the *right* lane, we a priori limited the set of considered observable variables to the subset

$$\{E_{ANL}, D_{ANL}, S_{ANL}, W_{ANL}, C_{ANL}\} \cup \{E_{ANR}, T_{ANR}, D_{ANR}, S_{ANR}, W_{ANR}\}$$
$$\cup \{E_{BNR}, T_{BNR}, D_{BNR}, S_{BNR}, W_{BNR}\} \cup \{V, F\}.$$

Errore. L'origine riferimento non è stata trovata. (left) shows the learned graph structure, omitting the conditioning $\boldsymbol{l}_L^t$ and d_R^t, for the embedded classifier realizing $p(O^t|O^{t-1}, \boldsymbol{l}_R^t, d_L^t, \boldsymbol{C^t})$ for M_1, Figure 19 (left) shows the learned graph structure for M_2.

- $p(O^t|O^{t-1}, \boldsymbol{l}_R^t, d_R^t, C^t)$: For the situation of recognizing the subjective opportunity to change to the *left* lane when located on the *right* lane, and having the desire to stay to the *right* lane, we had no annotations and therefore a priori defined $p(O^t|O^{t-1}, \boldsymbol{l}_R^t, d_R^t, C^t) = \boldsymbol{p}(O^t|\boldsymbol{l}_R^t)$, with $\boldsymbol{p}(o_R^t|\boldsymbol{l}_R^t) \cong 1.0$ and $\boldsymbol{p}(o_L^t|\boldsymbol{l}_R^t) \cong 0.0$.
- $p(O^t|O^{t-1}, \boldsymbol{l}_L^t, d_L^t, C^t)$: For the situation of recognizing the subjective opportunity to change to the *right* lane when located on the *left* lane, and having the desire to stay to the *left* lane, based on our assumption that a driver will always want to return to the right lane, we had no data

at all and therefore a priori defined $p(O^t|O^{t-1}, \boldsymbol{l}_L^t, d_L^t, C^t) = \boldsymbol{p}(O^t|\boldsymbol{l}_L^t)$, with $\boldsymbol{p}(o_L^t|\boldsymbol{l}_L^t) = \boldsymbol{p}(o_R^t|\boldsymbol{l}_L^t) = 0.5$.

- $p(\boldsymbol{E}^t|B^t, l_R^t)$: For factorizing $p(\boldsymbol{E}^t|B^t, l_R^t)$, we considered $\boldsymbol{E} = \{A_{Lat}, A_{Lon}, F, V, P, Y\}$. Figure 17 (right) shows the learned graph structure, omitting the conditioning $\boldsymbol{l}_R^t$ for M_1, Figure 20 (right) shows the learned graph structure for M_2.
- $p(\boldsymbol{E}^t|B^t, l_L^t)$: Given that F and V were only defined when the TeamMate vehicle was located on the right lane, we a priori defined $p(\boldsymbol{E}^t|B^t, l_L^t) = p(F^t, V^t|l_L^t)p(A_{Lat}, A_{Lon}, P, Y|B^t, l_L^t)$, with $p(F^t, V^t|l_L^t)$ cancelling during inference, and attempted to learn the remaining graph structure for $p(A_{Lat}, A_{Lon}, P, Y|B^t, l_L^t)$. Figure 17 (left) shows the learned graph structure, omitting the conditioning $\boldsymbol{l}_R^t$ for M_1, Figure 20 (left) shows the learned graph structure for M_2.

Together, they complete the conceptual graph structure of the probabilistic models M_1 and M_2 for intention recognition on rural roads during the third cycle.

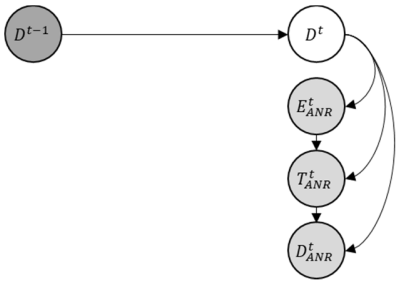


Figure 15: Learned graph structure of the embedded Bayesian classifiers realizing $\boldsymbol{p(D^t|D^{t-1}, l_R^t, C^t)}$ in $\boldsymbol{M_1}$.

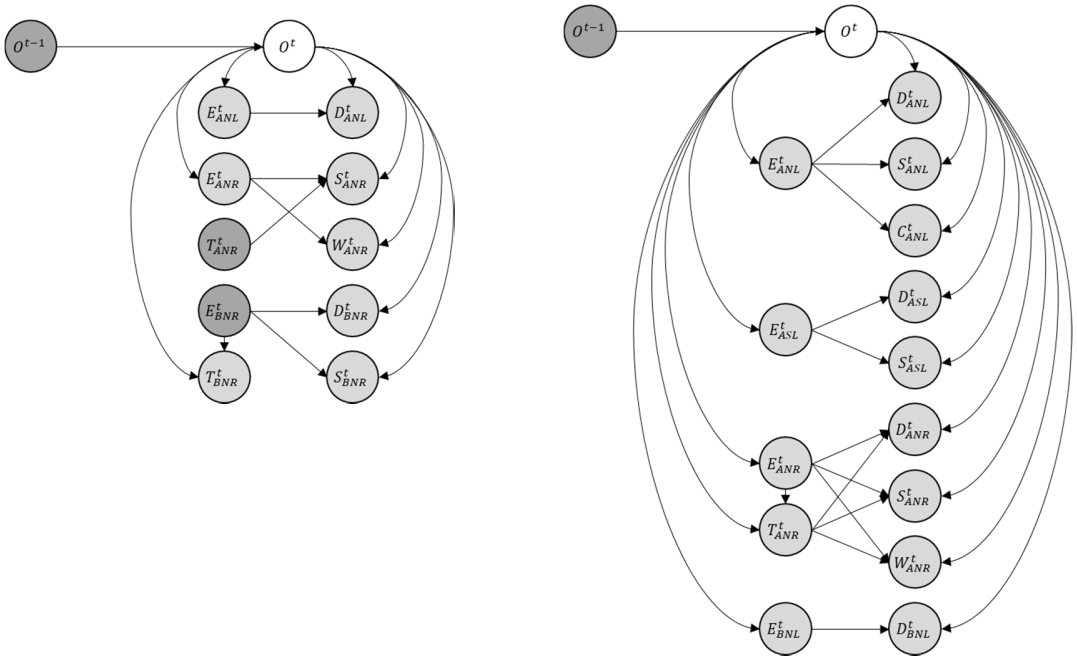


Figure 16: Learned graph structures of the embedded Bayesian classifiers realizing $p(O^t|O^{t-1}, l_L^t, d_R^t, C^t)$ (left) and $p(O^t|O^{t-1}, l_R^t, d_L^t, C^t)$ (right) in M_1.

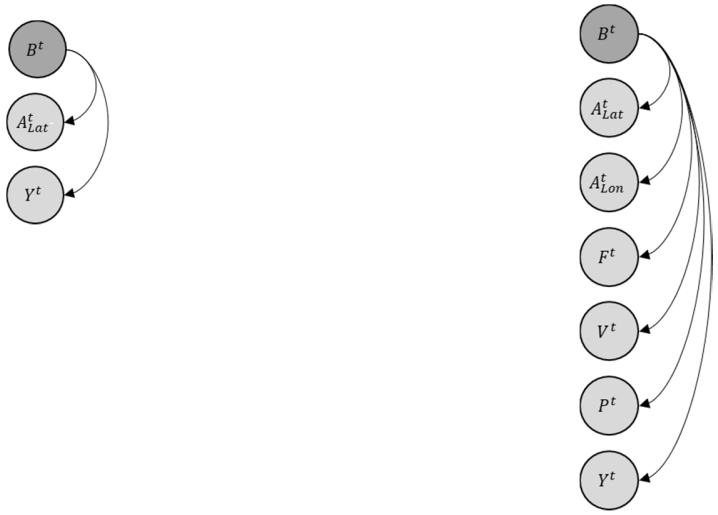


Figure 17: Learned graph structures of $p(E^t|B^t, l_l^t)$ (left) and $p(E^t|B^t, l_r^t)$ (right) in M_1.

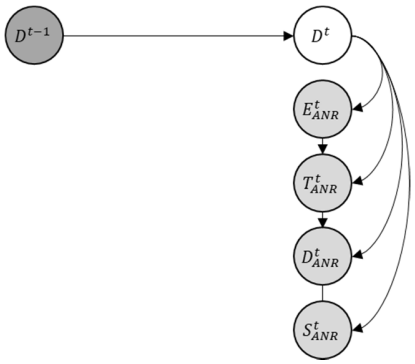


Figure 18: Learned graph structure of the embedded Bayesian classifiers realizing $p(\boldsymbol{D^t}|\boldsymbol{D^{t-1}}, \boldsymbol{l_r^t}, \boldsymbol{C^t})$ in $\boldsymbol{M_2}$

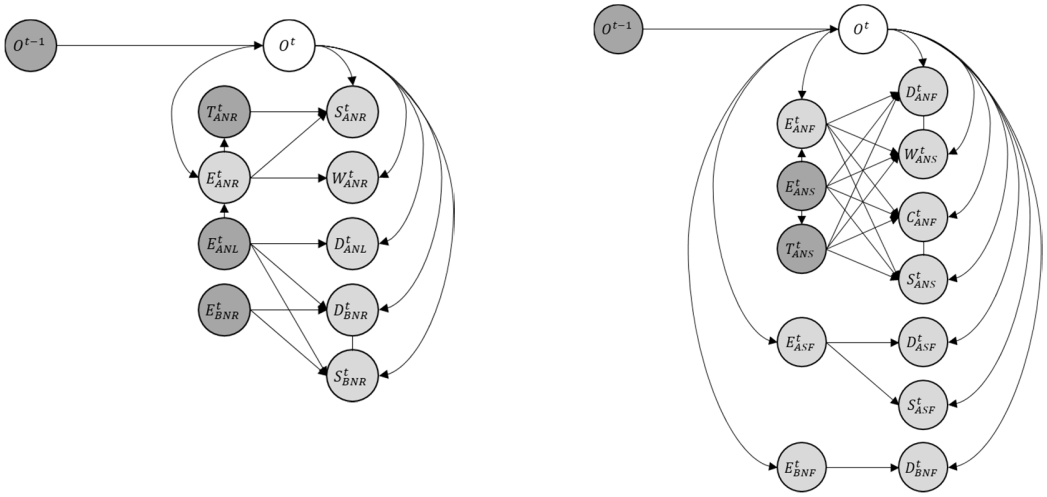


Figure 19: Learned graph structures of the embedded Bayesian classifiers realizing $p(\boldsymbol{O^t}|\boldsymbol{O^{t-1}}, \boldsymbol{l_l^t}, \boldsymbol{d_r^t}, \boldsymbol{C^t})$ (left) and $p(\boldsymbol{O^t}|\boldsymbol{O^{t-1}}, \boldsymbol{l_r^t}, \boldsymbol{d_l^t}, \boldsymbol{C^t})$ (right) in $\boldsymbol{M_2}$.

Figure 20: Learned graph structures of $p(E^t|B^t, l_l^t)$ (left) and $p(E^t|B^t, l_r^t)$ (right) in M_2.

For the integrating of the resulting models for driver intention recognition on rural roads into the TeamMate system architecture and potentially demonstrators, a framework for performing inferences in (Dynamic) Bayesian Networks following the conceptual graph structure (Figure 13) has been integrated together with the functionality for the prediction of the temporal and spatial evolution of the traffic scene, online risk assessment in respect to other traffic participants, and the online learning into a single C++ Dynamically Linked Library. For integration into the ULM simulator, this library is wrapped in a DPU, which is a format for exchangeable modules of the SILAB simulation software used by ULM. For the integration into the VED demonstrator the library is wrapped into a RTmaps package, which allows a seamless integration into the RTMaps system environment used by VED.

3.3.2.4 Environmental context information

The required input in terms of environmental context information of the overall module integrating the functionality of traffic prediction, online risk

assessment, driver intention recognition, and online learning conforms to the TeamMate system architecture and consists of

- the static environment model (including a digital road map)
- the dynamic environment model (including the state of the TeamMate vehicle and the state of all dynamic objects detected by the TeamMate vehicle)
- and an optional planned trajectory,

as defined in deliverable D5.1 "TeamMate System Architecture including open API for 2nd cycle". On an internal level, the probabilistic model for driver intention recognition on rural roads operates on the following input (for a full definition, we refer to deliverable D5.1) from which all subsequent measures are derived (As apparent from this list, the traffic prediction does not process any personal or private data):

- A digital road map that allows a reasonable reconstruction of the road structure along the prediction horizon for each considered object that shall be predicted. In the case of simulator environments, such a map can be constructed beforehand based on the simulation scenario. In the case of the VED real vehicle demonstrator, a map of the VED test course has been provided by VED.
- The state of the TeamMate vehicle, consisting (limited to the required input for the traffic prediction) of the following information:
    - Timestamp: Timestamp of the measurement.
    - PositionX: x-position of the centre of the bounding box.
    - PositionY: y-position of the centre of the bounding box.

- Heading: Heading in respect to the x-axis.
- VelocityX: Velocity in longitudinal direction.
- AccelerationX: Acceleration in longitudinal direction.
- YawRate: Radial velocity.

In the case of simulator environments, this input can be provided directly by the simulation software, in the case of the VED real vehicle demonstrator, this is provided by the VED real vehicle internal sensors, e.g., a high precision GPS.

- The state of all dynamic objects detected by the TeamMate vehicle, where each state consists (limited to the required input for the traffic prediction) of the following information:
  - Timestamp: Timestamp of the measurement.
  - PositionX: x-position of the centre of the bounding box.
  - PositionY: y-position of the centre of the bounding box.
  - Heading: Heading in respect to the x-axis.
  - VelocityX: Velocity in longitudinal direction.
  - AccelerationX: Acceleration in longitudinal direction.
  - YawRate: Radial velocity.
  - Length: Length of the bounding box in longitudinal direction.
  - Width: Length of the bounding box in lateral direction.

In the case of simulator environments, this input can be provided directly by the simulation software. In the case of potential real vehicle demonstrators,

the required input can, *in theory*, be provided by external sensors, like e.g., LIDARs, RADARs, or camera. *It is however important to note that the current level of sensor technology is not yet mature enough for the application of the models for driver intention recognition on rural roads.* The probabilistic driver model for intention recognition on rural roads was developed based on experimental data obtained in simulator experiments, with the ability to collect ground truth data in (basically) unlimited range, obviously vastly outperforming the capabilities and range of any real-world sensor available today. More specifically, the simulation environment, provided information about oncoming traffic up to a distance of 850m, unhindered by any occultation by other traffic, like e.g., a lead vehicle. Obviously, such requirements do not hold in current real-world scenarios, making the model currently inapplicable outside of simulation environments.

Nonetheless, we believe that there is a strong merit to the investigation of driver intention recognition on rural roads, even if, presently, limited to simulation environments. First, the resulting model structure and parameters help in knowledge discovery, revealing which information is potentially valuable for intention recognition, enabling us to assess e.g., the necessary range a sensor must provide to make intention recognition on rural roads feasible. As a simple demonstration, Figure 21 exemplarily shows the isolated influence of the distance to the nearest opposing traffic participant on the fast lane, D_{ANL}^t, on the probability that the driver recognizes an opportunity to change to the left lane, $\boldsymbol{p}(o_l^t | l_R^t, d_L^t, D_{ANL}^t, E_{ANL}^t = \text{true})$, while the TeamMate vehicle is located on the right lane and desires to change to left lane. The isolated influence can be obtained by assuming $\boldsymbol{P}(\boldsymbol{O^t})$ to be uniform, enabling us to and calculate

$$\boldsymbol{p}(\boldsymbol{o}_l^t|\boldsymbol{l}_R^t,\boldsymbol{d}_L^t,\boldsymbol{D}_{ANL}^t,\boldsymbol{E}_{ANL}^t=\text{true})=\frac{\boldsymbol{p}(\boldsymbol{D}_{ANL}^t|\boldsymbol{o}_l^t,\boldsymbol{l}_R^t,\boldsymbol{d}_L^t,\boldsymbol{E}_{ANL}^t=\text{true})}{\sum_{\boldsymbol{o}\in\boldsymbol{O}}\boldsymbol{p}(\boldsymbol{D}_{ANL}^t|\boldsymbol{o}^t,\boldsymbol{l}_R^t,\boldsymbol{d}_L^t,\boldsymbol{E}_{ANL}^t=\text{true})}.$$

As apparent, the probability for an opportunity is high when the opposing vehicle is near $\boldsymbol{D}_{ANL}^t < 100m$, enabling the driver to overtake after the opposing traffic has passed, but drops to a rather low probability within the region of approx. $125m < \boldsymbol{D}_{ANL}^t < 400m$, in which a collision would be potentially unavoidable if attempting a lane change. Beyond the distance of approx. $\boldsymbol{D}_{ANL}^t > 520m$, the probability for an opportunity exceeds the probability for no opportunity and steadily rises with increasing distance.

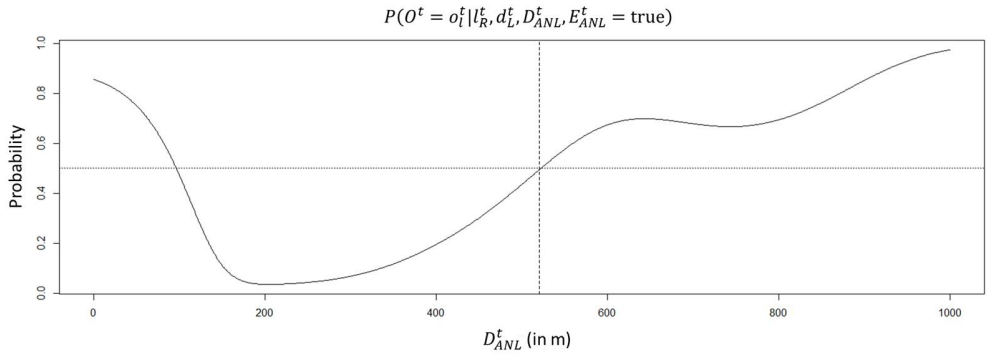


Figure 21: Exemplary visualization of $\boldsymbol{p}(\boldsymbol{o}_l^t|\boldsymbol{l}_R^t,\boldsymbol{d}_L^t,\boldsymbol{D}_{ANL}^t,\boldsymbol{E}_{ANL}^t=\text{true})=\frac{\boldsymbol{p}(\boldsymbol{D}_{ANL}^t|\boldsymbol{o}_l^t,\boldsymbol{l}_R^t,\boldsymbol{d}_L^t,\boldsymbol{E}_{ANL}^t=\text{true})}{\sum_{\boldsymbol{o}\in\boldsymbol{O}}\boldsymbol{p}(\boldsymbol{D}_{ANL}^t|\boldsymbol{o}^t,\boldsymbol{l}_R^t,\boldsymbol{d}_L^t,\boldsymbol{E}_{ANL}^t=\text{true})}$ (i.e., assuming $\boldsymbol{P}(\boldsymbol{O}^t)$ to be uniform). The dotted line represents $\boldsymbol{p}(\boldsymbol{o}_l^t|\boldsymbol{l}_R^t,\boldsymbol{d}_L^t,\boldsymbol{D}_{ANL}^t,\boldsymbol{E}_{ANL}^t=\text{true}) > \boldsymbol{p}(\boldsymbol{o}_r^t|\boldsymbol{l}_R^t,\boldsymbol{d}_L^t,\boldsymbol{D}_{ANL}^t,\boldsymbol{E}_{ANL}^t=\text{true})$, the dashed line represents $\boldsymbol{D}_{ANL}^t = \boldsymbol{520m}$.

From this simple demonstration, we can assume that a sensor range of approx. 520m would be adequate for driver intention recognition of rural road scenarios. Given the ongoing development of external sensors and other means to obtain necessary information, like e.g., V2X communication, we are confident that such information may become available in some foreseeable future. Just recently, a sensor supplier announced the first industry

demonstration[6] of an advanced radar that is able to detect vehicles and their velocities at 300m.

Furthermore, the underlying principles of the proposed model for driver intention recognition on rural roads can easily be transferrable to other scenarios, as demonstrated by the derived model for intention recognition in roundabout scenarios (c.f. section 3.3.2), that require less formidable sensor abilities. For the remainder of AutoMate, we will use experimental data obtained in the former European project HoliDes during real world driving studies on Italien highways by project partner CRF to prepare a model for driver intention recognition for the VED real vehicle demonstrator.

3.3.3 Driver Intention Recognition in roundabout scenarios

Next paragraphs describe the concept, the improvements and the implementation activities for the roundabout scenario.

3.3.3.1 Concept

The general concept of driver intention recognition is discussed in detail in deliverable D2.4 "Sensor Platform and Models including V&V results from 2nd cycle". Briefly, we refer to driver intention recognition as the problem of anticipating the maneuver, a driver is most likely to take over in the next few seconds. Such anticipation could be obtained by learning from the driver intentions in similar contextual situation.

---

[6] "https://www.businesswire.com/news/home/20181009005331/en/", last visited 19.12.2018.

The intention recognition in entering to the roundabouts mainly deals with identifying the proper moments, in which the driver, most likely, would intend to enter to roundabouts. The proper moments could also be understood as the proper traffic situations from the driver's point of view to safely enter to roundabouts. Therefore, intention recognition model detects the traffic-based opportunities in which the driver desires to enter to the roundabouts.

3.3.3.2 Implementation

In the third cycle of AutoMate, a Dynamic Bayesian Network is used to model the probabilistic driver intention recognition in entering roundabouts. The purpose of the model is to identify the proper traffic situations in which the driver intends to enter to the roundabouts. Therefore, the model learns about the proper moment to trigger a human-like entrance maneuver to the roundabouts in Automated Mode. Such model can be used to assess the safety of the entrance maneuver to roundabout in case of manual driving Mode.

In the Eva entrance scenario, we consider two potential intentions: intention to enter to roundabouts and intention to wait before the roundabouts, represented by a binary variable $I, Val(I) = \{i_E, i_W\}$.

We use information provided by the traffic situation as potential causes for the formation of intentions. We considered inputs from three alter vehicles inside the roundabouts: one vehicle in front (denoted as AN) and two vehicles behind (denoted as BN, BS) in roundabouts (Figure 22). The front and behind vehicles were defined relative to the "contact point" marked with red cross in Figure 22. Approximately at this location, the TeamMate vehicle enters to the roundabout. It also could be defined as a first possible collision point between the TeamMate vehicle with any other vehicle inside the roundabout.

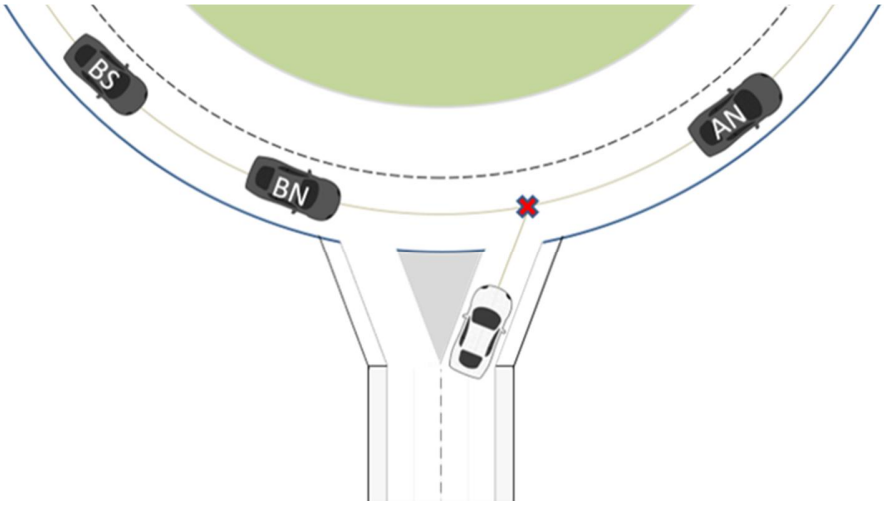


Figure 22: Schematic scheme of a roundabout. Contact point is marked with a red cross which depict the first entrance location of the TeamMate vehicle (shown in white) to the roundabout. The role of vehicles inside the roundabouts were defined relative to this point: AN (ahead next vehicle), BN (behind next vehicles) and BS (behind second vehicle) relative to the contact point.

Presence, speed and distances of these 3 vehicles (1 in front and two behinds) to the TeamMate vehicle was considered as relevant variables that potentially influence driver intention. Moreover, the time headway defined as the time required for a behind vehicle to arrive to contact point was considered for the two behind vehicles (Table 3).

Table 3: Summary of inputs from vehicles inside the roundabouts considered for modelling intention recognition in entering to the roundabouts.

Variable	Type	Description
E_X	Binary	Presence: Represents whether there exists a vehicle X in the traffic situation. X ∈ $\{BN, AN, BS\}$

D_X	Continuous	Distance: Represents the distance between the TeamMate vehicle and a vehicle X along the course of the road. X ∈ $\{BN, AN, BS\}$
S_X	Continuous	Speed: Represents the speed of a vehicle X. X ∈ $\{BN, AN, BS\}$
H_X	Continuous	Time headway: Represents the time required for a vehicle X to arrive to contact point defined by Distance of a vehicle X to contact point divided to the speed of a vehicle X. X ∈ $\{BN, BS\}$

Additionally, the speed of the TeamMate vehicle was considered as a relevant model variable, affecting the decision of the driver in entering to the roundabout.

Although the speed of the vehicle is controlled by the driver's decision to enter the roundabout or to wait before roundabout, the initial speed of the vehicle approaching to the roundabout would influence the driver decision. For example, in two identical "not heavy" traffic situation, the decision of driver might be differing if the driver is in full-stop (because of a very heavy traffic at the previous moment) or the vehicle is not in full stop (because it just arrives to the roundabout and still has an initial speed). In the case the driver is in full stop, he considers the required time for speeding up in his calculation of safe entrance maneuver, whereas, the vehicle approaching a roundabout could enter faster to the roundabout if he detects a safe distance to other vehicles.

Therefore, we build two separate models for each of the two roundabout types in driving scenario. One model considering the variables described in Table 3, focusing on inputs from other vehicles (traffic situation) and refraining inputs

from TeamMate vehicle. Another model considering the speed of TeamMate vehicle as model variable in addition to inputs from other vehicles.

The driving scenario for intention recognition contained two types of roundabouts: small roundabouts with a diameter of approximately 10 m and big roundabouts with a diameter of approximately 40 m. Because of the difference in dimensions, the range of the speed and gap size between vehicles inside the roundabouts were different (c.f. section 4.3.2.1.1). To interpret the resulting distributions and model outcomes, we analysed data from each type of the roundabouts separately. Such that we learned four models, two for small roundabouts and two for big roundabouts as explained above.

For the formulation of the probabilistic intention recognition model, let $\boldsymbol{O}_I$ denote the whole set of discrete and continuous variables representing the causes for the formation of intentions.

The component for intention recognition, then is realized akin to a Maximum-entropy Markov Model, where for any number of times steps T the (conditional) joint distribution $p(I^{1:T}|\boldsymbol{O}_I^{1:T})$ is defined as:

$$p(I^{1:T}|\boldsymbol{O}_I^{1:T}) = p(I^{1:T}|\boldsymbol{O}_I^{1:T}) \prod_{t=2}^{T} p(I^t|I^{t-1}, \boldsymbol{O}_I^t)$$

Meaning that intentions evolve based on the situational context encountered (explained in detail in deliverable D2.4 "Sensor Platform and Models including V&V results from 2nd cycle").

During runtime, the model can be used to continuously maintain a belief state over intention:

$$p(I^t|\boldsymbol{o}_I^{1:t}) \propto \sum_{i \in I} p(I^t|i^{t-1}, \boldsymbol{o}_I^{1:t}) p(i^{t-1}|\boldsymbol{o}_I^{1:t-1}).$$

The distribution for continuous variables were realized using Gaussian Mixture Models. We used structure learning method to identify the most important variables and their causal relations.

Two resulting graph structure for big roundabouts is shown in Figure 23. The variables depicted with circles were presented to the model (see Table 3 for definition of the variables). The structure learning method selected the most important variables out of the presented variables (represented with black borders). The non-selected variables were shown in pale. The causal relations between variables shown with black arrows, where the pale gray arrows show possible causal relations, tested by the model but did not get selected. The right plot demonstrates the structure graph obtained considering only causes of the intention formation as the model variables. The left model included the speed of the TeamMate (V^t) as an extra variable during structure learning and resulted in a different selected set of variables.

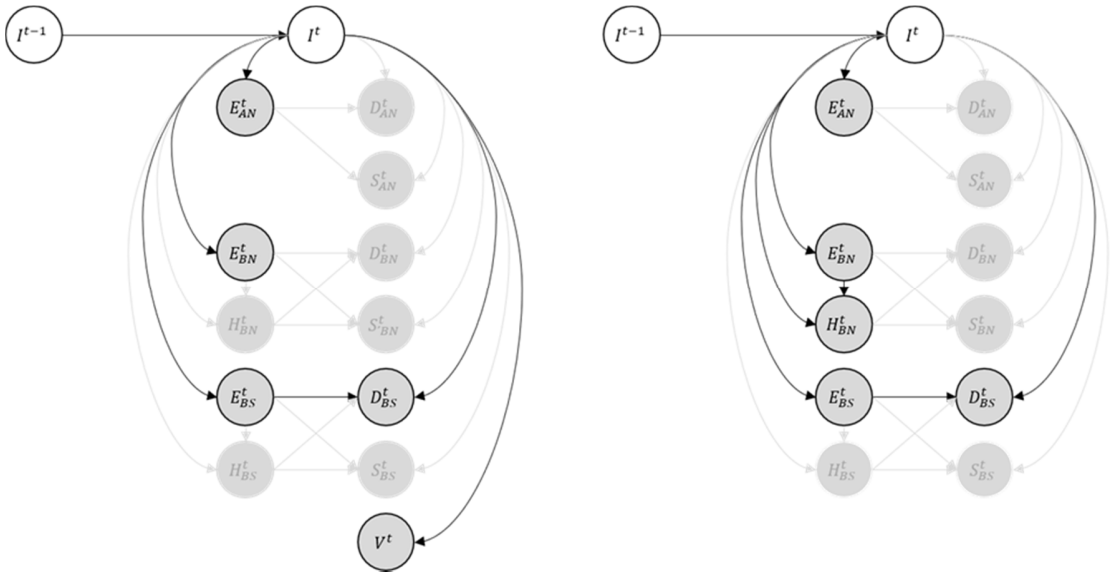


Figure 23: Graph structure learned for big roundabouts using structure learning method.

Similarly, Figure 24 depicts the resulting graph obtained for small roundabouts with (left) and without considering the speed of the TeamMate vehicle (right).

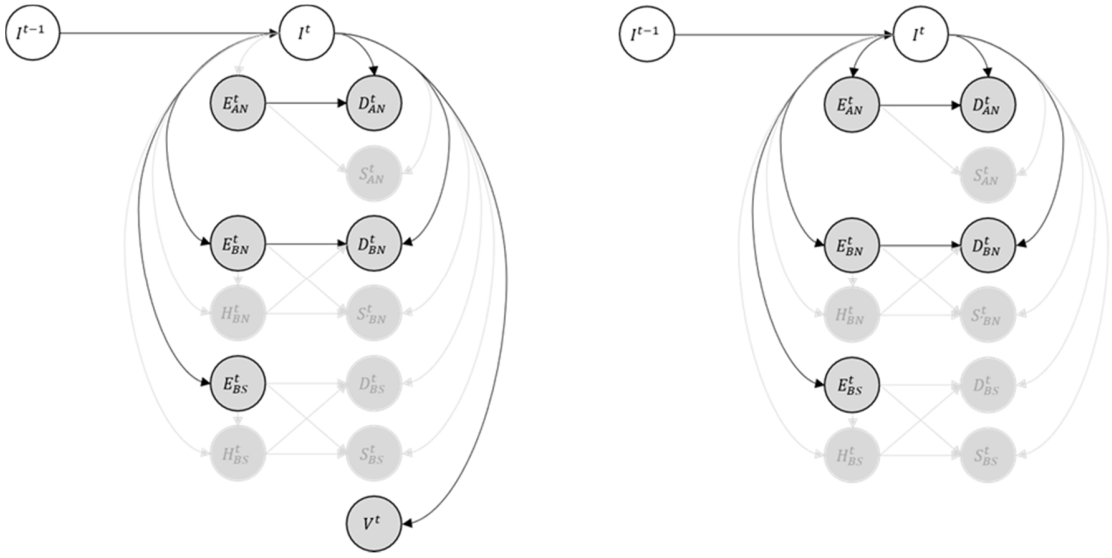


Figure 24: Graph structure for small roundabout.

3.4 E3.1 – Situation and vehicle model

The enabler E3.1 is now described in the next paragraphs, including its implementation.

3.4.1 Semantic enrichment of the situation model

Situation information from the perception layer like scene objects is enriched with semantic information, where enriched information describes the possible relationships and interactions between scene objects. Furthermore, the interaction and relationships are extended with the first ordered logic that allows to infer possible legal manoeuvres for vehicles in particular situations.
The ontology for building the relationships between the scene objects, the semantic web rule language for modelling the first ordered logic of traffic rules, and the reasoner to infer the task were presented during the 1st cycle of the project, in D2.2 results are recorded on the simulated data.
During the 2nd project cycle *JNIOWLBridge* was developed to integrate the concept presented in the 1st cycle of the project into the situation interpretation module. To recap *JNIOWLBridge* is wrapper that provides the bridge between the Ontology Web Language (OWL) and the reasoner which are available in Java to our C++ module. A brief illustration of the *JNIOWLBridge* is provided in D2.4.
The *JNIOWLBridge* is linked to our C++ semantic enrichment module as a library. The ontology and logical rules used for the semantic enrichment was described in D2.2. The ontology contains the taxonomy and semantic relations of relevant scenes objects as pedestrian, road, vehicle, traffic light and signal. The logical rules described basic traffic rules in urban scenes. The semantic

enrichment module takes as input the detected scene objects and the modelled ontology and executes the following steps:

1. loading of ontology by name
2. generation of relations between detected scene objects (which are treated as individuals), such as "vehicle x is on lane y"
3. adding of these individuals and theirs relations to the ontology
4. inference of new relations between the individual objects
5. inference of legally allowed manoeuvres per vehicle
6. deletion of previously added individuals and relations from the ontology
7. repetition of algorithm starting at step 2

In the 3rd project cycle the ontology was extended in such a way that most AutoMate use cases can be addressed, with the option of achieving full coverage of the use cases which are targeted by the TeamMate car. To perform comprehensive validation and verification that could fulfil the AutoMate use case, sophisticated synthetic test data D_{test} was generated. An interface to fetch the D_{test} for semantic enrichment component is being implemented within the context of verification and validation cycle. With the planned use of this submodule as a prior for predicting the future evolution of the situation, its accuracy in real traffic could be used as weight for the prior.

However, du to other ongoing integration activities, it has not been possible yet to determine the ultimate approach to the integration of the submodule in the final demonstration set up. The development of the submodule within the AutoMate project has shown that very sophisticated, complex and high accurate perception layer information is required to achieve the submodule's required output quality. Together with project partners, we are undertaking a

continued effort to prepare the type of data from real world driving which possesses all required information for the submodule to run. This process has proven to be very time consuming yet imperative.

Further, the integration of the reasoner itself into any more complex system the enabler may yield a runtime for the whole module which renders it insufficient to function properly within the system constraints. While it was possible to greatly reduce the runtime of the submodule, the runtime within the final system setup may be different matter

3.4.2 Predicting the future evolution of the traffic scene

The purpose of vehicle models is to predict the temporal and spatial evolution of the traffic scene (in the following simply referred to as *traffic prediction*), based on the information provided by the sensor and communication platform and the situation model, as a necessary input for online risk assessment (for more information on online risk assessment, we refer to D3.3 "Concepts and algorithms incl. V&V results from 1st cycle", D3.5 "Concepts and algorithms incl. V&V results from 2nd cycle"), and D3.7 "Concepts and algorithms incl. V&V results from 3rd cycle").

Concept, development, and implementation of the algorithm pipeline for the traffic prediction have been entirely developed within the context of AutoMate. No part of the component has been inherited from previous projects nor simultaneously addressed in any other European projects.

3.4.3 Concept

The underlying concept for traffic prediction has already been described in detail in deliverables D2.2 "Sensor Platform and Models including V&V results from 1st cycle" and D2.4 "Sensor Platform and Models incl. V&V results from

2nd cycle". In the following, we will recapitulate the most important aspects necessary for the understanding of the traffic prediction and provide any updates introduced for the third cycle.

Let $V = \{v_1, \ldots, v_{n_V}\}$ denote a number of n_V traffic participants detected in the vicinity of the TeamMate vehicle (if a prediction for the TeamMate vehicle is desired, the TeamMate vehicle can be considered as belonging to this set). At each time step t, the traffic prediction uses the environmental context information provided by the TeamMate vehicle's sensor platform, to create a multivariate Gaussian belief state $p(\boldsymbol{S}_v^t)$ for each $v \in V$, where

$$\boldsymbol{S_v} = \{X_v, Y_v, \Theta_v, V_v, A_v, W_v\},$$

with

1. X_v representing the global x-coordinate (in meters) of the centre of the bounding box of v in a two-dimensional spatial coordinate system,
2. Y_v representing the global y-coordinate (in meters) of the centre of the bounding box of v in a two-dimensional spatial coordinate system,
3. Θ_v representing the yaw angle (in radians) of v relative to a constant reference axis,
4. V_v representing the longitudinal velocity (in meters per second) of v along its heading,
5. A_v representing the longitudinal acceleration (in meters per second squared) of v along its heading,
6. W_v representing the yaw-rate (in radians per second) of v.

As previously described in deliverable D2.2 "Sensor Platform and Models including V&V results from 1st cycle" and deliverable D2.4 "Sensor Platform and Models including V&V results from 2nd cycle", the prediction of the spatial and temporal evolution of the traffic scene is based on so-called *Constant Turn*

Rate and Acceleration (CTRA), resp. *Constant Yaw-Rate and Acceleration* (CYRA) motion models operating on a corresponding six-dimensional state space $\boldsymbol{s}_v^t = (x_v^t, y_v^t, \theta_v^t, v_v^t, a_v^t, w_v^t)^T$.

Let Δt (in seconds) denote some prediction time, the state transition equation for this model is given by

$$\boldsymbol{s}_v^{t+\Delta t} = f_{CYRA}(\boldsymbol{s}_v^t, \Delta t) = \begin{pmatrix} x_v^{t+\Delta t} \\ y_v^{t+\Delta t} \\ \theta_v^{t+\Delta t} \\ v_v^{t+\Delta t} \\ a_v^t \\ w_v^t \end{pmatrix},$$

where

$$x^{t+\Delta t} = \begin{cases} x^t + \frac{1}{w^t}\left[\frac{a^t}{w^t}(\cos\theta^{t+\Delta t} - \cos\theta^t) + v^{t+\Delta t}\sin\theta^{t+\Delta t} - v^t\sin\theta^t\right], & w^t \neq 0 \\ x^t + \left(\frac{1}{2}a^t(\Delta t)^2 + \Delta t\, v^t\right)\cos\theta^t, & w^t = 0 \end{cases},$$

$$y^{t+\Delta t} = \begin{cases} y^t + \frac{1}{w^t}\left[\frac{a^t}{w^t}(\sin\theta^{t+\Delta t} - \sin\theta^t) - v^{t+\Delta t}\cos\theta^{t+\Delta t} + v^t\cos\theta^t\right], & w^t \neq 0 \\ y^t + \left(\frac{1}{2}a^t(\Delta t)^2 + \Delta t\, v^t\right)\sin\theta^t, & w^t = 0 \end{cases},$$

$$\theta^{t+\Delta t} = \theta^t + \Delta t\, w^t,$$

and

$$v^{t+\Delta t} = v^t + \Delta t\, a^t.$$

For the traffic prediction in the first cycle, we assumed the yaw-rate and acceleration to be kept constant, such that $a^{t+\Delta t} = a^t$ and $w^{t+\Delta t} = w^t$ for any temporal step width Δt and number of steps η_{max}, i.e., we assumed that a traffic participant keeps the current yaw-rate and acceleration over the complete prediction horizon $\eta_{max}\Delta t$. In the second cycle, we addressed this

limitation by using Dynamic Bayesian Networks (DBN) to infer the most probable behaviour amongst two hypothetical behaviours, lane-keeping (LK) and lane-changing (LC), for each traffic participant and incorporating simple but computationally inexpensive driver models to better predict the future behaviour of traffic participants for these behaviours.

3.4.4 Improvements

In the third cycle, we focussed on the improvement of the underlying driver models for predicting the lateral and longitudinal control of traffic participants. For this, we redefine the computation of the of the future state $\boldsymbol{s}_v^{t+\Delta t}$ as

$$\boldsymbol{s}_v^{t+\Delta t} = f_{CYRA}\left(\boldsymbol{s}_v^t, \Delta t, M, \boldsymbol{s}_{v_{Lead}}^t, h\right) = \begin{pmatrix} x_v^{t+\Delta t} \\ y_v^{t+\Delta t} \\ \theta_v^{t+\Delta t} \\ v_v^{t+\Delta t} \\ a_v^{t+\Delta t} \\ w_v^{t+\Delta t} \end{pmatrix},$$

with M denoting the digital road map, $\boldsymbol{s}_{v_{lead}}^t$ denoting the (potentially predicted) state of a potential vehicle v_{Lead} acting as the lead vehicle for v, and $h \in \{LK, LC\}$ denoting the assumed behaviour. Here, $x_v^{t+\Delta t}$, $y_v^{t+\Delta t}$, $\theta_v^{t+\Delta t}$ and $v_v^{t+\Delta t}$ are provided by the original CYRA motion-model. In contrast, $a_v^{t+\Delta t}$ is given by

$$a_v^{t+\Delta t} = g_A\left(\boldsymbol{s}_v^t, \Delta t, M, \boldsymbol{s}_{v_{Lead}}^t, h\right),$$

while $w_v^{t+\Delta t}$ is given by

$$w_v^{t+\Delta t} = g_W(\boldsymbol{s}_v^t, \Delta t, M, h).$$

For lateral control, $g_W(\boldsymbol{s}_v^t, \Delta t, M, h)$, we assume that v adapts its yaw-rate in order to keep itself aligned with the course of the road while simultaneously minimizing the lateral deviation to the intended target lane.

For longitudinal control, $g_A\left(\boldsymbol{s}_v^t, \Delta t, M, \boldsymbol{s}_{v_{Lead}}^t, h\right)$, we assume that v adapts its acceleration in order to slowly reach the current speed limit (as given by the digital road map). If travelling on or returning to the right lane and approaching a lead vehicle, we assume the v adapts its acceleration to slowly zero the inverse time to collision to the lead vehicle.

Given a multivariate Gaussian belief state $p(\boldsymbol{S}_v^t) = N(\boldsymbol{s}_v^t, \Sigma_v^t)$ and using the CYRA motion-model, we obtain a prediction for a future time step $p(\boldsymbol{S}_v^{t+\Delta t})$ by approximating

$$p\left(\boldsymbol{S}_v^{t+\Delta t}\right) = N(\boldsymbol{s}_v^{t+\Delta t}, \boldsymbol{\Sigma}_v^{t+\Delta t}) = \int f_{CYRA}\left(\boldsymbol{s}_v^t, \Delta t, M, \boldsymbol{s}_{v_{Lead}}^t, h\right) p(\boldsymbol{s}_v^t)\, d\boldsymbol{s}_v^t$$

using the technique of unscented transformation (as previously described in deliverable D2.2 "Sensor Platform and Models including V&V results from 1st cycle").

To counterbalance the assumptions concerning the selected yaw-rate and acceleration, which will reduce the uncertainties over time, we inflated the resulting covariance matrix $\boldsymbol{\Sigma}_v^{t+\Delta t}$ in the following way:

$$'\boldsymbol{\Sigma}_v^{t+\Delta t} = \boldsymbol{\Sigma}_v^{t+\Delta t} + \Delta t \begin{pmatrix} \max(0,d) \\ \max(0,d) \\ 0.01 \\ 1.0 \\ 0.1 \\ 0.00245 \end{pmatrix}^2 \boldsymbol{I}_6,$$

where d is chosen such that the three-sigma interval of the lateral deviation of a vehicle driving in the middle of a lane would completely cover the lane.

3.4.5 Implementation

The internal procedure for the traffic prediction has been described in deliverable D2.4 "Sensor Platform and Models incl. V&V results from 2nd cycle".

Concerning the actual implementation and integration into the TeamMate system architecture and demonstrators, the traffic prediction has been integrated together with the functionality for online risk assessment in respect to other traffic participants, the driver intention recognition, and the online learning into a single C++ Dynamically Linked Library. For integration into the ULM simulator, this library is wrapped in a so-called DPU, which is a format for exchangeable modules of the SILAB simulation software used by ULM. For the integration into the VED demonstrator the library is wrapped into a RTmaps package, which allows a seamless integration into the RTMaps system environment used by VED.

3.4.6 Environmental context information

The required input in terms of environmental context information of the overall module integrating the functionality of traffic prediction, online risk assessment, driver intention recognition, and online learning conforms to the TeamMate system architecture and consists of

- the static environment model (including a digital road map)
- the dynamic environment model (including the state of the TeamMate vehicle and the state of all dynamic objects detected by the TeamMate vehicle)
- and an optional planned trajectory,

as defined in deliverable D5.1 "TeamMate System Architecture including open API for 2nd cycle". On an internal level, the traffic prediction operates on the following input (for a full definition, we refer to deliverable D5.1):

- The state of the TeamMate vehicle, consisting (limited to the required input for the traffic prediction) of the following information:
    - Timestamp: Timestamp of the measurement.
    - PositionX: x-position of the centre of the bounding box.
    - PositionY: y-position of the centre of the bounding box.
    - Heading: Heading in respect to the x-axis.
    - VelocityX: Velocity in longitudinal direction.
    - AccelerationX: Acceleration in longitudinal direction.
    - YawRate: Radial velocity.
    - PoseMotionCovMate: Covariance matrix for pose and motion.
    - In the case of simulator environments, this input can be provided directly by the simulation software, in the case of the VED real vehicle demonstrator, this is provided by the VED real vehicle internal sensors, e.g., a high precision GPS.
- The state of all dynamic objects detected by the TeamMate vehicle, where each state consists (limited to the required input for the traffic prediction) of the following information:
    - Timestamp: Timestamp of the measurement.
    - ID: a unique and consistent ID, allowing the identification of the vehicle over time.
    - PositionX: x-position of the centre of the bounding box.
    - PositionY: y-position of the centre of the bounding box.

- Heading: Heading in respect to the x-axis.
- VelocityX: Velocity in longitudinal direction.
- AccelerationX: Acceleration in longitudinal direction.
- YawRate: Radial velocity.
- PoseMotionCovMate: Covariance matrix for pose and motion.
- Length: Length of the bounding box in longitudinal direction.
- Width: Length of the bounding box in lateral direction.
- Existence Probability: Confidence that the detected object is existing.

- In the case of simulator environments, this input can be provided directly by the simulation software, in the case of the VED real vehicle demonstrator, this is provided by the VED real vehicle external sensors, e.g., LIDARs.

- A digital road map that allows a reasonable reconstruction of the road structure along the prediction horizon for each considered object that shall be predicted. In the case of simulator environments, such a map can be constructed beforehand based on the simulation scenario. In the case of the VED real vehicle demonstrator, a map of the VED test course has been provided by VED.

As apparent from this list, the traffic prediction does not process any personal or private data.

3.5 Driving Task Model

The Driving Task model is described in the following sections.

3.5.1 Scenario and use cases where the driving task model is relevant

Although the driving task (*DriveGOMS*) model approach is not a technical enabler, it supports the development of the HMI concept of the TeamMate car concept. Specifically, together with ULM we are using it to develop a better understanding of the task structure behind the Peter use case. The focus is on how the drivers/users of the TeamMate concept see their task, interact with the system, monitor the environment, and make decisions. As in deliverable D4.4 *Metrics and plan for V&V of the TeamMate HMI software in the 2nd cycle* described, the knowledge from the modeling is then applied to enabler E6.1.

3.5.2 Improvements

During the 2nd cycle a preliminary model had been developed within an exploratory study in the ULM simulator. From this, we had derived an initial model for the task structure in the *Peter scenario*. The main work during the 3rd cycle went towards the validation of the model. To this end, we conducted and analysed a study at DLR using our methodology, attempting to replicate the task model we had identified from the data the ULM sample had provided. This experiment is described in section 4.4.

3.5.3 Comparison with similar State-of-the-Art approaches

Task analysis as used within Human Factors are not software systems, but methodologies to collect, prepare and analyse data about situations of interest. Their goal is not necessarily to predict individual human behaviour on a time

scale of seconds, but rather produce an insight into the basic goal structure of the task of interest. This is crucial information in the context of the design of Human-machine interfaces, as the design of the interface is also a design of the task itself. Further, a detailed understanding goals, cognitive, perceptual and motor aspects of a task greatly aid the formulation about specific aspects of the Human-machine interaction, which can then be further investigated in targeted experiments. This is the approach we are following regarding Enabler E6.1.

Of course, there are other approaches to driver modelling, as well as other task analysis methods. In our opinion, however, they are unsuitable for our specific purpose. The literature both on driver modelling and task analysis is vast, and a complete review would be impossible within the available space here.

There exist a few notable approaches to driving task analysis. The work of McKnight and Adams[7] from 1970 details thousands of single tasks or activities which need to be carried out under certain conditions when driving a car. The intended use of this work was improving driving school education.

However, it is neither complete (i.e., covers every conceivable situation), nor specific enough to be applied unambigously given a specific situation. For example, it states that when changing lanes, it should be checked for vehicles approaching from the rear on the new lane. It does not say though what might be an acceptable distance, differential speed or time-to-collision where a lane change still might be possible to change lanes.

[7] McKnight, A. J., & Adams, B. B. (1970). *Driver Education Task Analysis. Volume II: Task Analysis Methods. Final Report.* Washington, DC.

Being from 1970, of course it does say little about interactions with in car devices, or even how a driver could or should work together with an automated vehicle (such as the TeamMate car).

Another example is part of the *Generic Intelligent Driver Support* (*GIDS*[8]), the result of a European project from the early 1990s. It connects task analytic approaches with a specifically designed Small World to study and support human driving. Unfortunately, this task analysis is only documented in excerpts. Further, in our estimation, the task analysis and situation modelling work mainly because of the small world approach, which sets important constraints. This markedly differs from the TeamMate approach, where real world situations are addressed.

A wider known computational model stemming from the task analysis based tradition is Distract-R[9]. This is essentially a tool which computes task execution times of secondary tasks during manual driving. It is based on a driver model implemented in the cognitive architecture ACT-R[10]. Unfortunately, it models only a very small class of problems, namely driving on a straight vs curved road, while entering numbers into a cell phone via different input methods.

Thus, the available task analytic or even computational methods derived from them do not sufficiently cover the aspects relevant for AutoMate. Further, our DriveGOMS-approach puts special emphasis on qualitative empirical methods, which are used together with quantitative approaches (such as eye tracking)

---

8 Michon, John A. (Hg.) (1993): Generic Intelligent Driver Support. A comprehensive report on GIDS. London: Taylor & Francis.

9 http://cog.cs.drexel.edu/distract-r/

10 https://www.cs.drexel.edu/~salvucci/publications/Salvucci-HF06.pdf

to develop a full understanding of drivers' goals and strategies to task execution. This is an indispensible source of information which needs to be considered for the development of enabler E6.1.

Regarding machine learning based approaches to driver modelling such as E2.1 (Driver Intention Recognition), of course they are essential when trying to predict driver behavior in specific situations, especially when trying to achieve online prediction. In our estimation, here they are usually superior to rule based driver modelling approaches, since the pattern behind driver behavior often depends on complicated relationships between several continuous variable.

However, the model's behavior has to be learned explicitely. Small changes in the scenario, such as using different input methods to trigger the lane change maneuver in the *Peter scenario* require conducting empirical studies recording this exact behavior from human subjects. When trying to explore a design space, it is often preferable to iterate quickly through different possible solutions, and arrive at "guestimates" for expected driver behavior which can exclude bad design choices and help state hypotheses that can be tested with a larger sample.

It is for these types of scenarios where we believe DriveGOMS is a valuable addition to the toolkit the Human Factors researcher has available. While it is often desirable to predict behavior, sometimes it is more important to understand *why* a certain behavior has been shown, as well as describing it with a symbolic notation. These are two further very important features of DriveGOMS. Our validation of this approach for the *Peter scenario* is reported in section 4.4.

4 Validation of enablers

This section presents the verification and validation on component level of the enablers, i.e. how to validate that the enablers support the cooperation of the driver and the TeamMate car, in their final stage.

4.1 E1.1 – Driver monitoring system with driver state model for distraction and drowsiness

In the previous cycles we have performed drowsiness tests on a set of 60 hours of recording on 30 drivers in simulator conditions and presented related performance results.
The experiments of the third cycle have focuses on Visual distraction which is the output required for the Martha scenario.

4.1.1 Drowsiness experiment

The experiments of the previous cycles were conducted in the static vehicle. The camera was placed in its nominal position: behind the steering wheel looking upward of about 17° through the steering wheel. The aim of these tests were to validate the performances of the instrument identification output. The tests protocol consisted on asking subjects to look at different areas of the vehicle in agreement with the Automate requirements.
The experiments were performed with N=20 drivers. 10 men and 10 women; 5 wearing glasses. The raw video provided by the driver monitoring system were recorded and analysed offline.
The analysis has shown that for drivers not wearing glasses all areas are detected with a detection rate above 75% which is already fully acceptable

regarding the automate requirements. Still instruments close to the driver (central mirror, navigation display…) are not as well detected than the one further (left and right mirrors). This can be explained by the fact that the head gaze accuracy is better than the eye accuracy and the driver moves the head to look at far instrument while it will mainly move the eyes when the instruments are close. The results have also shown a performance variability depending on the subjects.

In this third cycle the driver monitoring system was integrated in the demonstrator vehicles and validation tests were initiated. In this report will focus our presentation on the tests performed with the Vedecom Picasso C4 demonstration car. Because of the non optimal position of the camera it is expected some performance degradation.

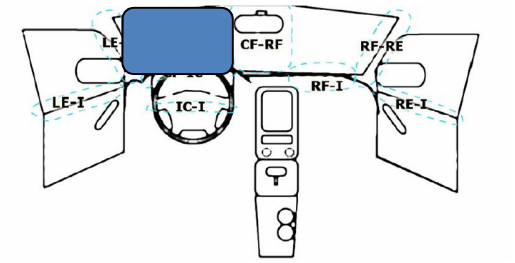
4.1.2 Test protocol

The purpose of this test is to determine a generic calibration of the driver's eye/head gaze and estimate the DMS performances regarding the area detection The test is performed in door in the Vedecom C4 picasso demonstrator car on 18 employes of Vedecom. Subjects are asked to look at various areas out and inside the cockpit according to the defined following sequence:

4.1.2.1 Step 1: Initialisation; duration: 40s

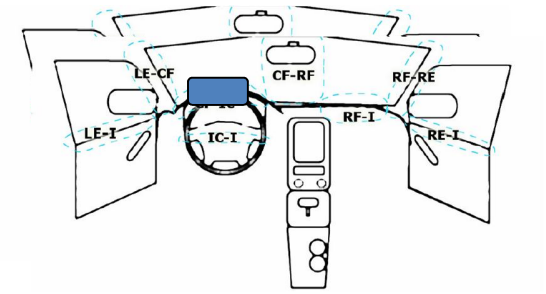
- Open the door, enter the car
- Adjust seat, position, put Seatbelt ON
- Start recording

- Adjust right mirror, left mirror and rear mirror
- Look at the road in front.
- The instructor will precise when to start moving the head.



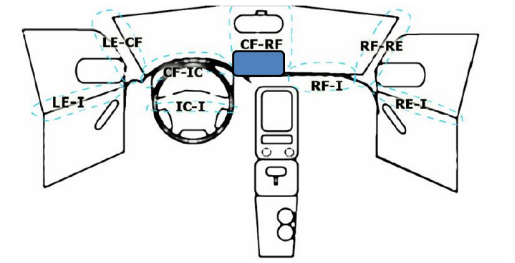
4.1.2.2 Step 2: focus on Automate Display; Duration: 5s

- Look at the Automate Display
- Back to road (neutral position)



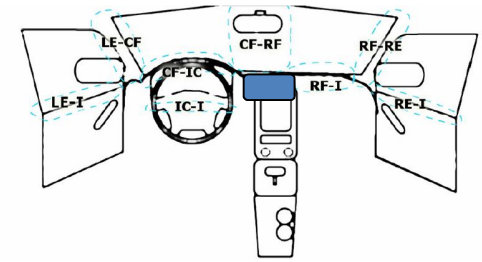
4.1.2.3 Step 3: focus on Instrument Cluster; Duration: 5s

- Look at the Instrument Cluster
- Back to road (neutral position)



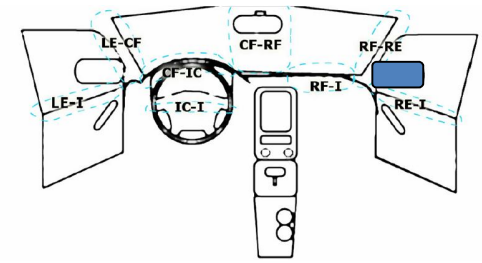
4.1.2.4 Step 4: focus on Central Display; Duration: 5s

- Look at the Central Display
- Back to road (neutral position)



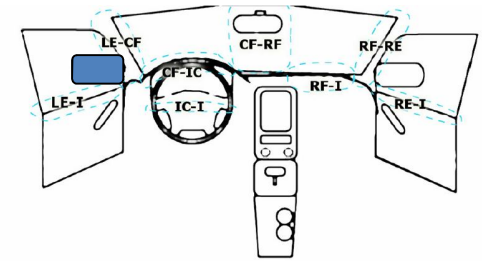
4.1.2.5 Step 5: focus on right rear-view mirror; Duration: 5s

- Look at the right-side mirror
- Back to road (neutral position)



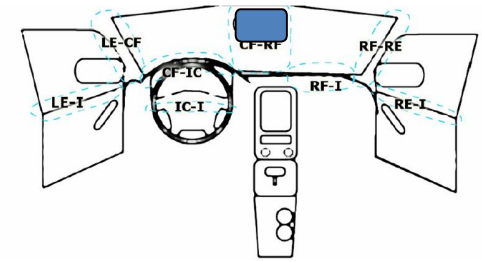
4.1.2.6 Step 6: focus on left rear-view mirror; Duration: 5s

- Look at the left-side mirror
- Back to road (neutral position)



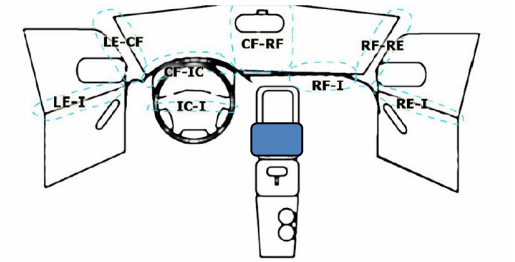
4.1.2.7 Step 7: focus on central rear-view mirror; Duration: 5s

- Look at the rear-view mirror
- Back to road (neutral position)



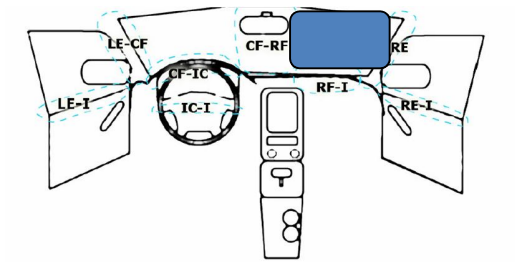
4.1.2.8 Step 8: focus on Central Display; Duration: 5s

- Change the radio frequency
- Back to road (neutral position)



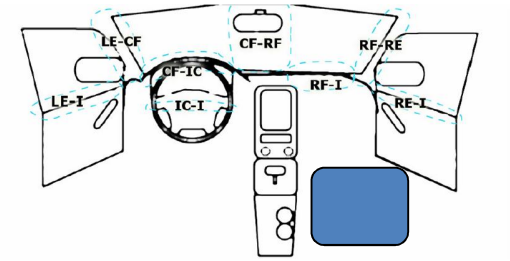
4.1.2.9 Step 9: focus on Right Windshield; Duration: 5s

- Look at the right Windshield (The subject must look at a post it placed outside so his gaze intersect the WS right in the middle horizontally and quarter of the bottom)
- Back to road (neutral position)



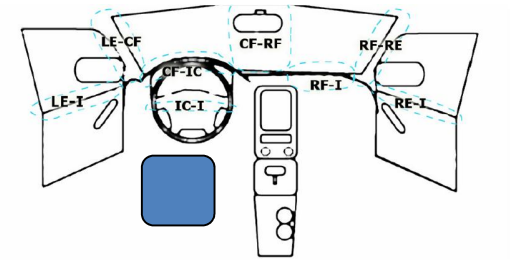
4.1.2.10 Step 10: focus on Passenger Seat; Duration: 5s

- Look at the smart phone placed on the passenger seat
- Back to road (neutral position)



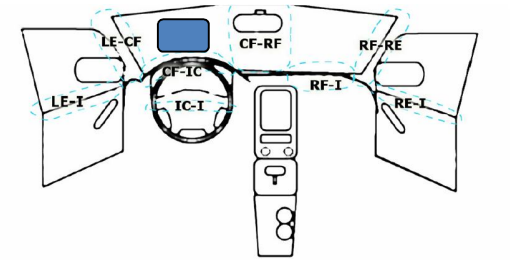
4.1.2.11 Step 11: focus on Knees; Duration: 5s

- Take the smart phone and place it on your left thigh
- Read the message written on it
- Back to road (neutral position)



4.1.2.12 Step 12: focus on blinks; Duration: 10s

- Make some blinks
- Do whatever you want for a few seconds.
- Back to road (neutral position)



4.1.3 Results

This section describes the results achieved so far for the DMS, considering both the quantitative and qualitative analysis.

4.1.3.1 Quantitative analysis

The table below show the average statistics on 50% of the "best performing drivers". These results confirm the results presented in the previous cycle:
The instrument display and the automate display are not as well detected as further instruments. It points out the difficulty to make the difference between the displays automate and instrument cluster and the windshield right. This is explained by the fact that these displays are closer to the bottom of the windshield that the ones of the previous test cycles.
We can note that the seat and knee area are badly detected due to the high head pitch. The detection of these areas is not required within automate still CAF will investigated further these issues as we think it would of interest to provide such information allowing to introduce off road levels.

Instrument	True Positive	False Positive	True Negative	False Negative	Sensitivity	Specificity	Number of labeled events
Road	928	53	3056	139	0.87	0.98	1067
Windshield Right	81	11	4084	0	1.00	1.00	81
Left Mirror	82	0	4094	0	1.00	1.00	82
Right Mirror	62	75	4033	6	0.91	0.98	68
Rear Mirror	89	0	4087	0	1.00	1.00	89
Automate Display	56	143	3935	42	0.57	0.96	98
Instrument Cluster	56	4	4062	54	0.51	1.00	110
Central Display	222	40	3898	16	0.93	0.99	238
Knees	166	0	3954	56	0.75	1.00	222

Table 4: DMS performances with respect to the different areas inside the vehicle.

4.1.3.2 Qualitative analysis

A qualitative analysis based on a visual analysis of the DMS video output shows heterogeneous results: for some subjects, the "eye gaze" point perfectly on the corresponding instrument while for others it is not at all the case.

Indeed some people have the look that points excessively down while others have the look that points abnormally up (this is especially the case of people with glasses).

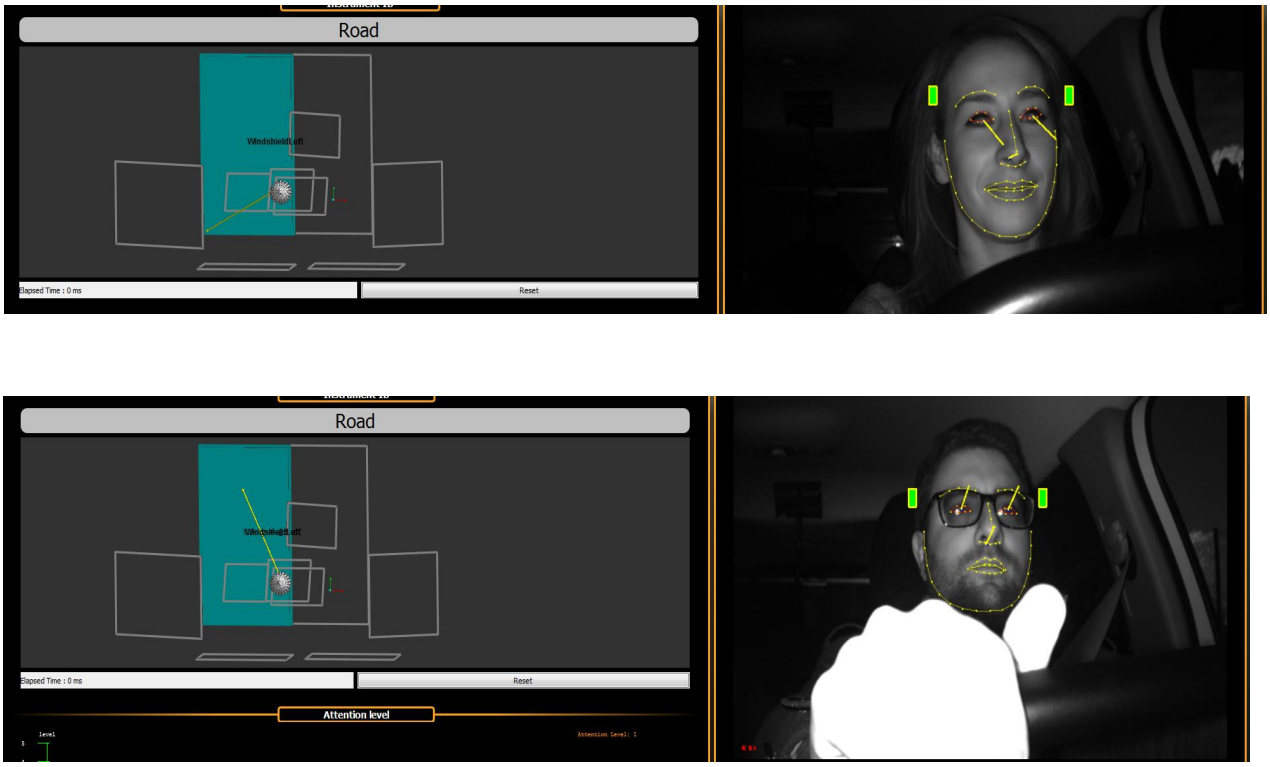


Figure 25: DMS video outputs (considering glasses as well).

It was therefore interesting to first analyze whether this abnormal behavior in these people was correctable via a fixed offset to "calibrate" the trajectory of their "eye gaze". Taking as an example the people in whom the results are very good, the offsets were calculated, and the conclusion was to see that the results were on the one hand much better but also showed that this offset was

not fixed: indeed, if for some instruments it was much better, for others it was less than before).

To solve this problem, one solution would be to perform a manual calibration for each person preliminary to the drive test.
To do this, we could invite the person the first time they get in the car observed each instrument for 5 secs, and automatically calibrate their eyes on the center of the instrument (thus an offset would be assigned for each instrument and for each person). This solution will be proposed to the partners.

4.2 E1.2 – V2x communication

During the 3rd cycle, conformance testing and field tests were performed. The conformance testing was took place in the test track of Satory (an area south of Versailles in France). An RSU was set up, and its broadcasted messages was received by an OBU from YoGoKo. The OBU was integrated in the test car. The purpose of such test is to confirm that the Cohda MK5 unit and YoGoKo's OBU unit are able to communicate and understand each other thanks to the standardized protocols. The tests were successful, thus, field test was carried out there to measure the communication distance of the RSU. Taking into account the environment of the test track it can be classified as rural environment. Based on numerous measurement the communication distance was around 600m, which fulfils the 400m threshold value that was set in Deliverable 2.5[11]. The deployment of the RSU and the measured communication distance in Satory test track are depicted in Figure 26.

[11] AutoMate Deliverable 2.5: "Metrics and Experiments for V & V of the driver, vehicle and situation models in the 3rd cycle"

Figure 26: Communication distance measured in Satory test track

Furthermore, field test was performed in Budapest as well. The selected place was the Óceánárok utca, which can be classified as rural-suburban environment located in north Budapest. The preparation and the cars used for the field test are visible in Figure 27. Similarly, the placement of RSU and measured distances are illustrated in Figure 28.

Figure 27: Antenna placement and cars used for the field test in Óceánárok utca

Figure 28: Measured communication distances from the position of RSU (it is located in the same place): 634m northward (left) and 526m southward (right)

In the previous cases the DENM-RWW message was configured as it was presented in D2.5.

Then, log files of previous field test around BIT's office was also used to determine the communication distance. The office is located in Ürömi utca,

Budapest, which is an urban canyon from wireless communication perspective. Here, we measured around 400m, where message was still transmitted between the devices. This measurement is illustrated in Figure 29, while Figure 30 shows the visualization framework including the sign of road works ahead. Finally, the reliability of the communication was also checked. During the field tests there was no incorrect message received from the RSU. That is probably because the Geonetworking layer or the MAC layer of the protocol stack discard any wrong packets.

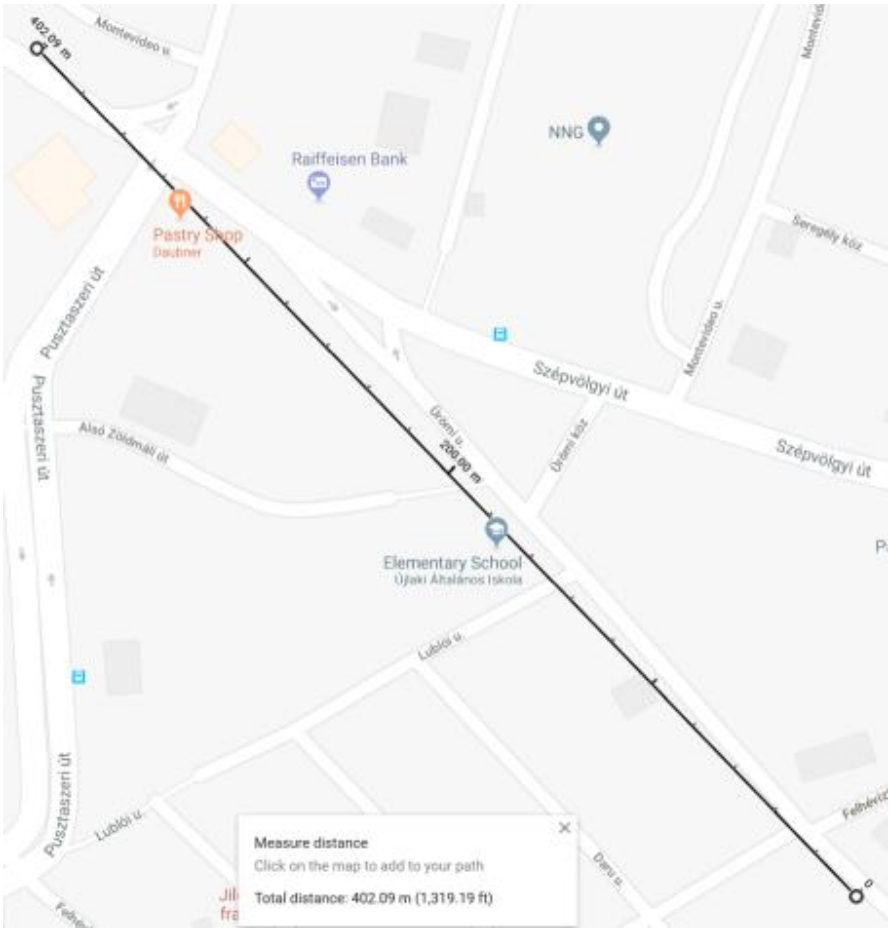


Figure 29: Communication distance measured in urban environment

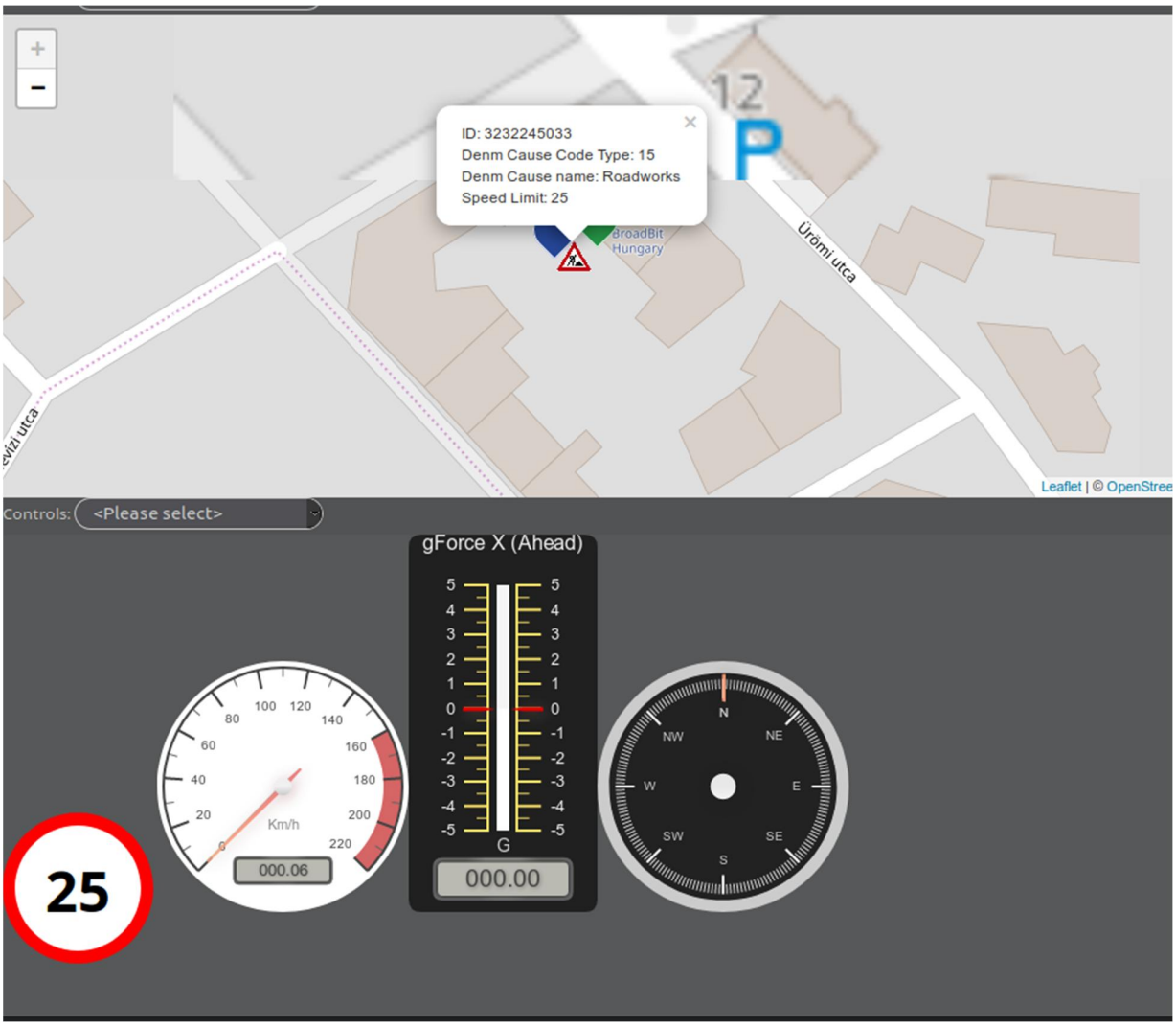


Figure 30: Part of the visualization framework. The upper section shows the road works warning sign with some additional information. The lower section includes the instruments with speed limit sign in the road work area.

4.3 E2.1 – Driver Intention Recognition

Following the plans described in deliverable D2.5 "Metrics and Experiments for V&V of the driver, vehicle and situation models in the 3rd cycle", validation of the driver model for intention and behaviour recognition was performed using sets of independent test data D_{Test}, representing annotated "ground-truth" time-series of manual driving on rural roads, akin to the Peter scenario, and time-series of manual driving in the vicinity of roundabouts, akin to the Eva scenario, using a variety of metrics, extending the formal requirements for the technical validation of E2.1 as stated in deliverable D1.5 "Definition of framework, scenarios and requirements incl. KPIs & Baseline for 3rd cycle". A summary of relevant requirements is shown in Table 5.

Table 5: Requirements and metrics used for the technical validation of E2.1

Requirement	Metric	Success criteria
R_EN2_model2.6	Correct rate of the classification	>80%

4.3.1 Driver intention recognition in rural road scenarios

Section 3 presented two alternative models for driver intention recognition on rural roads developed during the third cycle, referred to as M_1 and M_2, with M_2 allowing for interdependencies between continuous variables. Solely focusing on intention recognition, both M_1 and M_2 can be used for intention recognition in *automatic mode*, by inferring $P(I^t|l^{1:t}, \boldsymbol{c}^{1:t})$, or for intention recognition in *manual mode*, by inferring $P(I^t|l^{1:t}, \boldsymbol{c}^{1:t}, \boldsymbol{e}^{1:t})$.

4.3.1.1 Validation process

To recapitalize the overall validation process and metrics used for this validation cycle, let D_{Test} be composed by a number of m trials, where each trial is a time-series consisting of a number of $n_j, j = 1, \dots, m$ *complete* data samples $s_j^k = \left(i_j^k, b_j^k, l_j^k, d_j^k, o_j^k, \boldsymbol{c}_j^k, \boldsymbol{e}_j^k\right), k = 1, \dots, n_j$. The necessary annotations for the hidden intentions, behaviours, desires, and opportunities were obtained during post-processing (for a more detailed description, we refer to Section 4.3.1.2).

As in the previous cycles, we are primarily interested in the performance of the model concerning intention recognition when the automation is in control, therefore neglecting the influence of the component for behaviour recognition. As such, for each sample s_j^k, we used the models M_1 and M_2 to infer a filtered belief state over the intentions $P\left(I_j^k | l_j^{1:k}, \boldsymbol{c}_j^{1:k}\right)$, given all available sensory input in the resp. time-series up to the sample. The output of the models was then defined as the most probable target lane intention

$$i_{j,out}^k = \arg\max_{\mathrm{i}} P\left(I_j^k = i | l_j^{1:k}, \boldsymbol{c}_j^{1:k}\right).$$

For the assessment of intention recognition, the (annotated) "true" and predicted target lane intentions i_j^k and $i_{j,out}^k$ were first mapped onto actual lane change intentions (in that a lane change intention is present if the current lane and the target lane intentions differ) by defining $\hat{\imath}_j^k = \mathbf{1}\left(l_j^k \neq i_j^k\right)$ and $\hat{\imath}_{j,out}^k = \mathbf{1}\left(l_j^k \neq i_{j,out}^k\right)$, where $\mathbf{1}$ denotes the indicator function. Interpreting the existence (we note that the ground truth is based on a manual annotation of the test data and therefore subject to error) of a lane change intention as positive and the absence as negative, we constructed a binary confusion matric for each model, as shown in Figure 31.

		Predicted	
		Positive	Negative
Ground truth	Positive	TP	FN
	Negative	FP	TN

Figure 31: Binary confusion matrix.

Following deliverable D2.5 "Metrics and Experiments for V&V of the driver, vehicle and situation models in the 3rd cycle", we then derive the following set of metrics, summarizing different aspects of the performance of the model:

- The *accuracy*, representing the fraction of correctly recognized samples among all samples, defined as

$$ACC = \frac{TP + TN}{TP + FP + FN + TN}.$$

- The *precision*, representing the fraction of correctly recognized intentions among all predicted intentions, defined as

$$Precision = \frac{TP}{TP + FP}.$$

  A high precision indicates that the model only recognizes intentions if there truly exists an intention.

- The *recall* (also known as sensitivity or true positive rate (TPR)), representing the fraction of correctly recognized intentions over the total amount of true intentions, defined as

$$Recall = \frac{TP}{TP + FN}.$$

  A high recall indicates that most of the intentions are recognized as such.

- The harmonic mean of precision and recall, the traditional F-measure or balanced *F-score*, defined as

$$F\text{-}score = 2\frac{Precision \cdot Recall}{Precision + Recall}.$$

- And, for the sake of completeness, the *False Positive Rate* (FPR), defined as

$$FPR = \frac{FP}{FP + TN}.$$

Requirement R_EN2_model2.6 states a required correct rate of classification $\geq 80\%$. When interpreted strictly, the correct rate of classification refers to the accuracy, representing the fraction of correctly recognized samples among all samples. However, as the accuracy involves both the correct recognition of the presence of an intention and the correct recognition of the absence of an intention (which are usually dominating in amount), we extend the interpretation of R_EN2_model2.6 to both precision and recall, which focus on the actual recognition of the presence of intentions.

As described in deliverable D2.5 "Metrics and Experiments for V&V of the driver, vehicle and situation models in the 3rd cycle", in addition to the classification performance, we are interested in assessing the runtime performance, in terms of execution times, of the driver intention recognition. Although no formal requirement exists to limit the potential execution time for each inference, driver intention recognition on rural roads is intended to provide an output at a frequency of 20Hz, i.e., every 50ms. Given that the intention recognition must share computational resources with other systems, we aim at an average execution time $\leq$ 10ms.

Using an empirical approach for assessing the runtime performance for driver intention recognition on rural roads, we measured the average execution time required for performing inferences of interest in the proposed models M_1 and M_2.

4.3.1.2 Datasets used

For the development and validation of the probabilistic models for driver intention recognition on rural roads, OFF, ULM, and HMT conducted a dedicated simulator study in the OFF driving simulator, as described in detail in deliverable D2.5 "Metrics and Experiments for V&V of the driver, vehicle and situation models in the 3rd cycle".

During data preparation, each data sample was first *manually* annotated with the shown driving behaviour (LCL, LCR, or LK) based on visual judgement of the traffic situation and key measurements, such as e.g., heading angle, lateral deviation, and steering wheel angles, using an editor to visualize the recorded data (Figure 32).

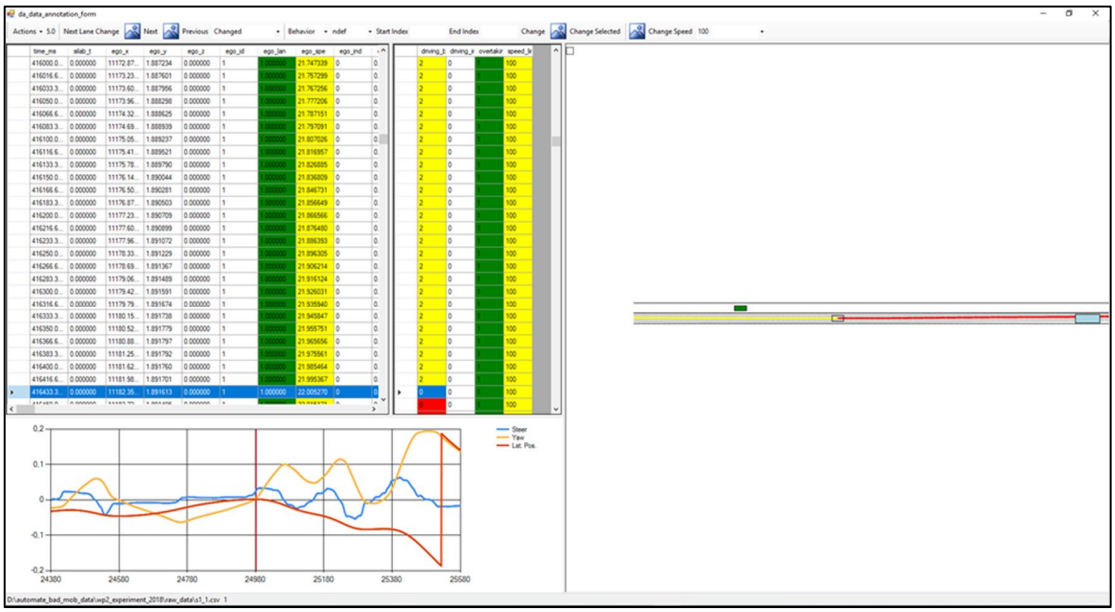


Figure 32:Screenshot of an internal tool used for the annotation of experimental data.

Afterwards, each sample was *automatically* annotated with whether the driver intended to drive on the right or on the left lane, based on the provided

definition of intentions in that *a change in the target lane intention is assumed to be formed at least up to one second prior to the annotated beginning of a lane change manoeuvre* (c.f., section 3.3.2.1).

During the experiment, participants had to signal their desire to overtake by pressing a button on the steering wheel. For the annotation process, we automatically labelled the sequence beginning with the press of such button until the actual lane crossing for the final overtaking as the presence of a desire. Furthermore, we labelled each situation in the actual and the target lane differed as such that an opportunity was present and absent otherwise. Lastly, we automatically relabelled LK behaviours in which a desire was present, but no opportunity given as CF.

During post-processing, the trial of a single participant was removed due to a (virtual) crash, possible changing the subsequent behaviour of the participants. As such, the result of the annotation process was a set of 35 coherent time-series of multivariate data for manual driving in the (simplified) rural road scenario. From this annotated experimental data, we selected 24 sequences (approx. 67.5% of the overall data) as a training set D_{Train}, consisting of 2138134 samples or approx. 594 minutes, while the remaining 11 trials, consisting of 1029216 samples or approx. 286 minutes were reserved as a test set D_{Test} for validation.

As described in section 3.3.2.3, the graph structures and parameters of the models M_1 and M_2 have been learned exclusively using the training data D_{Train}. The resulting models were then subsequently validated on the test set D_{Test}. Being temporal models, intended to provide an output each 50ms, we only use every third sample for the actual validation, resulting in an effective test set

D_{Test}, consisting of 343076 samples, covering 286 minutes of driving over each one trial for each of the 16 participants.

4.3.1.3 Results: classification performance

Figure 33 shows the results for intention recognition on rural roads in automatic mode using the model M_1 in terms of a binary confusion matrix and corresponding metrics. To allow an additional interpretation of the results, additional values in brackets denote the corresponding values if we limit the focus on the case, where the driver was located on the right lane, therefore allowing the interpretation of intentions purely as overtaking intentions.

		Predicted Intention		
		Positive	Negative	Total
„True“ Intention	Positive	TP=42616 (21648)	FN=9457 (6669)	52073 (28317)
	Negative	FP=8294 (5350)	TN=282709 (265102)	374601 (270452)
	Total	50910 (26998)	292166 (271771)	N=343076 (298769)

Metric	Value
Accuracy	0.948 (0.960)
Precision	0.837 (0.802)
Recall / TPR	0.818 (0.764)
F-score	0.828 (0.783)
FPR	0.029 (0.020)

Figure 33: Confusion matrix and corresponding metrics of interest for M_1, assuming automatic mode.

As apparent, the model M_1 achieves an overall high accuracy, precision, and recall. For comparison, Figure 34 shows the results for intention recognition on rural roads in automatic mode using M_2. Compared to the results for M_1, we see an additional improvement of all metrics.

		Predicted Intention		
		Positive	Negative	Total
„True“ Intention	Positive	TP=44103 (23165)	FN=8060 (5152)	52073 (28317)
	Negative	FP=8082 (5291)	TN=282921 (265161)	374601 (270452)
	Total	52095 (28456)	290981 (270313)	N=343076 (298769)

Metric	Value
Accuracy	0.953 (0.965)
Precision	0.845 (0.814)
Recall / TPR	0.845 (0.818)
F-score	0.845 (0.816)
FPR	0.028 (0.020)

Figure 34: Confusion matrix and corresponding metrics of interest for M_2, assuming automatic mode.

Given these results, we can conclude the requirement R_EN2_model2.6 to be fulfilled for both M_1 and M_2 in automatic mode, for accuracy as well as precision and recall.

When incorporating the influence of the component for behaviour recognition in manual mode, i.e., using $P(I_j^k | l_j^{1:k}, \boldsymbol{c}_j^{1:k}, \boldsymbol{e}_j^{1:k}) = \sum_{b \in B} P(I_j^k, b^k | l_j^{1:k}, \boldsymbol{c}_j^{1:k}, \boldsymbol{e}_j^{1:k})$ instead of $P(I_j^k | l_j^{1:k}, \boldsymbol{c}_j^{1:k})$, we are able to slightly improve the accuracy, precision, and false positive rate, but slightly decrease the recall and (consequently) F-score for both M_1 (Figure 35) and M_2 (Figure 36). Comparing Figure 33 and Figure 35 for M_1, resp. Figure 34 and Figure 36 for M_2, we see that using $P(I_j^k | l_j^{1:k}, \boldsymbol{c}_j^{1:k}, \boldsymbol{e}_j^{1:k})$, results in a decrease of false positives for the price of a (relatively higher) increase of false negatives. This can be interpreted as the models being more conservative when considering the behavioural effects, requiring the human behaviour to match the expected effects of a potential intention before classifying a situation as an overtaking intention.

		Predicted Intention		
		Positive	Negative	Total
„True“ Intention	Positive	TP=40817 (19904)	FN=11256 (8413)	52073 (28317)
	Negative	FP=6881 (4269)	TN=284122 (266183)	291003 (270452)
	Total	47698 (24173)	295378 (274596)	N=343076 (298769)

Metric	Value
Accuracy	0.947 (0.958)
Precision	0.856 (0.823)
Recall / TPR	0.784 (0.703)
F-score	0.818 (0.758)
FPR	0.024 (0.016)

Figure 35: Confusion matrix and corresponding metrics of interest for M_1, assuming manual mode.

„True“ Intention	Predicted Intention: Positive	Predicted Intention: Negative	Total
Positive	TP=41339 (20459)	FN=10734 (7858)	52073 (28317)
Negative	FP=5445 (2908)	TN=285558 (267544)	291003 (270452)
Total	46784 (23367)	296292 (275402)	N=343076 (298769)

Metric	Value
Accuracy	0.953 (0.964)
Precision	0.884 (0.876)
Recall / TPR	0.794 (0.722)
F-score	0.836 (0.792)
FPR	0.019 (0.011)

Figure 36: Confusion matrix and corresponding metrics of interest for M_2, assuming manual mode.

Given these results, we can conclude the requirement R_EN2_model2.6 to be fulfilled for both M_1 and M_2 in manual mode, for accuracy and precision, but nor for recall.

When compared to the results obtained during the second cycle (c.f. deliverable D2.4 "Sensor Platform and Models including V&V results from 2nd cycle"), we have achieved a major improvement in all metrics considered. Although this improvement may be owed to the different experimental data obtained in a more controlled simulator experiment, we believe that we managed to improve on the theoretical problems of the driver models in the second cycle. In the remainder of the project AutoMate, we will test this assumption by adapting the models for driver intention recognition on rural roads to the experimental data used in the second cycle.

4.3.1.4 Results: runtime performance

For an empirical approach for measuring the computational performance of the driver intention recognition on rural roads, we measured the average execution time required for performing inferences of interest in the proposed models M_1 and M_2.

Execution times were calculated as the average time required when performing inferences for all 343076 samples comprising the test set using an i7-6700 CPU @ 3.40GHz, 16GB desktop computer, running a Microsoft Windows 10 64-Bit operation system. The algorithms were compiled as 64-Bit applications using Visual Studio 2017 and we measured individual execution times in nanoseconds using the high_resolution_clock provided by the std::chrono library. We note, that these results measure the necessary time for performing inferences in isolation, and do not include the necessary time for the interpretation of the sensor data or updating any graphical user interfaces.
The results are shown in Table 6. As apparent, the average execution time for driver intention recognition is well below 1ms even for all considered models and inference queries, which fits well into the intended frequency of execution of 20Hz and is well below the aim of an average execution time ≤ 10ms.

Table 6: Average computation time for the prediction of the temporal and spatial evolution for a single object for different number of vertices per corner polygon.

Model	Inference query	Average inference time (ms)
M_1	$P\left(l_j^k \mid l_j^{1:k}, \boldsymbol{c}_j^{1:k}\right)$	0.131
M_1	$P\left(l_j^k \mid l_j^{1:k}, \boldsymbol{c}_j^{1:k}, \boldsymbol{e}_j^{1:k}\right)$	0.409
M_2	$P\left(l_j^k \mid l_j^{1:k}, \boldsymbol{c}_j^{1:k}\right)$	0.109
M_2	$P\left(l_j^k \mid l_j^{1:k}, \boldsymbol{c}_j^{1:k}, \boldsymbol{e}_j^{1:k}\right)$	0.360

4.3.1.5 Privacy

Finally, addressing requirement R_EN2_model2.7, we note that the probabilistic models and algorithms for driver intention recognition do neither hold, retrieve nor process any personal data of the driver in a not anonymized way.

4.3.2 Driver intention recognition in roundabout scenarios

In the third cycle we extended the model to one more driving intention in addition to overtaking another car. Eva scenario was chosen to reflect the driver intention in entering roundabouts dealing with the questions: When are the proper times to enter the roundabouts? Which traffic situations were taken by the human driver to enter the roundabouts?

4.3.2.1 Experiments for data gathering

To collect the data for driver intention recognition model in the third cycle, the Automate partners HMT and OFF conducted experiments in OFF driving simulator. The goal of these experiments was to study the intention of the driver in entering to the roundabouts. Therefore, different traffic situations were presented in driving simulator to detect the proper situation in which the driver would intend to enter to the roundabouts.

4.3.2.1.1 Scenario

The driving scenario containing several instances of two distinct urban roundabouts (with a general speed of 50 km/h) were used to identify the intention of driver in entering to the roundabout. The two distinct roundabouts were initially suggested and designed by the Automate project partner REL: A small roundabout with approximate diameter of 10 m and a big one with approximate diameter of 40 m.

The Automate project partner OFF, adapted these roundabouts and build a driving scenario which contained 8 instances of big roundabouts and 7 instances of small roundabouts. In the driving scenario these instances were connected to each other with urban roads. Extra instances of rural roads (with

a maximum speed of 100 km/h) were used between groups of three roundabouts instances, producing a smoother and less boring driving scenario. In total 4 tracks of rural roads were used in each scenario.

In order to study driver intention to enter the roundabouts, various traffic situations were presented in each of the roundabout's instances. The maximum number of 4 vehicles were used to simulate different traffic situation, altering following factors that potentially influence driver intention:

The gap size between vehicles driving through the roundabout:

- For the big roundabouts the gap size varied between 20 and 100 m with approximate values of {20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 80, 85, 95} meters
- For the small roundabout the gap size varied between 10 m as the minimum distance between two vehicles allowed in the OFF driving simulator, and 50 m, including approximate gap sizes of {10, 15, 20, 25, 30, 35, 40, 45, 50} meters

The speed of vehicles driving inside the roundabout:

- speed values varied among {36, 43, 47, 50} km/h for the big roundabout
- and {15, 18, 25} km/h for the small roundabouts

4.3.2.1.2 Participants and procedure

The experimental sessions were announced in local university (University of Oldenburg). 25 participants with valid driving licenses were participated in the study. 4 of the participants felt motion sickness and could not continue the experiments. Therefore, the data collected from 21 participant (12 males and

9 females) was used for modeling. The participants have an average age of 26 years old, the average driving mileage of 14014 km per year and an average driving experiences of 8 years.

After reading the handout of the instruction and filling the consent form, the participants performed a training session in which they drove approximately 10 minutes to get used to the vehicle control in the driving simulator. Then the participants drove through the scenario two times. Each time took about 12 to 15 minutes and the participants had a break of about 5 to 10 minutes between the two sessions. At the end the participants filled the questionnaire. The whole experiment took about 1 hour for each participant and they received a compensation of 10 Euro for their participations.

4.3.2.1.3 Data preparation

Because we were interested to study the intention formation in entering the roundabouts, we focused on the parts of the data related to entering to the roundabouts. Specifically, we extracted the pieces of the data from the time where the TeamMate vehicle locates in the vicinity of the roundabout (30 meters before entering the roundabout) until the time which the TeamMate vehicle locates inside the roundabout lane. The remaining part of the data was removed without further processing.

To annotate the data, we observed the speed and acceleration pattern of the participants in the vicinity of the roundabouts. Approaching to the roundabouts, the participants reduced their speed and evaluated the traffic situation inside the roundabouts. Then they either entered directly to the roundabout with a reduced speed or they further reduced their speed until they stopped fully before the roundabout and were waiting for a proper

moment to enter to the roundabout. As soon as they evaluated the traffic situation as "probable" proper to enter the roundabout they slightly accelerated (push the gas pedal gently) to be in a "start state". At this point they perform their final evaluation, and either accelerated more and entered to the roundabout (entrance behaviors), or they reduced their speed and stopped again and waited for the next proper moment. For annotating the data, we first searched for the entrance behavior, then we found the last slight push of the gas pedal before entrance behavior. The intention set to one as intended to enter to the roundabouts, for the whole duration of start state until entering to the roundabouts. The intention annotated to zero as not intended to enter to the roundabout, for the time before start state. In the case of direct entrance of TeamMate vehicle to the roundabouts, the intention set to intended to enter to the roundabout for the whole data section.

4.3.2.2 Validation process

To validate the model, we inferred a filtered belief state over the intentions $P\left(I_j^k|\boldsymbol{o}_{I_j}^{1:k}\right)$, given all available sensory input in the test time-series (D_{Test}) for each trial up to the sample. Where $n_j, j = 1, \dots, m$ denotes the number of data samples and m denotes the number of test trials. $d_j^k = \left(i_j^k, \boldsymbol{o}_{I_j}^{\ k}\right), k = 1, \dots, n_j$, is annotated considering the speed and acceleration pattern of the TeamMate vehicle (described in section 4.3.2) with the assumed correct intention i_j^k.
As such, the output of the model was then defined as the most probable driver intention

$$i_{j,out}^k = \arg\max_{\mathrm{i}} P\left(I_j^k = i|\boldsymbol{o}_{I_j}^{1:k}\right)$$

For the assessment of intention recognition, the (annotated) "true" and predicted intentions i_j^k and $i_{j,out}^k$ were used. Interpreting the existence of intention to roundabouts as positive and the absence as negative, we constructed a binary confusion matric for each model, as shown in Figure 31, and derived the metrics of accuracy, precision, recall, F-score, and FPR (see Section 4.3.1.1) based on the confusion matrix.

4.3.2.3 Datasets used

Each driving scenario (trial) contained 8 instances of big roundabout and 7 instances of the small roundabouts. We collected two trials from each of 21 participants and extracted and annotated the entrance sections to the roundabouts as explained above.

From every 4 trials, we used 3 trials for training data, 75% of the experimental data, and one trial for testing, 25% of the experimental data.

In total training data D_{Train} for small roundabouts consisted of 106018 samples (approx. 30 minutes), and the test data D_{Test} contained 35806 samples (approx. 10 minutes). The total training data D_{Train} for big roundabouts consisted of 116875 samples (approx. 32 minutes), and the test data D_{Test} contained 34876 samples (approx. 10 minutes).

4.3.2.4 Results

As explained in the section 4.3.2.1, the driving scenario for intention recognition contains two types of roundabouts: a small roundabout with a diameter of approximately 10 m and a big one with a diameter of approximately 40 m. The difference between the diameters result in different range of the speeds for alter vehicles inside the roundabout. Moreover, the

gap size between vehicles were spanning a smaller range in the small roundabouts.

Therefore, we analysed the data for each roundabout separately to be able to interpret the outcomes of the models and to better understand the distributions of variables. Once the roundabout model is understood, a single model can be trained using the datasets from both types of the roundabouts. Such that, we trained two models for big roundabouts, one focusing on the traffic inputs and refraining the speed of the TeamMate Vehicle and including both traffic inputs and the speed of the TeamMate vehicle. Similarly, two models were learned for the small roundabouts. The details of the traffic input variables are explained in detail in section 3.3.3.2.

Speed of the vehicle is regulated after formation of the intentions and could be considered as an effect of intention; however, the speed of the vehicle plays a role during making the decision to enter the roundabout. If the vehicle has an initial speed compared to the full stop state before roundabouts, it could enter faster to the roundabout, indicating that the smaller gap size could be used to enter the roundabout. However, the same driver would require longer time, if he was in stop state (maybe because of a heavy traffic situation at a while before).

As a reminder, the resulting structures for the small roundabouts are shown in Figure 24, section 3.3.3.2. The graph structures depict the important factors chosen by the model to identify the intention of the driver. In the model for the small roundabouts with traffic inputs, the presence of the alter vehicles inside the roundabouts and their distances to the TeamMate vehicle was chosen as important factors to identify the intention of the driver. The model

results in terms of a binary confusion matrix and corresponding metrics are shown in Figure 37.

		Predicted Intention		
		Positive	Negative	Total
„True“ Intention	Positive	TP=6377	FN=5408	11785
	Negative	FP=1923	TN=22098	24021
	Total	8300	27506	N=35806

Metric	Value
Accuracy	0.795
Precision	0.768
Recall (TPR)	0.541
F-score	0.635
FPR	0.080

Figure 37: Confusion matrix and corresponding metrics of interest for the model based on the traffic input from other vehicles in the small roundabouts.

The model has an acceptable accuracy and precision, and could reliably identify the true negative values, producing a very low false positives rate. However, the number of false negatives is high and therefore the recall is relatively low. Using the speed as an additional input to the model, the results have been improved (Figure 38), but still the number of false negatives is relatively high.

		Predicted Intention		
		Positive	Negative	Total
„True“ Intention	Positive	TP=7891	FN=3894	11785
	Negative	FP=2475	TN=21546	24021
	Total	10366	25440	N=35806

Metric	Value
Accuracy	0.822
Precision	0.761
Recall (TPR)	0.669
F-score	0.712
FPR	0.103

Figure 38: Confusion matrix and corresponding metrics of interest for the model based on the traffic input of other vehicles in the small roundabouts and additional input speed input of the TeamMate vehicle.

This could be explained by the larger sample size for the true negatives (24021 samples shown in Figure 37) compared to true positive values (11785 samples, Figure 37). Training dataset has also similar proportion of the true

negatives compared to true positives and therefore the model has been trained with a relatively small sample sizes of true positives. Therefore, the recall metric could be further improved by using a larger dataset.

The resulting structures for the big roundabouts are shown in Figure 23, section 3.3.3. In the first model for the big roundabouts where only traffic inputs were considered, the time headway of the behind vehicle in the roundabouts was chosen by the model as an important factor but not the distance variable. Compared to the structure of the small roundabouts, the distance of the front vehicle was chosen in this model while this variable was not chosen as important factor and instead the distance of the behind second vehicle in the roundabouts was chosen. Indicating that, the model of big roundabouts takes the traffic situation of the behind vehicles to account more than the model for the small roundabouts. The results for this model are shown in Figure 39.

		Predicted Intention		
		Positive	Negative	Total
„True“ Intention	Positive	TP=11435	FN=6012	17447
	Negative	FP=2533	TN=14896	17429
	Total	13968	20908	N=34876

Metric	Value
Accuracy	0.754
Precision	0.818
Recall (TPR)	0.655
F-score	0.727
FPR	0.145

Figure 39: Confusion matrix and corresponding metrics of interest for the model based on the traffic input from other vehicles in the big roundabouts.

This model has higher precision and recall than the model for small roundabouts. The number of true positive samples in test set is 17447 samples, approximately half of the samples. Still the model has a relatively high false negatives and the model could be further improved. Including the

TeamMate speed as an additional input to the model improved the results for big roundabouts, as expected (Figure 40).

		Predicted Intention		
		Positive	Negative	Total
„True“ Intention	Positive	TP=12657	FN=4790	17447
	Negative	FP=1717	TN=15712	17429
	Total	14374	20502	N=34876

Metric	Value
Accuracy	0.813
Precision	0.880
Recall (TPR)	0.725
F-score	0.795
FPR	0.098

Figure 40: Confusion matrix and corresponding metrics of interest for the model based on the traffic input from other vehicles in the big roundabouts and additional speed input of the TeamMate vehicle.

4.4 Driving Task Model Results

The validation process of Driving Task Model is described hereafter, whose procedure is equal to the one of the second cycle. Here, we have added a results section.

From a preliminary analysis, we can conclude that the basic model of task execution was the same for the participants in ULM and DLR. We conclude from this that DriveGOMS approach for collecting, organising and analysing data is a helpful, valid addition to the traditional tools available for Human Factors researchers. The model and the other insights from the study will aid the development and evaluation of enabler E6.1.

Due to the limited space available here we will constrict the report on the results on the basic model that was derived from the scenario. We are planning on making the more detailed results known to a wider audience via publications. The model can be seen in Figure 41. It was exhibited by all subjects, both in the ULM and the DLR study.

Two observations are important. First, subjects did have the opportunity to play the game of Solitaire the entire time if they wished to do so. All subjecs reported the scenery to be rather boring after a short while (latest towards the middle of the first block). Therefore, even if they did not like Solitaire too much, they all did interact with the tablet after a short while, for extended periods of time. This is exemplified by the goal “play”.

Second, subjects did not need to observe the road or scenery actively, because of the warning sound that chimed when approaching the slower lead vehicle up to an “ACC” distance. However, all subjects chose to intermittently look up (goal “update situation representation”). When asked why after the study’s end, subjects reported that they “wanted to see what was going on” and

"wanted to see where the slower lead vehicle might be". Latest during the second trial, they all did report to expect that lead vehicle, as well as being extremely annoyed by it.

However, usually it did take some time, until the ego vehicle approached the slower lead vehicle. During this time, subjects would return their attention to the game, instead of keeping on tracking the lead vehicle. This is represented by "decide-wait". They knew the vehicle was coming, but had to wait for the warning sound until they could act. Since there was nothing else left to do (a sentence uttered by all of the 12 participants regarding why they did return their attention to the game), they continued playing Solitaire.

Next, after having been alarmed to the presence of the slower lead vehicle, participants started to look actively for oncoming traffic, and also gauged the distance of the visible road. Due to the winding and hill climbing nature of the road this could vary. The decision to eventually overtake could be modelled by a function of the distance of visible road, the guessed differential speed of the oncoming traffic, and the likelihood of the presence of oncoming traffic. There were situations where obviously all subjects chose to overtake: If the road was visible far ahead, with no sign of oncoming traffic. Otherwise, subjects did sometimes wait for oncoming traffic, and overtake directly after it had passed. Once the basic decision had been made to wait for a good opportunity to activate the overtaking maneuver ("wait-for-gap"), subjects never returned to playing. If the decision ("decide-overtaking") for the "GO" was made, subjects would activate the left indicator, and then share their attention between observing the truck as they were passing it and possible oncoming traffic, while (at least early in the study) monitoring the system. When seeing the truck in

the center mirror, subjects would activate the indicator to the right, let the car center itself in the lane, and quickly go back to playing.

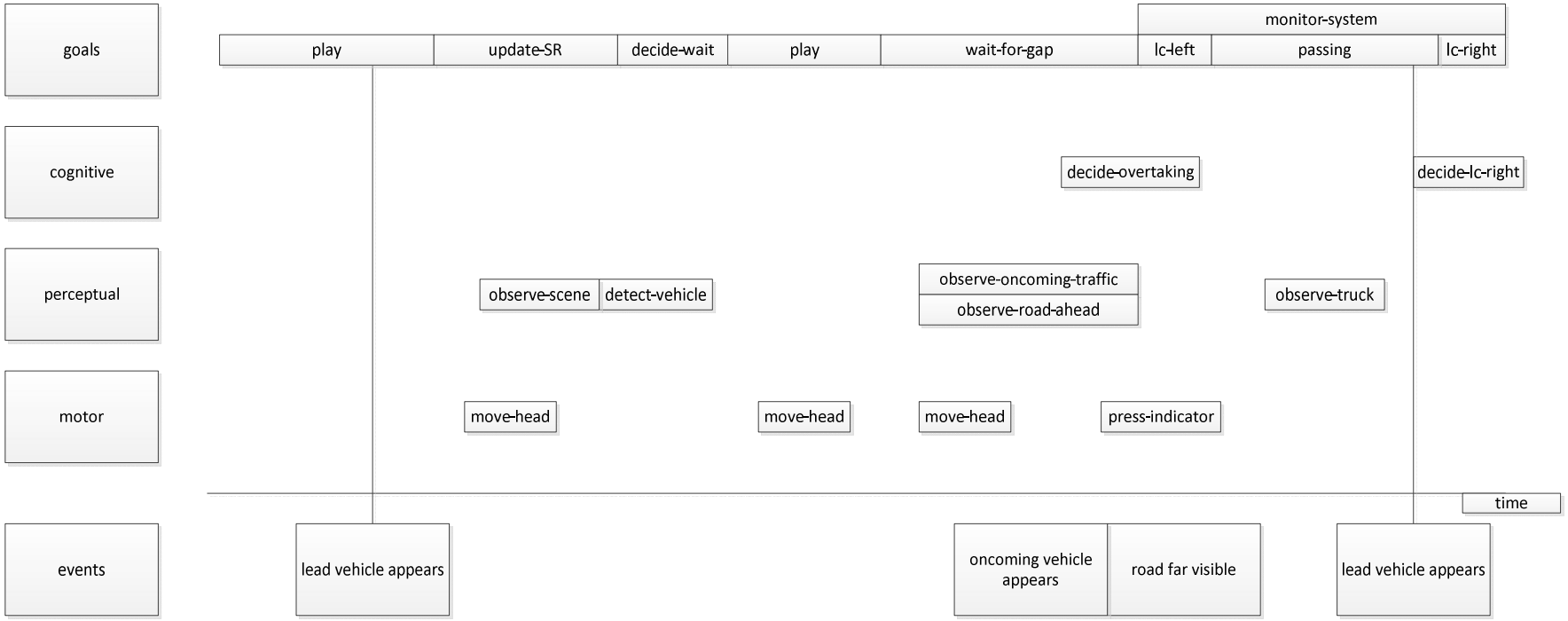


Figure 41: Basic model for the Peter scenario.

4.5 E3.1 – Situation and vehicle model

4.5.1 Predicting the future evolution of the traffic scene

Following the plans described in deliverable D2.5 "Metrics and Experiments for V&V of the driver, vehicle and situation models in the 3rd cycle", validation of the prediction of the temporal and spatial evolution of the traffic scene was performed using a set of independent test data, representing ground truth time-series of traffic situations, using a variety of metrics, extending the formal requirements for the technical validation of E3.1 as stated in deliverable D1.5 "Definition of framework, scenarios and requirements incl. KPIs & Baseline for 3rd cycle". A summary of relevant requirements is shown in Table 7.

Table 7: Requirements and metrics used for the technical validation of E3.1

Requirement	Metric	Success criteria
R_EN3_model1.3 and R_EN3_model1.4	Correct rate of the prediction	>90%

4.5.1.1 Validation process

To recapitalize and concretize the overall validation process and metrics used, let D_{Test} denote the test data, composed by a number of m trials, where each trial $j, j = 1, \dots, m$ is a time-series consisting of a number of n_j, data samples $d_j^k = \left(\boldsymbol{s}_{v_1}^k, \dots, \boldsymbol{s}_{v_{n_V}}^k\right), k = 1, \dots, n_j$ and a map M. For each sample d_j^k, and each object $v \in$

V, we infer the most probable behaviour hypothesis $h_{max} \in \{LF, LC\}$ and predict a sequence of future states $p\left(\boldsymbol{S}_{j,v}^{k+i\Delta} | h_{\max_{j,v}}^{k}\right), i = 1, \ldots, \eta_{max}$.

Concerning the validation of the prediction of the evolution of the traffic situation, it is most important that the predicted regions encompass the true future location of the vehicle. As a metric to validate the performance, we therefore chose the concept of a "correct classification rate" as the ratio of correct predictions and the number of total predictions. Let $0 < \delta < 1$ denote an arbitrary threshold, we can define a region that covers $100(1-\delta)\%$ of the probability density of a belief state $p\left(\boldsymbol{s}_{j,v}^{k+i\Delta t} | h_{\max_{j,v}}^{k}\right)$. For each predicted state $p\left(\boldsymbol{S}_{j,v}^{k+i\Delta t} | h_{\max_{j,v}}^{k}\right), i = 1, \ldots, \eta_{max}$, we then check whether the true state $\boldsymbol{s}_{j,v}^{k+i\Delta t}$ of object $v \in V$ is located outside of this region. Denoting such an occurrence as a failure and resp. as a success otherwise, the correct classification rate CR_{δ}^{i} is defined as

$$CR_{\delta}^{i} = \frac{\#_s}{\#_s + \#_f},$$

representing the *ratio of successes* $\#_s$ *and the sum of successes* $\#_s$ *and failures* $\#_f$ for a temporal prediction horizon $i\Delta$ and a specific level of δ for assessing the quality of the prediction of the temporal and spatial evolution of the traffic scene. For the actual validation, we abstract from the predicted velocity, acceleration, and yaw-rate, which are not used for online risk assessment, and instead focus on the valid prediction of the location and pose $p\left(x_{j,v}^{k+i\Delta t}, y_{j,v}^{k+i\Delta t}, \theta_{j,v}^{k+i\Delta t} | h_{\max_{j,v}}^{k}\right)$ and, for the sake of comparison, on the valid prediction of the location $p\left(x_{j,v}^{k+i\Delta t}, y_{j,v}^{k+i\Delta t} | h_{\max_{j,v}}^{k}\right)$ alone.

The metric is used to assess the fulfilment of requirements R_EN3_model1.3 and R_EN3_model1.4, stating that the "integrated model must predict possible

evolutions of the traffic situation in respect to potential interventions of the driver" (R_EN3_model1.3), resp. "[…] potential interventions of the automation" (R_EN3_model1.4) with a correct rate of the prediction above 90% to be fulfilled. For a perfect prediction and a region that encloses $100(1.0-\delta)\%$ of the probability mass, we would, in the perfect case, expect a failure-rate of $100\delta\%$. As such, we will treat the requirements as fulfilled, if the ratio of correct predictions is above $90(1.0-\delta)\%$ for each prediction horizon $i\Delta$ and level δ independently.

Newly introduced in deliverable D2.5 "Metrics and Experiments for V&V of the driver, vehicle and situation models in the 3rd cycle", we planned to assess the volume of the prediction area as an supplementary measure for assessing the quality of the traffic prediction, with smaller volumes, which can be interpreted as more certain predictions, being preferred over bigger volumes. For this, we calculated the mean area $\bar{A}_{\delta}^{i}$ derived from an polygonal approximation (using a number of 100 vertices) of the belief state over the location $p\left(x_{j,v}^{k+i\Delta t}, y_{j,v}^{k+i\Delta t} | h_{\max_{j,v}}^{k}\right)$ for a prediction horizon i and a specific level of δ.

Finally, for an empirical approach for assessing the computational performance of the prediction of the evolution of the traffic situation, we measured the runtime performance for the prediction using a temporal step width $\Delta = 1s$ and a maximum number of prediction steps $\eta_{max} = 10$.

4.5.1.2 Dataset for validation

As the validation process requires the knowledge of ground truth, we opted to perform the validation on simulator data. To allow for an easier comparison with the evaluation results obtained during the second cycle, we reused the

same test set D_{Test} as previously described and used in deliverable D2.4 "Sensor Platform and Models incl. V&V results from 2nd cycle".

Due to the test set arising from a simulator study (a detailed description of the experiment is also provided in deliverable D2.4 "Sensor Platform and Models incl. V&V results from 2nd cycle") in which the traffic flow was automatically controlled by a traffic simulation, the resulting behaviour of traffic participants in the vicinity of the TeamMate vehicle is highly predictable and potentially unrealistic, leading to overly optimistic results. For a more realistic assessment of humanly controlled traffic participants, we therefore primarily perform our validation for the humanly controlled "TeamMate" vehicle.

As D_{Test} provided ground-truth data, while the traffic prediction requires belief states with associated uncertainties, we needed to transform the ground-truth data into belief states. Let $s_{j,v}^k = (x_{j,v}^k, y_{j,v}^k, \theta_{j,v}^k, v_{j,v}^k, a_{j,v}^k, w_{j,v}^k)$ denote the ground truth of the state of a vehicle $v \in V$ in a sample d_j^k, we use the following belief state as a simulated provision of the sensor and communication platform:

$$p(S_{j,v}^k) = N\left(\boldsymbol{\mu} = \begin{pmatrix} x_{j,v}^k \\ y_{j,v}^k \\ \theta_{j,v}^k \\ v_{j,v}^k \\ a_{j,v}^k \\ w_{j,v}^k \end{pmatrix}, \boldsymbol{\Sigma} = \begin{pmatrix} 0.1 \\ 0.1 \\ 0.01 \\ 1.0 \\ 0.01 \\ 0.01 \end{pmatrix}^2 \boldsymbol{I}_6\right).$$

4.5.2 Results: Prediction performance

We performed the validation for five different levels of δ, $\delta_{0.5} = 0.5$, $\delta_{0.25} = 0.25$, $\delta_{0.1} = 0.1$, $\delta_{0.05} = 0.05$, and $\delta_{0.01} = 0.01$ using a temporal step width $\Delta t = 1s$ and a maximum number of prediction steps $\eta_{max} = 10$.

As the primary result of the validation, Table 8 shows the correct classification rate CR_{δ}^{i} for the humanly controlled vehicle for the different temporal intervals i (corresponding to a temporal interval $[k+(i-1)s{:}\,k+is]$) and different levels of δ. Bold values indicate that the ratio is above the required $90(1.0-\delta)\%$. We report both the results focusing on location and pose and focusing solely on location. To provide a more intuitive understanding of the results, we additionally report the average two-dimensional Euclidean distance (AED) between the ground truth and the mean of the predicted.

Table 8: Ratio of successes $\#_s$ and the sum of successes $\#_s$ and failures $\#_f$ for the prediction of the temporal and spatial evolution for human participants, for different prediction horizons i (in seconds) and different levels of δ. Ratios limited to the location are shown in brackets, AED denotes the average Euclidean distance between the true location and the mean of the prediction. Bold values denote that the result is above the required $90(1.0-\delta)\%$.

i	$\#_s+\#_f$	$CR^{i}_{\delta_{0.5}}$	$CR^{i}_{\delta_{0.25}}$	$CR^{i}_{\delta_{0.1}}$	$CR^{i}_{\delta_{0.05}}$	$CR^{i}_{\delta_{0.01}}$	AED
1	405947	**0.920** (**0.978**)	**0.945** (**0.996**)	**0.959** (**0.999**)	**0.966** (**1.000**)	**0.979** (**1.000**)	0.110
2	405627	**0.868** (**0.894**)	**0.903** (**0.932**)	**0.924** (**0.948**)	**0.932** (**0.953**)	**0.944** (**0.961**)	0.558
3	405307	**0.827** (**0.824**)	**0.875** (**0.886**)	**0.900** (**0.915**)	**0.911** (**0.926**)	**0.925** (**0.937**)	1.239
4	404987	**0.789** (**0.763**)	**0.848** (**0.853**)	**0.878** (**0.887**)	**0.891** (**0.901**)	**0.907** (**0.917**)	2.460
5	404667	**0.759** (**0.714**)	**0.827** (**0.823**)	**0.861** (**0.866**)	**0.875** (**0.882**)	**0.892** (**0.901**)	3.779
6	404347	**0.737** (**0.680**)	**0.813** (**0.801**)	**0.847** (**0.849**)	**0.862** (**0.866**)	0.879 (0.886)	5.255

7	404027	**0.723** (**0.659**)	**0.802** (**0.785**)	**0.838** (**0.837**)	**0.851** (**0.855**)	0.869 (0.871)	6.845
8	403707	**0.712** (**0.646**)	**0.793** (**0.773**)	**0.829** (**0.827**)	0.842 (0.844)	0.860 (0.864)	8.527
9	403387	**0.703** (**0.636**)	**0.783** (**0.763**)	**0.820** (**0.817**)	0.834 (0.835)	0.853 (0.855)	10.29
10	403067	**0.695** (**0.628**)	**0.775** (**0.754**)	**0.812** (0.807)	0.827 (0.826)	0.846 (0.848)	12.12

As indicated by bold values, the correct classification rate fulfils the requirement of $CR_{\delta}^{i} > 90(1.0 - \delta)\%$ up to $\eta_{max}\Delta t = 10s$ for $\delta_{0.5}$, $\delta_{0.25}$, and $\delta_{0.1}$ (however only up to $\eta_{max}\Delta t = 9s$ in the two-dimensional case). In contrast, $\delta_{0.05}$ is fulfilled up to $\eta_{max}\Delta t = 7s$ and $\delta_{0.01}$ is fulfilled up to $\eta_{max}\Delta t = 5s$. As such, compared to the second cycle, we can report an increase of $3s$ for $\delta_{0.05}$ and $1s$ for $\delta_{0.01}$.

As supplementary information, Table 9 shows the mean area $\bar{A}_{\delta}^{i}$ (in m^2) of the belief state over the location for the humanly controlled vehicle for the different temporal intervals i (corresponding to a temporal interval $[k + (i-1)s : k + is]$) and different levels of δ.

Table 9: Approximated mean area $\bar{A}_{\delta}^{i}$ of the belief state over the location, for different prediction horizons i (in seconds) and different levels of δ, as the average of # measurements.

i	#	$\bar{A}_{\delta_{0.5}}^{i}$	$\bar{A}_{\delta_{0.25}}^{i}$	$\bar{A}_{\delta_{0.1}}^{i}$	$\bar{A}_{\delta_{0.05}}^{i}$	$\bar{A}_{\delta_{0.01}}^{i}$
1	406267	1.521	3.042	5.052	6.573	10.10
2	406267	3.296	6.593	10.95	14.25	21.90
3	406267	5.507	11.02	18.30	23.80	36.59
4	406267	8.090	16.18	26.87	34.96	53.75
5	406267	11.02	22.04	36.62	47.64	73.23

6	406267	14.34	28.68	47.63	61.97	95.27
7	406267	18.17	36.34	60.37	78.54	120.73
8	406267	22.64	45.28	75.22	97.86	150.43
9	406267	27.90	55.80	92.68	120.58	185.36
10	406267	34.12	68.24	113.35	147.47	226.70

For the sake of comparison with the results reported in deliverable D2.4 "Sensor Platform and Models incl. V&V results from 2nd cycle", Table 10 shows correct classification rate CR_{δ}^{i} and Table 11 shows the mean area $\bar{A}_{\delta}^{i}$ for the other traffic participants, controlled by the driving simulation.

Table 10: Ratio of successes $\#_s$ and the sum of successes $\#_s$ and failures $\#_f$ for automatically controlled traffic participants.

i	$\#_s + \#_f$	$CR_{\delta_{0.5}}^{i}$	$CR_{\delta_{0.25}}^{i}$	$CR_{\delta_{0.1}}^{i}$	$CR_{\delta_{0.05}}^{i}$	$CR_{\delta_{0.01}}^{i}$	AED
1	974201	**0.986** (**0.992**)	**0.988** (**0.994**)	**0.990** (**0.995**)	**0.991** (**0.995**)	**0.994** (**0.995**)	0.130
2	947320	**0.981** (**0.982**)	**0.985** (**0.988**)	**0.986** (**0.990**)	**0.986** (**0.991**)	**0.987** (**0.993**)	0.382
3	920967	**0.976** (**0.974**)	**0.981** (**0.983**)	**0.983** (**0.986**)	**0.984** (**0.988**)	**0.985** (**0.989**)	0.681
4	895101	**0.972** (**0.963**)	**0.979** (**0.982**)	**0.981** (**0.985**)	**0.982** (**0.987**)	**0.984** (**0.988**)	0.985
5	869852	**0.967** (**0.955**)	**0.978** (**0.981**)	**0.980** (**0.985**)	**0.982** (**0.987**)	**0.983** (**0.989**)	1.289
6	845085	**0.962** (**0.944**)	**0.978** (**0.981**)	**0.981** (**0.986**)	**0.982** (**0.988**)	**0.983** (**0.989**)	1.620
7	820710	**0.959** (**0.941**)	**0.977** (**0.981**)	**0.981** (**0.987**)	**0.982** (**0.989**)	**0.983** (**0.990**)	1.989
8	796983	**0.958** (**0.941**)	**0.977** (**0.981**)	**0.980** (**0.987**)	**0.981** (**0.989**)	**0.982** (**0.991**)	2.396

9	773711	**0.958** (**0.939**)	**0.976** (**0.981**)	**0.979** (**0.988**)	**0.980** (**0.989**)	**0.981** (**0.991**)	2.837
10	750924	**0.956** (**0.936**)	**0.975** (**0.981**)	**0.978** (**0.987**)	**0.979** (**0.989**)	**0.980** (**0.990**)	3.295

As apparent, the correct classification rate is for the most part above the corresponding level of δ, therefore fulfilling the requirements of being above $90(1.0-\delta)\%$.

Table 11: Approximated mean area $\overline{A}_{\delta}^{i}$ for automatically controlled traffic participants.

i	#	$\overline{A}_{\delta_{0.5}}^{i}$	$\overline{A}_{\delta_{0.25}}^{i}$	$\overline{A}_{\delta_{0.1}}^{i}$	$\overline{A}_{\delta_{0.05}}^{i}$	$\overline{A}_{\delta_{0.01}}^{i}$
1	1001662	1.520	3.039	5.048	6.568	10.096
2	1001662	3.301	6.602	10.97	14.27	21.93
3	1001662	5.535	11.07	18.39	23.92	36.77
4	1001662	8.166	16.33	27.13	35.29	54.25
5	1001662	11.17	22.34	37.11	48.28	74.22
6	1001662	14.58	29.16	48.43	63.01	96.87
7	1001662	18.48	36.96	61.38	79.86	122.76
8	1001662	22.95	45.90	76.23	99.18	152.47
9	1001662	28.10	56.18	93.31	121.40	186.63
10	1001662	34.02	68.04	113.02	147.04	226.03

4.5.3 Results: Runtime performance

For an empirical approach for measuring the computational complexity of the traffic prediction, we measured the runtime performance for the prediction of the spatial and temporal evolution of the traffic scene using a temporal step width $\Delta t = 1s$ and a maximal number of prediction steps $\eta_{max} = 10$.

Execution times were calculated as the average of 406000 example executions using an i7-6700 CPU @ 3.40GHz, 16GB desktop computer, running a Microsoft Windows 10 64-Bit operation system. The algorithms were compiled as 64-Bit applications using Visual Studio 2017 and we measured individual execution times in nanoseconds using the high_resolution_clock provided by the std::chrono library.

To allow for a better extrapolation to the usually variable number of considered objects n_V in the vicinity of the TeamMate vehicle, we limited the assessment to a single object by only measuring the prediction of the spatial and temporal evolution for the TeamMate itself. We note, that these results measure the execution time of the prediction in isolation, and do not include the necessary time for the interpretation of the sensor data or updating any graphical user interfaces.

The resulting execution time averages on 1.390ms per considered object in the vicinity of the TeamMate vehicle. Extrapolating these results, this would allow for a prediction of the spatial and temporal evolution of $n_V = 7$ traffic participants within the duration of approx. 10ms, which fits well into the intended frequency of execution of 10Hz.

4.5.4 Privacy

Finally, addressing privacy concerns, the algorithms for prediction of the spatial and temporal evolution of the traffic participants do not process or retrieve any personal data of the driver.

5 Conclusions and Outlook

This document presented all the results related to the sensor platform and models including the verification and validation results of the activities from task 2.2 (T2.2) to task 2.5 (T2.5) from the 3rd cycle.

During the first cycle initial models were developed, their performance were validated and limitations were collected. During second cycle the enablers were improved to extend the known limitations and be able to meet the requirements of the defined use cases. Then, these enhanced models were evaluated as well in different ways.

Now, in the third cycle, the WP2 enablers have been improved: each related section of this document describes how (the implementation has been done) and how much (the performances are better). The results show that all the enablers are able to provide useful data for other components of the TeamMate car allowing for the unique feature set of it and are now ready to be integrated in the AutoMate project demonstrators (including both real vehicles and driving simulators).

In the next steps the emphasis will be on integration, as we will describe in the final documents of WP5.