Towards Advanced Information Fusion for Driver Assistant Systems of Modern Vehicles

Florian Dittmann and Konstantina Geramani  
Innovation Management  
TWT Gmbh, Science & Innovation  
Stuttgart, Germany  
florian.dittmann|konstantina.geramani@twt-gmbh.de

George Rigas, Christos Katsis, Dimitrios I. Fotiadis  
Department of Computer Science,  
University of Ioannina  
Ioannina, Greece  
rigas@cs.uoi.gr| ckatsis@cc.uoi.gr| fotiadis@cs.uoi.gr

Abstract—The ever increasing number of external and internal data available in modern cars can only become beneficial if the data are assessed correctly and presented according to the drivers’ needs. This paper presents a well-designed method based on advanced information fusion that assembles and processes data centralized, which has been developed within the framework of the European funded research project I-WAY. A key task of this method is the reduction of the input data to a manageable amount that supports the unobtrusive delivery of useful and safety-critical information to drivers. The solution proposed introduces the transformation of mainly cyclic arriving data from the time to the event domain. The main advantage is that the communication load is reduced to the minimum, as only relevant changes—events—must be processed in subsequent modules. The pre-selection of events thereby serves as data filter and data fusion, allowing for fast decision making. The use case at the end of the paper presents preliminary results on the application of this data fusion method gathered in a vehicle from video, radar, and weather information.

Decision Support Systems, Information Fusion, Time and Event Domain, Driver Assistance Systems, Interworking with Sensor Network Technologies

I. INTRODUCTION

The trend in the automotive domain is that vehicles function no more as stand-alone units, whose drivers are forced to drive based on their own perception, but they can make dispose of multiple data gathered from their surroundings, arriving to them from other cars, the road infrastructure, or on-board sensing units, all extending the perception of drivers. While the availability and quality of the data constantly increases, the requirements to achieve a beneficial presentation of these data to drivers often are not sufficiently considered. Information fusion from multiple sources thereby offers benefits of delivering complex information to drivers pre-processed and adapted to the needs of each case. More recently, there has been a push to extend systems to use not only multiple sensors, but multiple sensing modalities. Multi-sensor fusion using complementary sensing modalities greatly increases the robustness of any sensing system [4].

Information fusion in the automotive domain can be found in several works, for example [2], [3]. Many of these works also focus on advanced driver assistance systems, which already can be found partly in modern cars and are thought to be used also in future vehicles. Few systems however focus on the overall integration of data, and are applicable vendor independent.

Information fusion fundamentally discusses the combination of data within multisensor systems. Basically, two or more sensors $S_i$ shall be utilized appropriately to achieve a performance $L$ that outperforms the simple addition of their single performances $L_i$, thus $L(\bigcup(L_i)) > L(\sum L_i)$ [1]. An excellent overview on techniques and methods of information fusion, also focusing on wireless sensor networks, is given in [10]. Information fusion for multimedia data analysis, as targeted in this work, is discussed [11].

Fundamental consideration on time vs. event triggered processing is extensively covered in [7]. Targeting the domain of safety-critical applications, [5] discusses requirements concerning the transformation from the time to the event domain. Reference [12] gives an overview on the same issue for the automotive domain from the control theory point of view. Moreover, [6] discusses whether to use time- or event triggered communication as a non-functional feature that requires early consideration.

The I-WAY project deals with these issues and investigates how to integrate multiple sources and modalities of environmental data of cars and how to inform drivers in case of critical situations, primarily focusing on highways. A central control unit, called the Situation Assessment Module (SAM), thereby accumulates data and issues alert information based on up-to-date decision making. In its fundamental version, the project exploits infrastructural information offered wireless by the road (weather, construction areas, congestions, etc.), near-range information sent by other cars (obstacles behind curves, slow-driving vehicles, etc.), and local information gathered by on-board sensing systems such as video cameras and radar units (obstacles ahead, lane deviation, etc.). As already this amount of data quickly can overload the SAM, a preprocessing module is integrated into the system with the aim to fuse incoming information and forward tailored messages.

In this work, the methods and techniques of such a preprocessing module are discussed, and presented in such a way as to reduce continuously arriving data to meaningful event messages. The core method applied thereby is information fusion based on the transformation from the time to the event domain. As auxiliary means, the module accumulates information in an internal data-structure. To also
meet real-time requirements, the overall design follows a simple structure, which helps to reduce costs, provide well-defined interfaces, and become an open and vendor-independent standard. Moreover, extensions are foreseen, such as the integration of a plausibility analysis of incoming data to increase the reliability of the system.

II. OVERVIEW

Based on a brief discussion of the I-WAY system (Fig.1), including infrastructure to vehicle and vehicle to vehicle communication, we define the requirements of a preprocessing module for information fusion.

![Figure 1. Overview of the system](Image)

A. General

I-WAY is a novel cooperative driving platform that significantly enhances drivers' perception on road environment by producing high-level and useful information for the driver through the analysis and fusion of real-time data obtained from: (i) an in-vehicle sensing system, (ii) the road infrastructure, and (iii) neighboring cars. The core of this system is the I-WAY DSS, a Decision Support System that encapsulates all the intelligence required from an adaptive driver assistance system. The key features of I-WAY DSS are: (i) the proactive risk assessment, meaning the preview of an event/situation and provision of notification messages to the driver about potential risks and (ii) the scouting function, meaning the detection of events in the local environment, to send updated information to the road infrastructure and other cars. I-WAY DSS receives information about i) the surroundings of the vehicle through continuous monitoring of the road scenery with on-board sensors, ii) driver status from a dedicated driver monitoring module, iii) vehicle position, speed, and heading via GPS and CAN bus, iv) weather and traffic conditions or other incidents on the highway like road constructions, blocks etc. via road infrastructure, and v) local environment and behavior of other vehicles. The information is fused so that an enhanced situation assessment is performed and an optimal warning strategy for road hazards can be adopted. As optimal warning strategy is considered the delivery of tailored messages to the driver in terms of i) message content (information about the most critical event), ii) message type (notification, warning, alert), iii) message format (text, vocal), and iv) delivery time (considering also repeating messages when necessary). The one part of the I-WAY DSS dealing with the optimal driver assistance is a framework based on Dynamic Influence Diagrams, as described in [9]. The other part of the I-WAY DSS, called Alert Manager was developed in order to efficiently handle the incoming data, manage the event priorities and determine the appropriate message content for driver notification, see [8].

B. V2V and V2I Communication

This is the network layer where the communication between vehicles (V2V) and infrastructure and vehicles (I2V) are performed. Each vehicle produces a number of messages related to potential hazards either detected by the ego-vehicle or received as messages by other I-WAY vehicles or the Road Infrastructure through the Data Vehicle Router. Those messages constitute the external information sources that need to be filtered and merged during the data fusion process.

C. Preprocessing Module

The basic intention of the preprocessing module is to provide high-level output for the DSS, particularly fusing input information of different sources over time. The DSS then decides on the actions to be performed, e.g., generating warning messages to the driver. In order to allow for a fast and profound assessment of the situation, only meaningful data should be forwarded to the central intelligent system. The paradigm of the preprocessing module therefore is to apply advanced information fusion based on a transformation from time-triggered to event-triggered—a transformation from the spatial domain to the event domain. An abstract formulation of this paradigm comprises the following:

\[ XYZ \] and \[ dX/dt, dY/dt, \text{ and } dZ/dt \], \text{ etc.} \rightarrow \text{event,} \]

where \( XYZ \) denotes the location of critical objects and situations referring to the vehicles environment, \( dX/dt \) denotes the temporal changes of these coordinates, \( c \) other conditions.

On top of this fundamental behavior, several other features add to the overall performance of the preprocessing module: First of all, a tracking of raw monitoring information over time helps to increase the level of confidence and to detect wrong messages concerning the surrounding of a car. By virtue of this tracking mechanism, memory is introduced to the system and a more reliable decision becomes possible. Moreover, multiple sources basically can increase the reliability and confidence level of data. The intention of the module is to implement early sensor fusion in the sense that environmental information gained by multiple acquisition modules is analyzed and combined to generate high-level data, potentially having a higher reliability. However, the intention to combine data from different monitoring sources requires the negotiation to a canonical reference base; otherwise the quality of the data could even be degraded in the worst case.

Taking all these challenges and requirements into account, the preprocessing module thus targets on the best solution to extract data and forward relevant information to the central intelligent system—the DSS in the I-WAY case.
III. MODULAR DECOMPOSITION

For maintenance and complexity reasons, the preprocessing module was decided to be split into discrete sub-modules, cf. Fig. 2. By the partitioning into sub-modules, we can seamlessly integrate the concept of transformation from the time to the event domain and thus achieve the goal of reducing the amount of input data to meaningful messages.

A. Receive and Decode Data

In this sub-module the data is accepted and decoded. The benefit of having a specific module for this task manifests in the possibility to define a clear interface for arriving data. Future extensions in terms of additional sources can be seamlessly integrated. As the sub-module denotes the beginning for the sequence of sub-modules, the question of activation arises. Basically, the module can be triggered periodically by newly arriving data or operate completely time-triggered performing a cyclic behavior. Both versions serve the requirements of finally generating event-triggered messages.

B. Sanity Check/Plausibility Analysis

As test for data corruption and to remove redundant information, a sanity check/plausibility analysis is implemented. Based on elementary methods, we assess arriving data. The data is checked whether it fulfils the given range description, also including possible error messages (knowledge-based redundancy).

To avoid a possible roll-back, the plausibility analysis is not performed in parallel to the other sub-modules. As the plausibility tests are basic tests, they are not meant to consume a substantial amount of computing performance. Moreover, the input data often already is attached by a level of confidence; therefore a fundamental analysis of the plausibility of the monitored environment is given and can be used directly.

C. Update the Internal Structure

An internal structure (more details, see subsequent section) is used to track the environment. It can be seen as internal database. The structure not only holds the results of the information fusion, it also becomes an input for the information fusion itself—new states are calculated by referring to previous situations, thus adding memory to the system. Thus, a sensor fusion is done in this step. For example, notifications about weather changes are only accepted if multiple sources can verify the messages. False information thereby is filtered before event messages are generated.

The update is done by referring to multiple sources, exploiting the redundancy of them. Information of different systems that monitor the same area are checked versus each other, e.g. an object directly ahead must be identified by local as well as external sources of information. Moreover, a number of sequentially arriving snapshots combined leads to a reliable interpretation of the situation, including the assessment of the risks. Furthermore, sources that dispatch update information event-triggered can be included. Their messages, which otherwise could be isolated, are fused with other data and thus they get more relevance to the overall decision process.

We thus perform an information fusion based on newly arriving data and old data, originating from multiple sensors. The structure also allows storing so far not important information, which is kept for a specific time, depending on the memory depth of the internal structure.

D. Event Message Generation

For the event message generation, we monitor changes in the environment and trigger new assessments, based on pattern matching using techniques like decision trees or Bayesian Networks. The process also is state based, i.e., the differences between current and previous environmental conditions are detected and suitable events are prepared. Thus, periodic updated data is transformed into events in this stage, marking the boarder-line between the time-triggered and the event-triggered domain. As monitoring systems often operate on a cyclic behavior, and therefore transmit information periodically, the module can also be described as module that encapsulates the time-triggered monitoring domain generating event-triggered messages.

E. Select Messages

Based on the current load of a system, a final selection of the messages for transmission can take place. In this module, also requests from the central decision making unit can be respected, which may indicate whether more or less information is currently required.

F. Encode and Send Data

Finally, the event messages get encoded and are sent to the central decision unit—the I-WAY DSS.

IV. INTERNAL STRUCTURE

As the generation of proper events heavily relies on an adequate representation of the environment, the internal structure of this environment becomes a key part of the system. Basically, it must be easy accessible for fast updates triggered by newly arriving data. We also want to obtain an increase of
the information density, particularly evoked by the sensor/data fusion. Finally, the changes must be tracked and made accessible by providing a sufficient memory depth. We therefore decided to use an UML class diagram to abstract the environment. Each time a new message arrives, it causes a potential update of the status and a new virtual image of the environment is generated based on newly arriving and previously received data. A specific number of older versions of the environment are kept in memory to support further decisions during the information fusion.

**A. Class Diagram**

In Fig. 3 we depict a reduced version of our UML class diagram. It displays five parent classes and two subclasses that are used to track the environmental conditions. In the diagram, the own (ego) vehicle is associated to a lane, which can only exist as part of a road. Further instances associated to the road are abstracted as objects and can be specified into vehicle, constructions areas, etc. An extra class is devoted for general environmental conditions. The class diagram thus displays the current information in a compact format, while additional attributes are stored in the internal attributes of the classes and are easily accessible.

To add memory to the system, we decided to use a mixed model, where we basically store small changes locally in each object. This strategy helps us to exploit the temporal redundancy of data, where multiple frames holding similar information finally increases the reliability. Moreover, we also take periodic snapshots of the whole object diagram, thus allowing for situation estimations also considering a longer time-span.

Please note that the intended storage as a class diagram must not necessarily result in a memory-intensive implementation consisting of objects, etc. Rather, we target a light-weight implementation using simple arrays that have a small memory footprint, however maintaining the same semantics as the class diagram. We thus also reduce the response time.

**B. Update Techniques**

The update of the internal structure implements the first step of the information fusion as the data of different sensors is merged together. However, the information shall not be abstracted (as is done in the event generation step), but packed as densely as possible. For example, messages on critical road conditions ahead, which are issues by different sources, are merged together into one single object. Based on the complexity of the situation, we use simple voting methods or method from statistics like Bayesian Networks or the Dampert-Shafer theory. Moreover, information often comes with a level of confidence, which we merged with previous levels—generating a more reliable representation. By keeping not only the current but also some previous images of the environment, we can merge objects that vanish in one image and reappear in a closely following one at a later time.

Finally, the confidence levels of the environmental parameters are updated. As the new confidence levels are compared to older ones, also an indication of the change in the confidence concerning this object becomes possible.

**V. EVENT MESSAGE GENERATION**

The event message generation sub-module implements the core intelligence of the system. To meet real-time requirements, the generation must occur predictable within a given time-frame. Moreover, the events should indicate a level of criticalness to emphasize them if necessary, while also be neutral to allow the DSS for the final decision making additionally based on current status monitoring, etc.

The domain of pattern matching, where a situation is assessed based on well-defined specifications, optimally serves our requirements. Pattern matching allows us to implement a predictable behavior and a finite response time for the system. The process of classification thereby relies on trees, similar to decision trees, which we use a priori knowledge to decide on the tree structure.

**A. I-WAY Preporcessing Module**

We have implemented a first prototype of the module, see Fig. 4. The preprocessing module shall provide structured and event-based data for the I-WAY DSS, which is extracted from the environmental data collected by the road sensing systems. Concerning the local and near-range environment, there have been four different events identified. These are presented next, always including the source information these events rely on.

**B. Event Details**

There are constantly numerous objects on the road reported. However, only critical ones, then becoming an obstacle, should trigger a generation of an event message. In detail, such an obstacle event comprises the following:

\[
xyzy_\text{ego}lwvhd \rightarrow \text{obstacle event},
\]

which uses \(x, y, z\) coordinates of an object (radar), \(l_{\text{ego}}\) lane of the ego vehicle, \(l\) lane of the object (video), \(w\) width of the object (video + radar), \(h\) height of the object (video + radar), \(v\) velocity of the object/object relative speed (video + radar), \(d\) distance of the object to the ego vehicle (video), and \(v_{\text{ego}}\) velocity of the ego vehicle (CAN). For this type of event, we have implemented a calibration for the fusion of video and radar information, as those systems often use different coordinate systems.

Another event is called the narrow road event, which indicates a narrow road section:

\[
nl_{\text{ego}}w \rightarrow \text{narrow road event},
\]
with \( n \) number of lanes (video), \( l_{ego} \) lane of the ego vehicle, and \( w \) width of the ego lane (video). The lane derivation event even requires less information:

\[
\text{hp ln} \rightarrow \text{lane derivation event},
\]

with \( \text{hp} \) horizontal position of the ego vehicle in current lane (video) and \( \text{ln} \) lane number the vehicle is in (video). Finally, a traffic load event can be issued based on \( n \) number of lanes (video), \( o \) number of objects (video/radar), and \( v_{ego} \) velocity of the ego vehicle (CAN):

\[
\text{n o v}_\text{ego} \rightarrow \text{traffic event}.
\]

The generated events will not always be of use for the ego car, as a narrow road event might be indicated rather late. However, such events are forwarded to other cars and the road infrastructure.

![Diagram of the Preprocessing Module](image)

**Figure 4. I/O of the preprocessing module**

### C. Event Tracking

The occurrence of an event changes the conditions of the system. For example, no two narrow road events should be sent directly following each other, rather there should be a given minimum interval. Moreover, the thresholds to enter a state and to leave a state could be different. For example, to enter the narrow road ahead condition, a lane must be smaller than 3 m; while leaving such a condition, we would require a width of at least 4 m. Thus, we can prevent jitter behavior, where the sending of a narrow event condition occurs multiple times due to an unclear condition. Particularly the narrow event and the lane derivation event require a tracking of the lane information over time. We have thus added attributed to the UML class diagram, which hold the current state of the system, thus also indicating whether an event concerning this already has been sent.

### VI. CONCLUSIONS

This paper presents the design concept of a preprocessing module as basis for integrating signals gathered from multiple data sources within vehicles. The module transforms data from the time to the event domain achieving by this way a significantly reduced final communication load and allowing fast decision making on targeted driver information. Furthermore, the transformation to the event domain ensures no additional delay for sending messages, and therefore can satisfy real-time requirements.

The designed module achieves to remain within the decided configuration an affordable solution for merging signals from multiple data sources. Among the rest of benefits are also the possibility to return real-time notification and the filtering of the delivered information only to the relevant ones. The general concept is in line with the idea to provide information to drivers of significant and critical information of its surrounding but keeping at last the final decision on the driver’s will. Moreover, the preprocessing module can potentially provide a basis for high quality information exchange with other cars or the infrastructure.

Further future activities are targeting the improvement of the system’s performance. Among the planned steps is the partitioning of the preprocessing unit into modules for different ranges in relation to the distance and on hierarchical acting. Furthermore, the current implementation can exploit temporal parallelism by taking advantage from the designed partitioning into sub-modules. The final target is to provide a basis that can be applied into vehicles on a vendor independent basis.

### ACKNOWLEDGMENT

The work is partially funded by the European Commission, Information Society Technologies (IST) as part of the project I-WAY (intelligent co-operative system in cars for road safety), IST-2004-027195.

### REFERENCES


