

State Machine Based Method for Consolidating Vehicle Data

Florian Dittmann¹, Konstantina Geramani¹, Victor Fäßler¹, and Sergio Damiani²

¹ TWT GmbH Science & Innovation, Bernhäuser Str. 40 -42,
73765 Neuhausen, Germany

{florian.dittmann,konstantina.geramani,
victor.faessler}@twt-gmbh.de

² Centro Ricerche Fiat S.C.p.A., Strada Torino, 50
10043 Orbassano (TO), Italy
sergio.damiani@crf.it

Abstract. The increasing number of information and assistance systems built into modern vehicles raises the demand for appropriate preparation of their output. On one side, crucial information has to be emphasized and prioritized, as well as relevant changes in the driving situation and surrounding environment have to be recognized and transmitted. On the other side, marginal alterations should be suitably filtered, while duplications of messages should be avoided completely. These issues hold in particular when assistance systems overlap each other in terms of their situation coverage. In this work it is described how such a consolidation of information can be meaningfully supported. The method is integrated in a system that collects messages from various data acquisition units and prepares them to be forwarded. Thus, subsequent actions can be taken on a consolidated and tailored set of messages. Situation assessment modules that rely on immediate estimation of situations are primary recipients of the messages. To meet their major demand—rapid decision taking—the method generates events by applying the concept of state machines. The state machines form the anchor to merge and fuse input, track changes, and generate output messages on higher levels. Besides this feature of consolidating vehicle data, the state machines also facilitate the transformation of continuous data to event messages for the rapid decision taking. Eventually, comprehensive driver support is facilitated, also enabling unprecedented features to improve road safety by decreasing the cognitive workload of drivers.

Keywords: automotive domain, time-triggered, event-triggered, information fusion, state machines, data aggregation, architectures and models.

1 Introduction

The number of advanced driver assistance systems (ADAS) and in-vehicle information systems (IVIS) is constantly increasing in modern vehicles. Despite ADAS and IVIS systems are implemented to the benefit of the driver, a not proper integration within ADAS/IVIS easily can impose a hard-to-manage workload. At the same time,

interoperation between those systems should be fostered to exploit their capabilities for improving road safety. Meaningful combination of their output could even lead to more useful information with potential added benefit to the safety. For example radar and vision for the lane detection can cooperate to better describe the scenario ahead.

The method was designed and developed in the course of the European Commission funded I-WAY project [9], where it forms the pre-processing system for a situation assessment module. The situation assessment module takes over the responsibility to finally rate incoming information by rapid decision taking, performing tailored user warning. Besides reducing the complexity, the pre-processing module was also decided to be implemented for becoming a clear-defined border between time-triggered or continuous information and event messages on higher levels.

To achieve the overall demands of meaningfully combining and consolidating vehicle data, well-thought-out organization of the information issued by IVIS and ADAS is required. Only then, their output can undergo a rapid assessment and combination to become beneficial for the driver. In particular, a situation assessment system that takes vehicle status as input and intends to warn drivers in a sensible way will operate best if the information is available in a well-organized style. Hence, appropriate information fusion and subsequent filtering should take place to allow for presentation of messages in a consolidated manner.

The increasing number of ADAS and IVIS systems to be integrated has to be taken into account: this can be met by fostering a well-defined and modular architecture to manage the complexity. To organize the data meaningfully and facilitate rapid decision making, it is proposed a suitable transformation of information to higher levels as inevitable key factor, reflected in the overall architecture. The objective of the fusing thereby is to combine input information of different sources over time and only forward significant messages.

In this work, a method based on state-machine approach to meet the above requirement of transforming information fusion to a higher level is presented. The method derives from a dedicated unit for processing and consolidating the vehicle relevant data. The unit takes over the responsibility to merge and fuse, filter and suppress, generate and trigger output tailored to the requirements of subsequent systems, which decide when and in which way to display the information. The state machine combines time- and event-triggered input and allows the issuing of messages in an event-based style. In some cases the exploitation of state machines facilitates the method to transform information fusion to a higher level.

The unit was implemented to be portable, generic and suitable for multiple purposes. The rest of the paper is organized as follows. In the next two sections, we review related work and give an overview of the architectural integration. The application of the state machines is discussed in section 4 on the method details. Section 5 shows an exemplary implementation before section 6 concludes the work.

2 Related Work

Information fusion in the automotive domain can be found in several works, for example [2], [3]. These works often focus on advanced driver assistance systems. A tracking based information fusion technique is presented in [15]. It tracks data of

sensors having complementary or/and redundant field of views. The technique of multi-level fusion for vehicular environment recognition is discussed in [14]. Here fuzzy operators are used and confidence levels are introduced. In [16] the architecture of the SASPENCE project is presented, which introduced separate fusion modules for individual features in a vehicle. A layered data-fusion process is discussed in [17]. Object as well as situation refinement is targeted. In [18], a step based interaction and communication assistant is presented, which defines a technique how to process information for HMI in vehicles.

In general, information fusion fundamentally discusses the combination of data extracted from multi sensors 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 [1]. A comprehensive overview on techniques and methods of information fusion is given in [10]. Information fusion for multimedia data analysis is discussed in [11]. While the application area is different, the methods and concepts of the reference are very valuable for this work. More recently, there has been a push to extend systems to multiple sensing modalities. Multi-sensor fusion using complementary sensing modalities greatly increases the robustness of sensing systems [4].

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, while [6] discusses whether to use time- or event triggered communication as a non-functional feature that requires early consideration in the development process.

To conclude, substantial work on information fusion also in the automotive domain has been done. The approach presented in this paper builds on the results and complements them by exploiting a state machine based method that allows for generating tailored event-triggered messages for processing and decision making on higher levels.

3 Architectural Integration

The objective of the method is the preparation and provision of high-level output for subsequent systems, particularly fusing input information of different sources over time and only forwarding significant messages. By evaluating these high level messages rapid decision making can take place. Actions necessary to be performed can be issued, e.g., generating warning messages to the driver. The method therefore performs advanced information fusion by applying a transformation from time-triggered to event-triggered, a transformation from the space-time domain to the event domain. This can be abstractly described as follows:

$$XYZ \text{ and } dX/dt, dY/dt, \text{ and } dZ/dt, c \rightarrow \text{event}, \quad (1)$$

where XYZ denotes the physical location of data (e.g. critical objects referring to the vehicles environment), dX/dt denotes the temporal changes of these coordinates, and c other conditions. The latter covers global changes like entering a tunnel, which can be derived from digital maps based on GPS localization. State machines thereby become

the primary technique to facilitate the transformation from the (continuous) input data into event messages (see Section 4).

On top of this basic behavior, other features are added to the overall performance: 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 environment of a car. This tracking mechanism introduces a memory in the system; as consequence a more reliable decision becomes possible and multiple sources basically can increase the data's reliability and the confidence level. Concepts of temporal redundancy are exploited therefore.

The method implements early sensor fusion: environmental information gained by multiple acquisition modules are analyzed and combined to generate high-level data, with potential higher reliability. To avoid data degradation, the implementation reflects a negotiation to a canonical reference base to combine data from different monitoring sources.

For maintenance and complexity reasons, the method is enveloped in five logical steps, which each take over a significant part of the overall processing. The steps then are executed cyclic, triggered by newly arriving input data. Figure 1 depicts the logical steps and indicates their order of execution. The two core steps *information fusion* and *event generation* are preceded by a *basic plausibility analysis*, which ensures valid data in terms of range, resolution, etc. In the two surrounding steps, data from acquisition units is received and decoded, as well as encoded and send to listening systems.

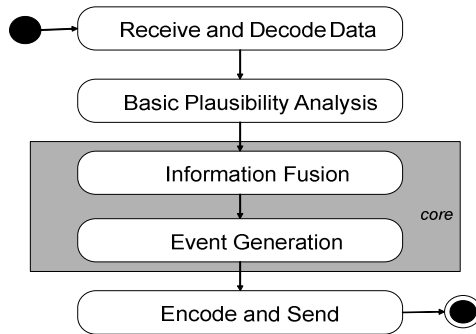


Fig. 1. Logical steps of the processing architecture

Well-defined communication techniques are used for interaction with the input and output systems. The overall communication architecture was designed to allow for seamless integration of additional input devices, enabling future extensions. Newly arriving messages trigger the cyclic behaviour, irrespective of their origin or source. The architecture thus allows producing synergy between different products / systems from different vendors. Eventually, it provides at least a means how to synchronize input data.

Besides the physical communication requirements, also the timing characteristics of the communication are of special concern: the system shall operate in vehicles on roads requiring real-time behaviour. A solid strategy was developed and integrated to target the real-time requirements. It is based on time stamps that are crucial to prevent

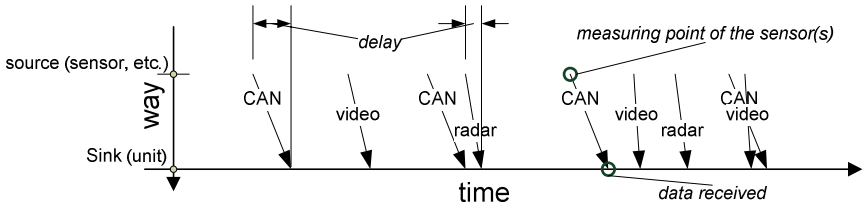


Fig. 2. Messages over the time axis

wrong behavior. Figure 2 depicts the situation along a time axis showing the message travelling times. In the example, different messages travelling times can impose a false ordering of messages when they arrive at the unit. Time stamps then ensure a correct sorting.

4 Method Details

The method has to handle sources of input in an adequate manner, fuse the information when appropriate, generate events and send messages. The general concept of information fusion thereby is based on: physical redundancy (the same kind of information is given by multiple sensors); temporal/knowledge based redundancy (the same sensor is monitored over time and conclusions are drawn by referring to an expected behaviour of the sources). Moreover, the exploitation of the concept of temporal redundancy (information is tracked over time and historical knowledge is taken into account to generate events) is realized giving to the system the capability to take memory / history.

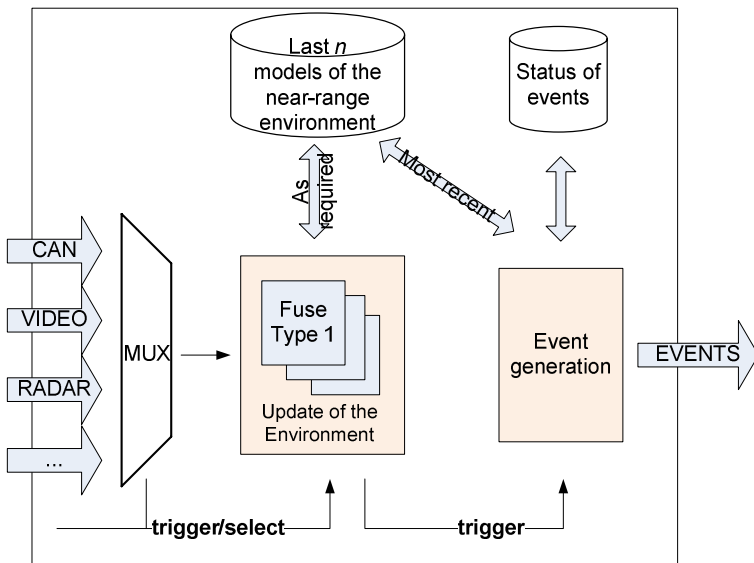


Fig. 3. The internal architecture: including input and output

In Figure 3 the internal structure of the method is depicted. Besides an input message selection mechanism (multiplexer MUX), the pillars are: the fusion strategies and the event generation section; databases to hold former snapshots (models) of the environment and the current status of the events. The data base for holding the status of the events thereby plays a central role and it is implemented using state machines. It becomes the means to generate messages in a structured, adaptable and controlled manner. In the following, the information fusion and the event generation are discussed in detail.

4.1 Information Fusion

The information fusion step evaluates input data and combines them into a virtual representation of the environment, which is also used to track the environment. It is implemented as internal database holding the last n models. The virtual representation has a double functionality: it does not only hold 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, adding memory to the system.

The update of the environment is done by referring to multiple sources, exploiting the redundancy of them, also using temporal redundancy of the same source. The fusion thereby also includes an advanced plausibility analysis taking place on a higher level. The analysis performs checks on the physical logic of the data. For example, input data that denotes a high velocity is only accepted, if the RPM (rates per minute) of the engine also has shown some high values in the close past.

Eventually, the system performs an information fusion based on newly arriving data and old data, originating from multiple sensors (complementary redundancy) or the same sensors (temporal redundancy) over time. The structure allows information storing also in the case that it is of no relevance at the time when it is received. Obviously at this level redundant information can be removed.

Information often comes with a level of confidence, which is used to perform the merging with previous levels, generating a more reliable representation. By keeping not only the current but also some previous “images of the environment”, the system can merge objects that vanish in one “image” and reappear in a closely following one at a later time. In general, incoming information is already related to previous information in terms of temporal redundancy. Monitoring signal characteristics over time also can help to identify untypical behaviour. Thus, peaks of values, which could describe malfunctioning, can be sorted out and ignored for the event generation.

There is not a unique fusion technique usable for all inputs. As consequence, depending on the input message at the multiplexer level, the appropriate fusion algorithm is selected and triggered. Thus, individual fusing algorithms are used for the variable sources of input.

During the update of the internal structure the system resembles the data core of the information fusion as the data of different sensors merging all together. False information thereby is filtered before event messages are generated, adding a high-level plausibility analysis to the method. However, the information is not abstracted (as done in the event generation step), but packed as densely as possible. For example, messages on critical road conditions ahead, which are issued by different sources, are merged together into one single data.

4.2 Event Message Generation

The event message generation step implements, with the fusion technique that here is not described, the core of the method. It refers to the virtual representation of the environment, evaluates the current values and issues events, transforming the low (raw) input data to a higher level of abstraction. Thereby, also the amount of information for a listening system is reduced. For example, an HMI can make rapid decision and issue appropriate messages, thereby also becoming compliant to principles of the EU project AIDE, which is currently undergoing an ISO standardization process. Nevertheless, the immediate transmission of alarm from a monitoring system (e.g. radar) is foreseen also. On the other hand a level of criticalness is forwarded for rapid decisions to subsequent units. The level allows for rating the events in a more consolidated manner. Besides these functional requirements, the event generation method focuses on reaching real-time requirements, by guaranteeing predictable decisions within a given time-frame.

The method relies on state machines. Each state defines an event and when entering the state an event message is sent. The decision to change a state is based on decision trees. Figure 4 displays the concept. The process of classification thus relies on these trees, where a priori knowledge is used to decide on the tree structure. The deterministic depth / length of the decision trees thereby allows guaranteeing the real-time behaviour.

The system reacts on changes in the environment and triggers new assessments. The process is state based, i.e., the differences between current and previous environmental conditions are detected and suitable events are prepared. Figure 5 displays an example for lane change events. Please note that the reduced number of states describing the location of vehicles allows for rapid decision making on issuing warning messages to drivers. Summarizing the concept behind the state machine of Figure 5, the vision system produce only information on the vehicle position in the lane, the proposed system takes care to evaluate the lane change.

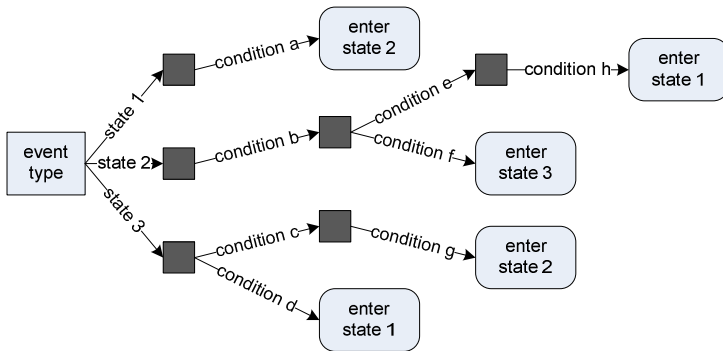


Fig. 4. The concept of decision trees for the state changes

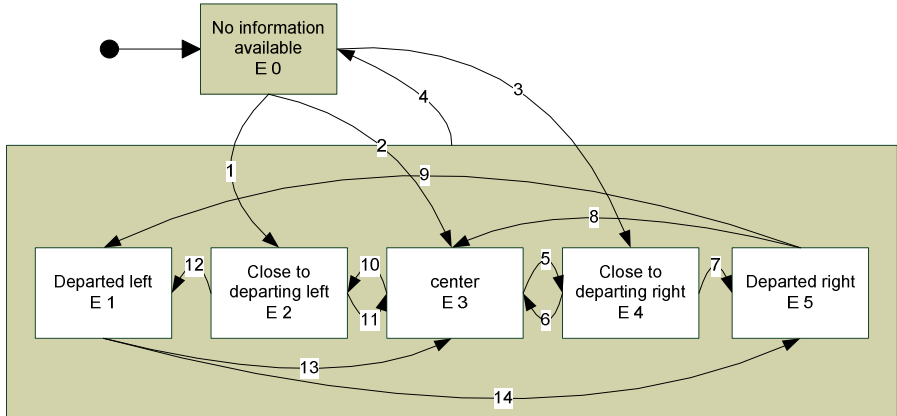


Fig. 5. Exemplary Event Generation

The decision trees to be followed are hidden behind the transitions of the state machine and identified by numbers in the figure. A table is used to hold the details on the decision trees. The source of information for the event message generation—exploited by the decision trees—is twofold: on one hand, the virtual representation of the environment—as presented above—serves as major source of input; on the other hand also the previous state of the events is required, avoiding the sending of the same event twice.

In the end, periodic updated data is transformed into events in this stage, marking the final boarder-line between the time-triggered and the event-triggered domain. As monitoring systems often operate on a cyclic behaviour, and therefore transmit information periodically, the method thus generates event-triggered messages

The occurrence of an event changes the conditions of the system. For example, the same events should not be sent twice (directly following each other), rather a minimum interval should be respected. The state machine ensures this functionality. Moreover, the thresholds to enter a state and to leave a state could be different. For example, to classify an object as being an obstacle, the distance should be smaller than a given value x m; while leaving such a condition, we would require a distance of at least $x + Y$ m. Thus, jitter behaviour can be prevented, where the sending of a specific event condition occurs multiple times due to an unclear condition. This requirement can be implemented through suitable selection and definition of the rules attached to the transformations of the state machines.

5 Prototype Implementation

The above concepts and ideas have been tested and implemented within the I-Way project for the implementation of a pre-processing module embedded to an overall architecture of a decision support system [8, 13].

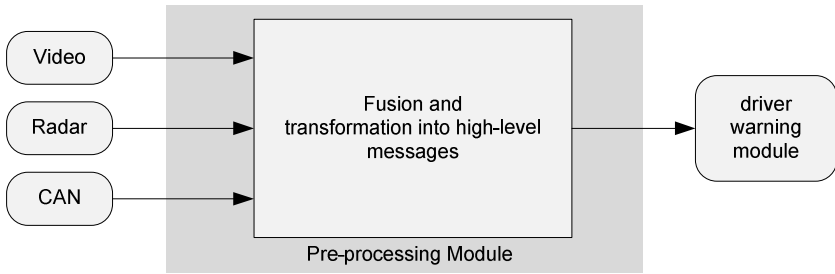


Fig. 6. Pre-processing module: prototype implementation

The decision for implementing a pre-processing module has been made upon the need to bundle data gathered from the near and medium range of a vehicle, including the vehicle itself and providing them on a higher level to the decision support system.

The scope of the pre-processing module is to facilitate the decision making process and enable fast reactions to issue alerts on time by reducing the overall data traffic of the final system. The advantage in that case is that higher reliability is achieved while the probability of error detection in the final system is reduced and the confidence in sensor observations is increased. Moreover, the combining and pre-processing of the above gathered information within a separate module gives the overall system both modularity and reduced complexity, permitting by this way easier integration and faster testing. Further possibilities as detecting the plausibility of the collected data have also been explored in order to secure the correctness of the final decisions at the end system and to avoid overlap of the collected data.

The processed data are derived from physically separated sources (Radar, video and vehicle network CAN), sending their outputs to the pre-processing system for decoding, fusing and transformation (ref. to Figure 6). The decision to enhance video information with radar and CAN data is justified upon the fact that video detection algorithms strongly depend on weather and lighting conditions thus need to be complemented with other sensing modalities.

The communication is based on TCP/IP messages sent via Sockets. The distribution of the modules to multiple and different hosts thus becomes possible, only requiring Ethernet access for the communication. For receiving messages, the unit runs one single server at a single port. Each input device has to start a client and connect to the server of the unit. The server thereby was designed to accept multiple connections of different clients at the same time. For the output messages, the unit itself starts a client and requests a connection to a server. Again, this connection is kept alive and messages (events) are transmitted directly.

Event generation state machines have been implemented for event messages classified into object detection, narrow road, and lane change as well as weather conditions. The method has been selected as it permits much more reliable and robust situational awareness.

6 Conclusion and Outlook

A method is presented that transforms data into the event domain in a structured manner leveraging ADAS/IVIS in modern vehicles. The state machine based technique ensures a controlled behaviour. Through the technique of event generation, the method achieves the submission of information in a precise manner, both in terms of content and timeliness. The messages forwarded are of high-level information and they are only submitted in case their sending is of value—significant changes have occurred. Thus, the amount of input information for any listening module is reduced to a well manageable amount. Moreover, real-time requirements are respected during the design and implementation. The modular structure of the method allows for easy extensions and maintainability.

As an enhancement to the current system, the thresholds for the decision trees could be made adaptable during run-time. The listening system then can ask for more fine or coarse grain information, depending on current condictions. Thereby, an in-field adaptation and customization can be achieved. Moreover, the straight forward technique of decision trees could be replaced by more complex decision making techniques like support vector machines or Bayesian networks. However, real-time capabilities as given for the deterministic decision trees have to be ensured also for support vector machines or Bayesian networks.

To conclude, the method resembles a technique to achieving safety for all in a cost efficient and effective manner. It proposes a technique for consolidating vehicle data to be forwarded as high level events. The events can be used for rapid decision making by modules that issue tailored messages to the benefit of drivers and the safety on roads in general.

Acknowledgements. The work was partially funded by the European Commission, Information Society Technologies (IST) as part of the project I-WAY: IST-2004-027195.

References

1. Luo, R.C., Kay, M.G.: Multisensor integration and fusion in intelligent systems. *IEEE Trans. On Systems, Man, And Cybernetics* 19(5), 901–931 (1989)
2. Kopf, M., Simon, J.: A concept for a learn-adaptive advanced driver assistance system. In: *Conference on Cognitive Science Approaches*, Neubiberg (September 2001)
3. Yoshida, T., Kuroda, H., Nishigaito, T.: Adaptive driver-assistance systems. *Hitachi Review* 53(4) (November 2004)
4. Ploetner, J., Trivedi, M.: A Multimodal Approach for Dynamic Event Capture of Vehicles and Pedestrians. In: *VSSN 1996 Proceedings of the 4th CM international workshop on Video surveillance and sensor networks* (October 2006)
5. Obermaisser, R.: *Event-Triggered and Time-Triggered Control Paradigms*. Springer Series: Real-Time Systems Series, vol. 22 (2005)
6. Scheler, F., Schröder-Preikschat, W.: Time-Triggered vs. Event-Triggered: A matter of configuration? In: *MMB Workshop Proceedings GI/ITG Workshop on Non-Functional Properties of Embedded Systems Nuremberg*, pp. 107–112. VDE Verlag (2006)

7. Kopetz, H.: Event-triggered versus time-triggered real-time systems. In: Karshmer, A.I., Nehmer, J. (eds.) *Dagstuhl Seminar 1991*. LNCS, vol. 563, pp. 87–101. Springer, Heidelberg (1991)
8. Rigas, G., Katsis, C.D., Bougia, P., Fotiadis, D.I.: IWAY: Towards Highway Vehicle-2-Vehicle Communications and driver support. In: *The IEEE Conf. on Systems, Man, and Cybernetics* (2008)
9. Rusconi, G., Brugnoli, M.C., Dosso, P., Kretschmar, K., Bougia, P., Fotiadis, D.I., Salgado, L., Jaureguizar, F., De Feo, M.: I-WAY, intelligent co-operative system for road safety. In: *IEEE Intelligent Vehicles Symposium*, June 13-15, pp. 1056–1061 (2007)
10. Nakamura, E., Loureiro, A., Frery, A.: *Information Fusion for Wireless Sensor Networks: Methods, Models, and Classifications*. *ACM Computing Surveys* 39(3), Article No. 9 (2007)
11. Wu, Y., Chang, E., Chang, K., Smith, J.: Optimal multimodal fusion for multimedia data analysis. In: *12th ACM International Conference on Multimedia*, pp. 572–579 (2004)
12. Albert, A.: Comparison of Event-Triggered and Time-Triggered Concepts with Regard to Distributed Control Systems, *Embedded World*, Nuremberg, Germany, pp. 235–252 (2004)
13. Dittmann, F., Geramani, K., Rigas, G., Katsis, C., Fotiadis, D.: Towards Advanced Information Fusion for Driver Assistant Systems of Modern Vehicles. In: *Proceedings of the 68th IEEE Vehicular Technology Conference (VTC)*, Calgary, Alberta, Canada, September 21-24 (2008)
14. Lindner, P., Scheunert, U., Richter, E.: Multi Level Fusion for Environment Recognition. *ProFusion2 e-Journal* 2, 24–30 (2008)
15. Floudas, N., Tsogas, M., Amditis, A., Polychronopoulos, A.: Track Level Fusion for Object Recognition in Road Environments. *ProFusion2 e-Journal* 2, 16–23 (2008)
16. Tango, F., Saroli, A., Cramer, H., Floudas, N., Da Lio, M., Biral, F.: The SASPENCE Project: data-fusion and other data-processing modules. *ProFusion2 e-Journal* 2, 31–35 (2008)
17. Ahlers, F., Schendzielortz, T., Tango, F., Lytrivis, P., Zott, C., Fürstenberg, K.: Data Fusion Structure for the SAFESPOT Platforms. *ProFusion2 e-Journal* 2, 48–58 (2008)
18. <http://www.aide-eu.org/>