

Personalizing HMI Elements in ADAS using ontology meta-models and rule based reasoning

Yannis Lilis¹, Emmanouil Zidianakis¹,
Nikolaos Partarakis¹, Margherita Antona¹, and Constantine Stephanidis^{1,2}

¹ Institute of Computer Science, FORTH

² Department of Computer Science, University of Crete

{lilis, zidian, partarak, antona, cs}@ics.forth.gr

Abstract. Advanced Driver Assistant systems (ADAS) are receiving increased research focus as they promote a safer and more comfortable driving experience. In this context, personalization can play a key role as the different driver/rider needs, the environmental context and driver's/rider's state can be taken into account towards delivering custom tailored interaction and performing intelligent decision making. This paper presents an ontology-based approach for personalizing Human Machine Interaction (HMI) elements in ADAS systems. The main features of the presented research work include: a) semantic modelling of relevant data in the form of an ontology meta-model that includes the driver / rider information, the vehicle and its HMI elements, as well as the external environment, b) rule-based reasoning on top of the meta-model to derive appropriate personalization decisions, and c) adaptation of the vehicle's HMI elements and interaction paradigms to best fit the particular driver or rider, as well as the overall driving context.

Keywords: Ontology, ADAS system, Personalization, Adaptation, Human Machine Interaction.

1 Introduction

Recently, research in the automotive domain has targeted to improve driving safety through the development of preventive support systems, called Advanced Driver Assistance Systems (ADAS). State of the art systems offering such functionality include adaptive cruise control, automatic emergency braking, lane keeping assist, lane departure warning, traffic jam pilot, dynamic maps, eCall, and driver state monitoring [1].

In this context, a prominent direction for further improving safety and the overall driving experience is to offer *personalized interaction* that takes into account the driver, the vehicle and the driving environment. Adapting the Human Machine Interaction (HMI) elements to fit the driver or rider, the vehicle and the environment is crucial for providing safer driving conditions [49], hopefully limiting the number of serious car and motorcycling accidents. For example, the driver's tiredness, distraction or lack of

experience may affect decision making so as to trigger proactive ADAS decisions earlier. Additionally, the means of delivering warning messages may vary depending on the particular environmental conditions and driving context. For example, in the case where sunlight or headlights of other vehicles compromise the driver's vision, an auditory message should be preferred over a visual one. On the contrary, an auditory warning would be inappropriate for an environment with loud noise, e.g., a motorcycle or a vehicle with open windows, requiring alternative interaction methods, such as haptic signals in combination with visual cues.

Offering such personalized functionality requires storing information about the driver characteristics and preferences, as well as constantly monitoring the state of the driver, the vehicle and the environment. The latter may include information about weather and traffic conditions, digital maps, and V2X communication [2, 16]. To efficiently and effectively organize and process such amount of data towards making personalization decisions requires introducing a semantic knowledge representation. The latter relates to ontologies, a formal way for naming and defining the types, the properties and the interrelationships among the entities of a target domain. Knowledge is typically represented in the Resource Description Framework (RDF) [67] as triplets with the form subject–predicate–object, where the subject and the object are linked with the relationship expressed by the predicate. Data stored using this representation can then be retrieved and manipulated through the SPARQL Protocol and the RDF Query Language (SPARQL) [58]. Furthermore, it is possible to express rules and reasoning logic based on this data using the Semantic Web Rule Language (SWRL) [34], and evaluate such rules and logic through semantic reasoners (e.g. Pellet, Hermit) in order to make appropriate decisions.

A lot of research has focused on ontology-based modelling in the automotive domain in general [4, 7, 8, 11, 14, 19, 25, 31, 32, 33, 43, 47, 51, 57], and for ADAS systems in particular [5, 6, 17, 22, 39, 42, 45, 54, 55, 63, 68, 69], as well as on the personalization of HMI [1, 3, 13, 20, 23, 24, 26, 29, 30, 59, 65]. However, there has been little work that explores the combination of the two fields and adopts an ontology-based approach for delivering personalized HMI elements in ADAS systems. There are existing ontologies that model some aspects of the driver, the vehicle, and the environment, thus offering the basis for personalizing interaction. However, they do not cover all relevant driver aspects such as mental, physiological and emotional state, characteristics, personality and preferences, and lack proper modelling of significant vehicle information such as the available HMI elements. This paper argues that a more comprehensive ontology that covers all relevant driver or rider information, as well as static and dynamic information regarding the vehicle and the surrounding environment, can greatly improve the potential for personalized interaction and enable ADAS systems to offer personalized driving assistance, effectively leading to safer driving and reduced car and motorcycling accidents.

This paper presents an ontology-based approach for delivering personalized HMI elements in ADAS systems. The proposed approach combines the following aspects: a) semantic modelling of relevant data in the form of a meta-model, by extending existing models when appropriate, to gather information regarding the driver or rider, the vehicle and its HMI elements, as well as the external environment; b) performing rule-

based reasoning on top of this meta-model to derive appropriate personalization decisions, and c) using these decisions to adapt both HMI element and interaction modalities to best fit the particular driver / rider and context of use.

2 Background and Related Work

The process of driving a car has not changed significantly during the last 80 years. On the contrary, what has been changing significantly is the integration of electronics and, more recently, computers (e.g., ADAS, telematics, infotainment systems, etc.). To this end, this section presents the main technologies currently leading the automotive industry towards more safe, proactive and ultimately personalized vehicles.

2.1 Advanced Driver Assistance Systems (ADAS)

ADAS systems are developed to help the driver in the driving process in terms of automation features of the vehicle, adaptation of HMI elements and driving safety [16, 48]. The aim of ADAS systems is to avoid collisions and accidents by offering technologies that alert the driver to potential problems, or by implementing safeguards and taking over control of the vehicle. Adaptive features may automate lighting, braking, provide adaptive cruise control, incorporate traffic warnings, alert driver to other cars or dangers, keep the driver in the correct lane, etc. ADAS systems rely on input from multiple data sources, including automotive imaging, LiDAR, radar, image processing, computer vision, and vehicle communication [48, 53]. As reported in [62], the following indicative ADAS systems are available in various production models from a variety of original equipment manufacturers (OEMs): a) autonomous cruise control, b) automotive navigation, c) driver drowsiness detection, d) electronic stability control, e) intersection assistant, etc. ADAS systems apply not only to cars, but also to trucks and buses. In addition, considerable efforts also focus on the development of ADAS systems for powered-two-wheelers, aiming at minimizing the risk of accidents.

2.2 HMI elements and ADAS

In the context of ADAS systems, the HMI elements serve both as a communication bridge between the vehicle and the driver and as a means for the driver to access information and services provided through the smart infrastructure (e.g., Vehicle-to-vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication). Although many of these systems build on advances in diverse technologies, such as vision systems, sensors, and connectivity, the success of ADAS systems relies on the provision of distraction-free methods for interacting with the driver. For instance, advances in touchscreen technology offer more intuitive interaction with passengers and the driver, minimizing the need for embedding a lot of dashboard controls with, in some cases, questionable affordances. However, for ensuring the safety of the driving process itself, the optimal design and deployment of HMI technologies to vehicular systems seem to be a vital aspect already adopted by a variety of OEMs.

In the same direction, research has focused on the provision of novel alternative input and output modalities for HMI elements. HMI input is provided through explicit commands, as well as through analysis of implicit gestures and poses. For instance, touch-free HMI elements promise mechanisms for driver interaction without requiring drivers to move their hands from the steering wheel. HMI output is provided explicitly through visual, acoustic and haptic signals, as well as implicitly using ambient light, background sound and smooth force feedback on the steering wheel, pedals or handle bars. Visual feedback is offered typically through visual displays, like clusters on the dashboard, or through head-up displays (HUD). The latter usually project a virtual image in the windshield of the vehicle helping drivers maintain roadway focus [60, 64]. Auditory warnings are more appropriate than visual ones for urgent situations because they induce a quicker reaction [44]. Finally, haptic interaction can be used on the steering wheel or on the seat. Some studies show that this modality is considered as more appropriate and less annoying than the auditory one [46].

2.3 Personalized interaction with HMI elements in automotive applications

The introduction of HMI elements in the automotive domain has introduced new layers of interaction complexity due to completely changing cognitive models of interaction patterns and expectations. Traditionally, HMI technologies are deployed as monolithic blocks of embedded hardware and software that remain unchanged for the entire lifetime of a vehicle platform. With the advent of multi-modal HMIs, drivers encounter an increasing information flow due to the increasing number of on-board functions (not only related to the driving task) and the massive introduction of ADAS systems. Often, due to their physiological state (tired, absent minded, etc.) and complex traffic environment, drivers are not always capable of perceiving and understanding the plethora of messages produced by the vehicle/system [30]. To this end, the development of HMI technologies needs to be context aware (i.e., driver, vehicle and environmental state) as well as to be adapted to user's characteristics, needs and expectations. Towards this direction, some initial research efforts [24] have targeted the potential of developing a personalized, safe in-car HMI that automatically adapts to the targeted design and interaction concept, as well as to the personal needs of the driver.

In the same context, various efforts have been made to increase driver's performance and satisfaction by employing personalized HMI technologies. Spoken dialogue systems can be used to operate devices in the automotive environment. Since drivers using these systems usually have different levels of experience, [26] has proposed a method to build a dialogue system in an automotive environment that automatically adapts to the user's experience with the system. The proposed method was implemented and the prototype was evaluated, with results showing that adaptation increases both user performance and user satisfaction.

Research activities to date have focused on providing personalized interaction mainly with in car information systems and navigation systems. In the context of information systems, the most typical example of a first generation system is COMUNICAR [1]. The main project goal was to design and develop a new concept of an integrated, in-vehicle multimedia HMI able to harmonize the messages coming from the ADAS

systems, the telematics services (telephone, route guidance, etc.), and the entertainment functions (radio, CD, etc.). Similarly, the AIDE project [13] investigated the integration of different ADAS systems and in-vehicle information systems that take into account the driver and the traffic conditions. In particular, information presented to the driver could be adapted on the basis of environmental conditions (weather and traffic), as well as on the basis of assessed workload, distraction, and physical condition of the driver. Information management must be done in a way that guarantees drivers and vehicles safety [3] and at the same time, HMI elements should be able to control and manage all the different input and output devices of the vehicle in order to provide optimum interaction.

The domain of navigation systems is also extremely important, as such systems are highly complex having countless functions and in some cases coexist with infotainment systems of a car and other components (i.e., radio, phone, CD/mp3 player, etc.). The driver navigation system demands many highly interactive activities from the driver [29]. According to [59, 65], during stressful situations, the HMI of the driver navigation system can be made adaptive to reduce the mental workload of the driver, depending on the driver's characteristics. Other concern the personalization of in-car-infotainment systems. The work presented in [23] introduces two different approaches to simplify the task of executing a preferred entertainment feature by either personalizing a list of context-dependent shortcuts or by automatically executing regularly used features. The myCOMAND case study explores the vision of an interactive user interface (UI) in the vehicle providing access to a large variety of information items aggregated from Web services [20]. It was created for gaining insights into applicability of personalization and recommendation approaches for the visual ranking and grouping of items using interactive UI layout components (e.g., carousels, lists).

2.4 Existing Knowledge models for the Automotive Industry

Ontologies, a hierarchically structured set of concepts describing a specific domain of knowledge [8], can be valuable for the automotive domain. Ontologies play a major role in supporting information exchange processes in various areas [18]. With regard to the area of automotive industry, a large amount of ontology-based knowledge models can be found in the literature mainly related to ADAS systems, autonomous vehicles, contextual awareness, adaptive HMIs and vehicle diagnostics and self-testing.

A vast variety of vehicular systems builds upon or extends ontologies relative to *ADAS systems* or *autonomous vehicle controlling*. The work presented in [42] proposes an ontology modelling approach for assisting vehicle drivers through safety warning messages during time critical situation. Tonnis et al. [63] present an ontology-based approach for deducing spatial knowledge in the context of driver-assistance systems. The authors of [45] present an ontological model of the driver as well as the vehicle. Based on these models and the information available from a specific infrastructure (i.e., cameras, sensors, etc.), the system is able to detect dangerous situations. A modular ontology supporting a car ADAS system is presented in [55], aiming at making road transport more efficient and effective, safer and more environmentally friendly.

With regard to autonomous vehicles, [4] proposes the use of a semantic control paradigm to model traffic control, vehicle path planning and steering control. Furthermore, a simple ontology that includes context concepts such as mobile entities (i.e., pedestrian and vehicle), static entities (i.e., road infrastructure and intersection), and context parameters (i.e., *isClose*, *isFollowing*, and *isToReach*) is modeled to enable the vehicle to understand the context information when it approaches road intersections [5]. Another example of autonomous vehicles is reported in [39] representing the situation at intersections for reasoning on it using traffic rules. In addition, the work presented in [43] models the traffic light control domain using a fuzzy ontology, and applies it in order to control isolated intersections. Likewise, a semantic fusion of laser point sensor data and computer vision sensing is used to support pedestrian detection as presented in [51]. Moreover, an ontology dealing with emergency situations (e.g., quitting the leftmost lane on a highway when an emergency vehicle is quickly approaching) is proposed in [7].

Additional efforts in the literature mainly focus on semantic modelling of specific essential aspects of the driving process, mainly concerning the *driver*, the *vehicle* and the *surrounding environment*. For instance, one of the goals of the PADAS project [6] was the definition of an overall methodological approach for modelling the interaction between the driver and the vehicle and its correlation with the external environment. Furthermore, research has also been conducted on the behavior of the driver (exploited in the design and safety assessment of automated systems [14]). In [22], OWL-based context model for abstract scene representation of driving scenario has been proposed which extends behavior knowledge with contextual elements of the environment such as traffic signs, the state of the driver and the vehicle itself. A more detailed representation of the driver and the environment is proposed in [11] contributing to the body of knowledge in the domain of prevention of vehicular traffic accidents.

With the main objective to facilitate commercial needs of the automotive industry, several automotive ontologies have been designed to be used in combination with the GoodRelations [31] commercial oriented vocabulary. Some concepts from these ontologies, e.g., the Volkswagen Vehicles Ontology [33] or Vehicle Sales Ontology [32], are also relevant in the context of vehicular communication including model, dimensions of the vehicle, engine, type of the vehicle (such as van, truck, etc.). Further ontology-based knowledge support is proposed also by [47] in the context of an automotive troubleshooting service system. Likewise, SAMOVAR (Systems Analysis of Modeling and Validation of Renault Automobiles) relies on ontologies aiming at preserving and exploiting previous automobile design projects [25].

Development of personalized interaction and adaptive HMI elements requires, among others, semantic knowledge regarding the user, the vehicle/environment and the current driving context [17], and can be built upon the advances presented above. The AIDE project [13] models driver-vehicle-environment aiming at the creation of adaptive HMI elements for certain assistance systems. Moreover, a modular ontology supporting an on-board vehicle multimodal interaction system is introduced by Pisanelli et al. [56]. This ontology comprises five vital domains (vehicle security, road and traffic security, meteorological, user's profiles and travel) for safer and more efficient road

transport and mobility. Finally, Feld & Müller [17] describe the “Automotive Ontology” for automotive human-machine-interaction which evolves both the concepts and the ontology design, giving a solid description of the knowledge representation aspect. Feld & Müller contribute towards a reference ontology design that highlights vital areas of the automotive application domain knowledge, as well as a collection of meta-properties related to situation-aware in-car functions and a way to model them.

3 Semantic Modelling

To efficiently and effectively organize and process the information required for personalizing the HMI elements of an ADAS system, this information is semantically modeled in the form of an ontology meta-model. Following most ontologies that incorporate aspects of the automotive domain in general, and ADAS systems in particular, semantic information is classified in three broad categories: a) the driver, b) the vehicle, and c) the environment and context of use. A high level overview of the ontology meta-model highlighting these categories is illustrated in Fig. 1, while a more elaborate discussion on the modelling of each category is provided in the following sections.

3.1 Driver and Rider

The most important requirement for providing personalized interaction in any system and context is to adopt an elaborate profile model, containing any relevant information about the user. For this purpose, several profile model standards have been proposed in recent years, including GUMO [28], FOAF [12] and SIOC [10]. Each of these models is specialized in representing different aspects of the user. For example, the FOAF (Friend Of A Friend) ontology targets the representation of user characteristics and their connections with other users. This work adopts the General User Model Ontology (GUMO), as it collects a wide range of user characteristics that are commonly modeled within user-adaptive systems, and extends it to include additional information that is relevant in an automotive context in order to enable the personalization of HMI elements in ADAS systems. Introduced extensions include driving related information (e.g., driving style and experience, risk attitude and involvement in accidents) and detailed information regarding disabilities or medical conditions that may affect driving (e.g. eye conditions). For example, the system should take into account a driver’s color blindness in order to adjust the colors of the vehicle screens to enable the driver to better distinguish vehicle notifications and possible obstacles. In addition, audio notifications may be deployed to alert the driver about road signs that could be difficult to the user to discern. Furthermore, the model includes physiological states (e.g., sleepiness, inattention, workload, etc.) and potential physiological impairments (e.g., faint, dehydration) that are of high importance in the course of driving. For instance, when a driver is identified to be sleepy, the system may start playing some energetic music to arouse the driver and choose to use louder sound notifications for informing him/her. It will also take into account the sleepiness state of the driver so that in case of an emergency it will take over control sooner than it would normally do for an alert driver. Another

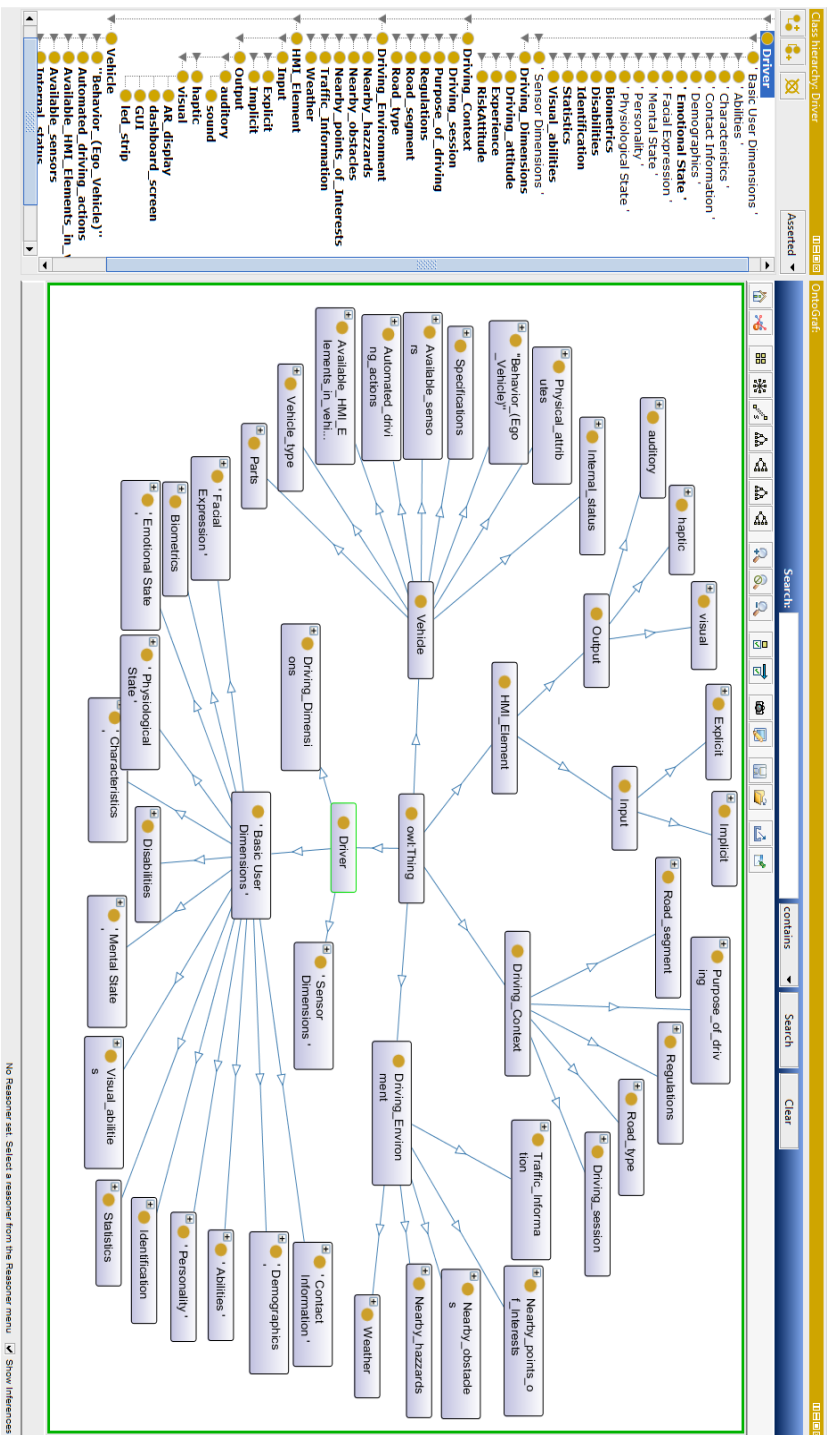


Fig. 1. A high level overview of the proposed ontology.

extension in the ontology, relevant to the context of HMI personalization, involves detailed information about user interface preferences for the interaction with the HMI elements of the vehicle. The latter includes information for both high level aspects, such as which input and output modalities are preferred by the driver, and low level aspects, such as the fonts and colors of a particular output modality. Finally, to enable the system to become more knowledgeable in the course of time, we introduce the notion of storing history and inferred values/states (will occur in the future by performing statistical analysis on pre-recorded data and applying rules on significant driver/rider states and actions). For example, for a consistently distracted driver or a driver that repeatedly ignores warning messages, the system may opt to directly take over control in case of an emergency without first notifying the driver.

The proposed ontology classifies driver/rider dimensions in two categories, static ones and dynamic ones. Static dimensions regard permanent driver/rider characteristics that are not subject to change across driving sessions, while dynamic dimensions may change both across driving sessions and in the course of a single driving session. From a data collection perspective, static dimensions typically involve information that needs to be provided as input either from the drivers/riders themselves or through some profile provider service, while information for dynamic dimensions is typically retrieved through driver monitoring using in-vehicle available sensors. Overall, the ontology models the following driver/rider dimensions:

— *Static:*

- Contact Information (e.g., name, city, emergency contact)
- Demographics (e.g., age, gender, language)
- Personality (e.g., careless, calm, neurotic, tempered)
- User Interface Preferences (e.g., fonts, colors, layout, modalities)
- General knowledge and driving experience (e.g., computer expertise, familiarity with the road)
- Driving and risk attitude (e.g., driving style, involvement in previous accidents, sensation seeking)
- Disabilities and Medical conditions (e.g., deafness, Parkinson's disease, sleep disorders)
- Visual ability (e.g., visual acuity, color blindness, contrast sensitivity)

— *Dynamic:*

- History and statistics (e.g., previous warning and user reactions, normal heart rate, history of sleepiness while driving)
- Physiological state (e.g., sleepiness, distraction, rest, stress)
- Physiological impairment (e.g., dehydration, frostbite, faint, hypothermia)
- Mental state (e.g., cognitive load)
- Emotional state (e.g., happiness, anger, road rage)
- Physiological parameters (e.g., blood pressure, current heart rate, current temperature)

3.2 Vehicle

Having information about the vehicle is also important in the context of ADAS systems in general, and for personalized interaction in particular. The proposed ontology builds on previous work on vehicle modelling [6, 11, 17, 42, 45, 55, 68, 19, 69], directly extending existing ontologies where possible and incorporating design knowledge from the ones unavailable for extension. Additionally, new parameters are introduced that are of vital importance in the personalization context, most notably the available HMI elements of the vehicle. As with the driver/rider case, all relevant vehicle parameters are also classified into static and dynamic, with static ones typically provided by the vehicle manufacturer and dynamic ones provided using vehicle systems and sensors.

The most important static parameter is the type of the vehicle, as there are different means of interaction and different potential for personalization across cars, trucks, buses and motorcycles. For example, on a motorcycle it would be ineffective to use audio notifications because of the noise from the surroundings. If such notifications would need to be used for some reason, they should be set at a very high volume level. In a vehicle with a closed-cabin, such as a car, truck or bus, such high volume levels would only be used for drivers with hearing disabilities or in cases of extreme emergencies, e.g., if the driver has fallen asleep. Other significant static vehicle characteristics include its formal specifications, as well as its structural elements, covering both interior and exterior parts and sensors. For instance, when performing an automated emergency brake, the system will take into account the braking performance of the vehicle so as to start braking earlier if necessary.

Dynamic parameters include information about the state of the vehicle (e.g., speed, location), the status of its elements (e.g., sensor values, windows being open or closed, etc.), and any relevant driving actions. For example, the speed of the vehicle may be taken into account to derive the best interaction strategy for issuing a notification message to the driver. Using visual output, e.g., displaying the message to a vehicle screen, is usually efficient, but forces the driver to take their eyes from the road in order to view the message. If the vehicle is stationary or at a low speed this may be an acceptable, however when driving at a high speed even a split second of taking the focus from the road can be fatal. Thus, visual notifications are avoided for high speeds and audio or haptic feedback is preferred instead. Another example is the consideration of the current driving action towards deciding whether or not to notify the driver about an incoming call; during manual driving or handover between manual and automated driving, the call would probably be dismissed, while during automated driving the driver would be available to take the call.

Overall, the ontology models the following vehicle aspects:

— *Static*

- Type (e.g., car, truck, bus, motorcycle)
- Specifications (e.g., max speed, horsepower, fuel consumption, braking performance)
- Interior parts (e.g., doors, windows, sunroof, pedals, gear shift, throttle)
- Exterior parts (e.g., trunk, lights, side mirrors)

- Physical attributes (e.g., dimensions, weight)
- Available sensors (e.g. GPS, camera, lidar, level of light, temperature)
- Available HMI elements (e.g., speakers, screens, microphones)

— *Dynamic*

- Vehicle behavior/Ego-vehicle (e.g., speed, acceleration, location, orientation)
- Internal status (e.g., window status, light conditions, sound level)
- Automated driving actions (e.g., turning, following, taking automatic control)

Semantic information about the available HMI elements is particularly important in the context of personalization, as such elements provide the means for receiving input from and giving output to the driver. However, previous vehicle ontologies for ADAS systems lack such information. In this work, particular attention is paid to the available HMI elements of the vehicle, modeling all possible element categories and interaction methods. Effectively, this is a sub-ontology of vehicle rather than a separate ontology, but is critical to the purposes of this work so it is presented here separately. The following aspects of HMI elements are modeled:

- Physical characteristics (e.g., dimensions, location in the vehicle, mounted/free)
- Interaction types (e.g., input / output, implicit / explicit)
- Modality types (e.g., visual, auditory, haptic)
- Input (e.g., touch, speech, gestures, hardware buttons, dashboard controls)
- Output (e.g., dashboard screen, AR display, sound, vibration)
- Explicit interaction (e.g., GUI, voice commands, dashboard notifications)
- Implicit interaction (e.g., gestures, poses, visual attractors, ambient light, force feedback, wearable systems)

For example, let's revisit the example of determining the best interaction strategy for issuing a notification message to the driver. Such a decision would take into account the available output modalities and whether it would be more appropriate to use a visual, auditory or haptic output modality based on the current situation. If for instance only visual modalities are present, say a dashboard screen, the center display, and a led strip, then the system would select based on their location, so as to allow the driver to view the notification with minimal visual distraction. If on the other hand the notification involves a message with content difficult to visualize on a led strip or a small dashboard screen, then the selection would also take into account the screen size and possibly opt for the center display. In both cases, if audio feedback was available, it would probably be preferred as a less distracting way of reaching out to the driver.

3.3 Environment and Context of Driving

Besides information about the driver/rider and the vehicle, personalization logic should also take into account data about the surrounding environment, as well as the overall context of driving. Prominent such parameters for personalization include the weather and traffic conditions. For example, if glare from the sun or the headlights of other

vehicles compromise the driver's vision, an auditory message should be preferred over a visual one. As another example, when trying to decide about notifying the driver about an incoming call, the system would also have to take into account traffic information and any nearby obstacles. For instance, when maneuvering between fallen rocks it wouldn't be a good time to answer a phone call. Other important parameters include the particular driving session, with information ranging from the starting point and destination to the purpose of the drive, the chosen route and the points of interest along the road. For example, in a routine drive from home to work, the system may turn on the news, while in a leisure drive it may put on some relaxing music. With respect to data collection, most of the environment and context information is derived from external resources such as weather, traffic information and navigation services.

Various existing ontologies model environment and context information [5, 17, 22, 45-51], however none of them seem to provide a holistic approach towards modeling environment and context driving aspects. The proposed ontology draws from existing models and aggregates all relevant environment and context elements that can be useful for personalizing HMI elements. In particular, the ontology includes the following aspects:

— *Driving Environment*

- Weather (e.g., light conditions, sun glare, fog, rain, snow, hail, wind)
- Traffic information (e.g., flow, accidents, diversions, closed roads)
- Nearby obstacles (e.g., other vehicles, pedestrians, fallen rocks)
- Nearby hazards (e.g., potholes, speed bumps, spilt oil, ice)
- Nearby points of interest (e.g., restaurants, gas stations)

— *Driving Context*

- Regulations (e.g., traffic lights, traffic signs, speed limits, priorities)
- Road type (e.g., highway, private road, national road)
- Road segment information (e.g., roundabout, intersection, number of lanes, bus lane, pedestrian crossing)
- Driving session (e.g., start point, destination point, route)
- Purpose of driving (e.g., routine, profession, emergency, leisure)

4 Semantic Reasoning for Personalized HMIs

4.1 Employing reasoning into vehicle's HMIs

In general, reasoning means deriving facts that are not expressed in ontology or in knowledge base explicitly. Furthermore, reasoning describes the task of answering complex questions using the facts stored in a knowledge base, and possibly using a mechanism that describes how further facts can be automatically derived. For the purposes of this research work, the selection of the appropriate reasoning engine is considered to be of the utmost importance. Using rules and facts regarding the *driver*, the

vehicle and the *surrounding environment*, an inference engine is able to deduce conclusions and therefore to produce the appropriate personalization behavior from an HMI and automation preferences perspective.

According to the literature, an inference engine adopts two strategies of execution: a) *forward chaining* and b) *backward chaining*. Also, there are engines that implement both, called *hybrid chaining engines*. Forward chaining starts with the available data and uses inference rules to extract more data until a goal is reached. Backward chaining starts with goals, and works backward to determine what facts must be asserted so that the goals can be achieved [27]. In the present work, forward chaining, as a “data-driven” and thus reactionary reasoning strategy, seems like a prerequisite for deducing into new conclusions using data stemming from in-vehicle available sensors and facts stored in the knowledge base. In addition, one of the advantages of forward-chaining over backward-chaining is that the reception of new data can trigger new inferences, which makes the engine better suited to dynamic situations in which conditions are likely to change [41].

4.2 Advantages of a Rule Engine based approach

Delivering personalization HMI decisions based on rules can be facilitated by a generic rule reasoner approach. A generic rule reasoner is a rule based reasoner that supports user defined rules. Usually, a rule engine decides which rules to apply, and computes the result of their application, that may deduce new knowledge, or an action to perform. In particular, a rule engine includes the following components: a) a *rule base*, containing user defined rules, b) a *knowledge base* that contains known facts and c) an *inference engine* for processing rules. Rules operate on facts of the knowledge base. Facts may change over time with new facts being added and old facts being removed. Rules are based on conditions which are evaluated against facts.

Rules are usually specified in a Rule Language such as RuleML, OCL, SWRL, etc. [66] which captures the rules and facts in a human readable form. Each rule engine technology supports one or more rule languages, thus offering many advantages over hand coded “if...then” approaches. To this end, rules are *easier to understand* than procedural code, so they can be effectively used to bridge the gap between domain knowledge experts (who mainly are non-technical) and developers [61]. The key advantage is the capability of *declarative programming*, making it easier for domain experts to express the logic of a computation in an abstract way, without having to describe its control flow. Among the main benefits of a rule engine is the fundamentally breaking of data and logic (*logic and data separation*). Logic, laid in rules, can be much easier to maintain and modify. Also, keeping rules in a separate repository facilitates the *centralization of knowledge* which allows seamless adaptation to decisions when they are changing and enables *greater flexibility* and *reusability*.

4.3 Comparison of applicable Rule Engines

Performance issues, rule language expressiveness, community support, software license, platform compatibility are some of the arguments taken into account for comparing existing rule engines. Most rule engines employ the Rete algorithm [38] which is still the leading algorithm for general-purpose Rule Engines. However, the Rete algorithm sacrifices memory for speed. However, since speed is of the utmost importance in automotive applications, Rule Engines which implement this algorithm, are preferred. In [37] more than 30 readily available Rule Engines and reasoners implemented in Java are listed. The most prominent systems are JBoss Drools [36] which is a free, Open Source, forward chaining inference rule engine based on an enhanced implementation of Charles Forgy's Rete algorithm [21]. Pellet 2 is an OWL-DL Java-based reasoner which provides standard and advanced reasoning services for OWL ontologies. Jena and JenaBean are an open source Java-based framework for "semantic web" application [15]. FuzzyDL System is a description logic reasoner that supports both Fuzzy Logic and fuzzy Rough Set reasoning [9].

The number of .NET compatible Rule Engines is quite limited. To begin with, the Drools.NET is a .NET port for Drools that enables .NET developers to exploit the powerful that Drools Rule Engine provides through a completely managed .NET code base. However, the Drools.NET is still in Beta version and only available for outdated .NETv1_1, v2. Another .NET approach is the SRE (Simple Rule Engine) [35] which is a lightweight forward chaining inference rule engine for .NET. It allows developers to combine rule-based and object oriented programming methods to add rules written in XML to new and existing applications. Windows Workflow Foundation (WF) ships with a robust business Rule Engine that can be incorporated into workflows to assist in managing business processes. The Rule Engine can be used outside of workflows in any .NET application to provide robust rule based capabilities. These capabilities range from simple conditions that drive activity execution behavior to complex rulesets executed by a full-featured forward-chaining Rule Engine.

The majority of Rule Engines have their own unique "native language" and given the complexity of the automotive domain, it normally takes a considerable amount of time for domain knowledge experts or developers to learn the language. For the purposes of the present work, the creation of domain specific rule scripting language may be an alternative approach worth considering. Domain-specific languages allow to specify and express domain objects and idioms as part of a higher-level language for programming. By providing a higher level of abstraction, domain-specific languages allow to focus on the application or domain while concealing details of the programming language or platform. In the context of delivering personalized interaction with HMI elements in automotive applications, the main purpose of such a rule scripting language should be to offer higher expressiveness and manageable complexity at the same time. ACTA is an indicative example of domain specific rule based language aiming at facilitating the activity analysis process during smart game design by early intervention professionals who are not familiar with traditional programming languages [70]. Developers can use ACTA also for applications whose behavior is composed of a

finite number of states, transitions between those states and actions, as well as for application based on rules driven workflows. ACTA's runtime is based on the WF.

For the purposes of the present work, further investigation of the aforementioned Rule Engines will be conducted in order to select the most appropriate one in terms of performance, efficiency, expressiveness, etc.

5 Conclusions and Future Work

ADAS systems promise to deliver capabilities and features needed to simplify the driving process and reduce vehicular accidents. Thanks to rapid advances in vision, sensors, connectivity, infrastructure and HMI technologies, automotive engineers will continue to find cost-effective solutions for realizing ADAS designs. The next big step is towards offering adaptive and personalized ADAS systems, where automated functions and interaction between the driver/rider and the vehicle takes into account the driver's/rider's state as well as the current situational and environmental context. In this context, the work presented here adopts an ontology-based modelling approach for semantically representing all relevant information, and uses it to personalize the HMI elements of an ADAS system.

Central to our proposition is a comprehensive ontology that models all relevant driver and rider, vehicle, and surrounding environment data. Driver and rider modelling takes into account both static information such as characteristics, personality, preferences, driving experience and relevant medical disabilities or medical conditions, and also attributes that change dynamically during a driving session such as mental, physiological and emotional state. Semantic modelling regarding vehicle data also considers both static and dynamic aspects, and covers all attributes from vehicle type, specifications and structural elements, to the current vehicle state and element status while driving. In the context of personalization, of particular interest are also the available HMI elements of the vehicle, which are modelled based on their physical characteristics, interaction types and modality types. Finally, the environment and context of driving are thoroughly modelled, including information about traffic regulations, weather and traffic conditions, nearby obstacles and points of interest, as well as information about the particular driving session.

With the ontology including all relevant semantic information, supporting personalized interaction requires transforming the abstract knowledge provided by automotive HMI domain experts into concrete rules that can be deployed for reasoning on ontology model instances. A rule-based reasoning engine will be used to infer conclusions (new knowledge) and therefore to produce the appropriate adaptation decisions. Thereafter, the decisions can be used to deliver user interaction through the vehicle's HMI elements that best fit the particular driver or rider, surrounding environment, and the overall driving context.

Currently, the proposed ontology has been implemented in OWL (Ontology Web Language) [52] using the ontology editor Protégé [50], and the focus is now on selecting the most appropriate approach for expressing the logic rules and performing reasoning. To this end, it is planned to explore available alternatives, such as using SWRL

rules and the Drools reasoner, or adopting the Microsoft Workflow Foundation rules in combination with code actions, and evaluate them in terms of expressiveness, effectiveness and efficiency. Future work also includes the design and implementation of an HMI personalization framework that will act as the middleware between the reasoning system and the eventual HMI. This framework will initiate the reasoning process based on input from the driver and the various sensors, and will use the decision making results to present the output to the driver. In particular, it will handle aspects regarding how the user interface will appear by activating and deactivating adaptive GUI components, as well as maintaining binding to all available HMI elements and invoking them as needed. It will also dictate how high-level reasoning results such as ‘use audio notifications exclusively’, ‘utilize haptic feedback’, and ‘simplify the user interface’, are manifested for each particular HMI element in isolation, and then orchestrated to provide a personalized user experience.

Acknowledgments. This work has been conducted in the context of the Project ICT-ADAS&ME “Adaptive ADAS to support incapacitated drivers & Mitigate Effectively risks through tailor made HMI under automation”, funded by the European Commission under the Horizon 2020 Framework Programme (Grant Agreement 688900). The sole responsibility for the content of this paper lies with the authors. It does not necessarily reflect the opinion of the European Union. Neither the project nor the European Commission are responsible for any use that may be made of the information contained therein.



6 References

1. Amditis, A., Bekiaris, E., Montanari, R., Baligand, B., Perisse, J., Belotti, F., & Kuhn, F. (2001). An innovative in-vehicle multimedia HMI based on an intelligent information manager approach: the Comunicar design process. In 8th World Congress on Intelligent Transport Systems.
2. Amditis, A., Pagle, K., Joshi, S., Bekiaris, E. (2010) Driver–Vehicle–Environment monitoring for on-board driver support systems: Lessons learned from design and implementation, *Applied Ergonomics*, Volume 41, Issue 2, pp. 225-235.
3. Amditis, A., Polychronopoulos, A., Andreone, L., & Bekiaris, E. (2006). Communication and interaction strategies in automotive adaptive interfaces. *Cognition, Technology & Work*, 8(3), 193-199.
4. Amin, S. M., Rodin, E. Y., Liu, A. P., Rink, K., Cusick, T. W., Ghosh, B. K., ... & Wootton, J. R. (1995, September). A semantic control approach to intelligent transportation systems. In *Intelligent Vehicles' 95 Symposium., Proceedings of the* (pp. 430-435). IEEE.
5. Armand, A., Filliat, D., & Ibanez-Guzman, J. (2014, June). Ontology-based context awareness for driving assistance systems. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE* (pp. 227-233). IEEE.

6. Bellet, T., Mayenobe, P., Baumann, M., Alonso, M., Vega, M. H., Martín, Ó., et.al.. (2010). Integrated Human Modelling and Simulation to support Human Error Risk Analysis of Partially Autonomous Driver Assistance Systems.
7. Bermejo, A. J., Villadangos, J., Astrain, J. J., & Cordoba, A. (2013). Ontology based road traffic management. In *Intelligent Distributed Computing VI* (pp. 103-108). Springer Berlin Heidelberg.
8. Blomqvist, E., Ohgren, A., 2008. Constructing an enterprise ontology for an automotive supplier. *Eng. Appl. Artif. Intell.* 21 (3), 386–397.
9. Bobillo, F., & Straccia, U. (2008, June). fuzzyDL: An expressive fuzzy description logic reasoner. In *Fuzzy Systems, 2008. FUZZ-IEEE 2008.*(IEEE World Congress on Computational Intelligence). IEEE International Conference on (pp. 923-930). IEEE.
10. Bojars, U., Breslin, J.G.: SIOC Core Ontology Specification. Namespace document, DERI, NUI Galway, January 2009. <http://rdfs.org/sioc/spec/>
11. Boudra, M., Hina, M. D., Ramdane-Cherif, A., & Tadj, C. (2015). Architecture and ontological modelling for assisted driving and interaction. *International Journal of Advanced Computer Research*, 5(20), 270.
12. Brickley, D., Miller, L.: FOAF Vocabulary Specification 0.91. Namespace document, FOAF Project, November 2007. <http://xmlns.com/foaf/0.1/>
13. Brouwer, R. F., Hoedemaeker, M., & Neerincx, M. A. (2009, July). Adaptive interfaces in driving. In *International Conference on Foundations of Augmented Cognition* (pp. 13-19). Springer Berlin Heidelberg.
14. Cacciabue, P. C., & Carsten, O. (2010). A simple model of driver behaviour to sustain design and safety assessment of automated systems in automotive environments. *Applied ergonomics*, 41(2), 187-197.
15. Carroll, J. J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A., & Wilkinson, K. (2004, May). Jena: implementing the semantic web recommendations. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters* (pp. 74-83). ACM.
16. Craig, J. (2012) Map Data for ADAS. *Handbook of Intelligent Vehicles*, pp. 881-892, Springer London.
17. Feld, M., & Müller, C. (2011, November). The automotive ontology: managing knowledge inside the vehicle and sharing it between cars. In *Proceedings of the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications* (pp. 79-86). ACM.
18. Fensel, D., Van Harmelen, F., Horrocks, I., McGuinness, D. L., & Patel-Schneider, P. F. (2001). OIL: An ontology infrastructure for the semantic web. *IEEE intelligent systems*, 16(2), 38-45.
19. Fernandez, S., Hadfi, R., Ito, T., Marsa-Maestre, I., & Velasco, J. R. (2016). Ontology-Based Architecture for Intelligent Transportation Systems Using a Traffic Sensor Network. *Sensors*, 16(8), 1287.
20. Fischer, P., & Nürnberger, A. (2010, June). myCOMAND automotive user interface: Personalized interaction with multimedia content based on fuzzy preference modeling. In *International Conference on User Modeling, Adaptation, and Personalization* (pp. 315-326). Springer Berlin Heidelberg.
21. Forgy, C. L. (1982). Rete: A fast algorithm for the many pattern/many object pattern match problem. *Artificial intelligence*, 19(1), 17-37.
22. Fuchs, S., Rass, S., Lamprecht, B., & Kyamakya, K. (2008, February). A model for ontology-based scene description for context-aware driver assistance systems. In *Proceedings of*

- the 1st international conference on Ambient media and systems (p. 5). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
23. Garzon, S. R. (2012, June). Intelligent in-car-infotainment systems: A contextual personalized approach. In *Intelligent Environments (IE)*, 2012 8th International Conference on (pp. 315-318). IEEE.
 24. Garzon, S. R., & Poguntke, M. (2011, July). The personal adaptive in-car HMI: integration of external applications for personalized use. In *International Conference on User Modeling, Adaptation, and Personalization* (pp. 35-46). Springer Berlin Heidelberg.
 25. Golebiowska, J., Dieng-Kuntz, R., Corby, O., & Mousseau, D. (2001, October). Building and exploiting ontologies for an automobile project memory. In *Proceedings of the 1st international conference on Knowledge capture* (pp. 52-59). ACM.
 26. Hassel, L., & Hagen, E. (2006). Adaptation of an automotive dialogue system to users' expertise and evaluation of the system. *Language resources and evaluation*, 40(1), 67-85.
 27. Hayes-Roth, F., Waterman, D. A., & Lenat, D. B. (1983). An overview of expert systems. *Building expert systems*, 1, 3-29.
 28. Heckmann, D., Schwartz, T., Brandherm, B., Schmitz, M., & von Wilamowitz-Moellendorff, M. (2005, July). Gumo—the general user model ontology. In *International Conference on User Modeling* (pp. 428-432). Springer Berlin Heidelberg.
 29. Heimgärtner, R. (2007). Towards Cultural Adaptability in Driver Information and -Assistance Systems. In *Usability and Internationalization. Global and Local User Interfaces: Second International Conference on Usability and Internationalization, UI-HCII 2007, Held as Part of HCI International 2007, Beijing, China, July 22-27, 2007, Proceedings, Part II*, pp. 372-381, Springer Berlin Heidelberg.
 30. Héléne, T. V., Thierry, B., Serge, B., Matti, K., Jouko, V., Evangelos, B., Panou, M., Engström, J., & Anders, A. (2005). Development of a driver situation assessment module in the AIDE project. *IFAC Proceedings Volumes*, 38(1), 97-102.
 31. Hepp, M. (2008, September). Goodrelations: An ontology for describing products and services offers on the web. In *International Conference on Knowledge Engineering and Knowledge Management* (pp. 329-346). Springer Berlin Heidelberg.
 32. Hepp, M. (2010). Vehicle Sales Ontology. Available online: <http://www.heppnetz.de/ontologies/vso/ns>
 33. Hepp, M. (2010). Volkswagen Vehicles Ontology. Available online: <http://www.volkswagen.co.uk/vocabularies/vvo/ns>
 34. Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosz, B., & Dean, M. (2004). SWRL: A semantic web rule language combining OWL and RuleML. *W3C Member submission*, 21, 79.
 35. <http://simpleruleengine.tripod.com/>
 36. <http://www.drools.org/>
 37. http://www.manageability.org/blog/stuff/rule_engines/view
 38. https://en.wikipedia.org/wiki/Rete_algorithm
 39. Hülsen, M., Zöllner, J. M., & Weiss, C. (2011, June). Traffic intersection situation description ontology for advanced driver assistance. In *Intelligent Vehicles Symposium (IV)*, 2011 IEEE (pp. 993-999). IEEE.
 40. Hummel, T., Kühn, M., Bende, J., & Lang, A. (2011). Advanced driver assistance systems. German Insurance Association Insurers Accident Research. Available on www.udv.de, accessed at, 6(01), 2015.
 41. Kaczor, K., Bobek, S., & Nalepa, G. J. (2010). Overview of Expert System Shells. Kaczor K., Szymon B., Grzegorz J.- Krakow, Poland: Institute of Automatics: AGH University of Science and Technology, Poland, 5.

42. Kannan, S., Thangavelu, A., & Kalivaradhan, R. (2010). An intelligent Driver Assistance System (I-DAS) for vehicle safety modelling using ontology approach. *International Journal of UbiComp*, 1(3), 15-29.
43. Keyarsalan, M., & Montazer, G. A. (2011). Designing an intelligent ontological system for traffic light control in isolated intersections. *Engineering Applications of Artificial Intelligence*, 24(8), 1328-1339. Chicago
44. Kohfeld, D. L. (1971). Simple reaction time as a function of stimulus intensity in decibels of light and sound. *Journal of experimental psychology*, 88(2), 251.
45. Lashkov, I., Smirnov, A., Kashevnik, A., & Parfenov, V. (2015, September). Ontology-based approach and implementation of ADAS system for mobile device use while driving. In *International Conference on Knowledge Engineering and the Semantic Web* (pp. 117-131). Springer International Publishing.
46. Lee, J. D., Hoffman, J. D., & Hayes, E. (2004, April). Collision warning design to mitigate driver distraction. In *Proceedings of the SIGCHI Conference on Human factors in Computing Systems* (pp. 65-72). ACM.
47. Liang, J. S. (2014). The service task implementation in automotive trouble-shooting using an ontology-based knowledge support system. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 228(13), 1599-1621.
48. Lu, M., Wevers, K., & Van Der Heijden, R. (2005). Technical feasibility of advanced driver assistance systems (ADAS) for road traffic safety. *Transportation Planning and Technology*, 28(3), 167-187.
49. Mueller, M. (2014). Deficiency drive. *Vision Zero International*.
50. Musen, M.A. The Protégé project: A look back and a look forward. *AI Matters*. Association of Computing Machinery Specific Interest Group in Artificial Intelligence, 1(4), June 2015. DOI: 10.1145/2557001.25757003.
51. Oliveira, L., Nunes, U., Peixoto, P., Silva, M., & Moita, F. (2010). Semantic fusion of laser and vision in pedestrian detection. *Pattern Recognition*, 43(10), 3648-3659.
52. OWL Web Ontology Language Reference. Available online: <http://www.w3.org/TR/2004/REC-owl-ref-20040210/>
53. Piao, J., & McDonald, M. (2008). Advanced driver assistance systems from autonomous to cooperative approach. *Transport Reviews*, 28(5), 659-684.
54. Pisanelli, D. M., De Lazzari, C., Bugli-Innocenti, E., & Zanetti, N. (2008). An Ontology Supporting a Car Advanced Driver Assistance System. In *SoMeT* (pp. 403-412).
55. Pisanelli, D. M., De Lazzari, C., Bugli-Innocenti, E., & Zanetti, N. (2008). An Ontology Supporting a Car Advanced Driver Assistance System. In *SoMeT* (pp. 403-412).
56. Pisanelli, D. M., De Lazzari, C., Innocenti, E. B., & Zanetti, N. (2009). An ontology supporting an on-board vehicle multimodal interaction system. In *Multimodal Human Computer Interaction and Pervasive Services* (pp. 230-242). IGI Global.
57. Pollard, E., Morignot, P., & Nashashibi, F. (2013, July). An ontology-based model to determine the automation level of an automated vehicle for co-driving. In *Information Fusion (FUSION), 2013 16th International Conference on* (pp. 596-603). IEEE.
58. Prud, E., & Seaborne, A. (2008). SPARQL query language for RDF. Available online: <https://www.w3.org/TR/rdf-sparql-query/>
59. Recarte, M. A., & Nunes, L. M. (2003). Mental workload while driving: effects on visual search, discrimination, and decision making. *Journal of experimental psychology: Applied*, 9(2), 119.
60. Rockwell, T. H. (1971). Visual acquisition of information in driving through eye-movement techniques: an overview. Highway Safety Research Center. University of North Carolina. Section II, 64-95.

61. Staab, S., Horrocks, I., Angele, J., Decker, S., Kifer, M., Grosz, B., & Wagner, G. (2003). Where are the rules?. *IEEE Intelligent Systems*, 18, 76-83.
62. Tigadi, A., Gujanatti, R., Gonchi, A., (2016). ADVANCED DRIVER ASSISTANCE SYSTEMS, *International Journal of Engineering Research and General Science* Volume 4, Issue 3, May-June, 2016 ISSN 2091-2730.
63. Tonnis, M., Klinker, G., & Fischer, J. G. (2007, March). Ontology-based pervasive spatial knowledge for car driver assistance. In *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops' 07. Fifth Annual IEEE International Conference on* (pp. 401-406). IEEE.
64. Tonnis, M., Lange, C., & Klinker, G. (2007, November). Visual longitudinal and lateral driving assistance in the head-up display of cars. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on* (pp. 91-94). IEEE.
65. Verwey, W. B. (2000). On-line driver workload estimation. Effects of road situation and age on secondary task measures. *Ergonomics*, 43(2), 187-209.
66. Wagner, G., Giurca, A., & Lukichev, S. (2006). A usable interchange format for rich syntax rules integrating OCL, RuleML and SWRL. *Proc. of WSh. Reasoning on the Web*.
67. World Wide Web Consortium. (2014). RDF 1.1 concepts and abstract syntax. Available online: <https://www.w3.org/TR/rdf11-concepts/>
68. Zhao, L., Ichise, R., Mita, S., & Sasaki, Y. (2014, November). An ontology-based intelligent speed adaptation system for autonomous cars. In *Joint International Semantic Technology Conference* (pp. 397-413). Springer International Publishing.
69. Zhao, L., Ichise, R., Mita, S., & Sasaki, Y. (2015). Core Ontologies for Safe Autonomous Driving. In *International Semantic Web Conference (Posters & Demos)*.
70. Zidianakis, E. (2015). Supporting young children in ambient intelligence environments (Doctoral dissertation, University of Crete, Computer Science Department).