Data synchronization strategies for multi-sensor fusion

Nico Kaempchen
BMW

DATA SYNCHRONIZATION STRATEGIES FOR MULTI-SENSOR FUSION

Nico Kaempchen, Klaus Dietmayer
University of Ulm, Department of
Measurement, Control and Microtechnology
Albert-Einstein-Allee 41, D-89081 Ulm, Germany
Tel: +49 731 50 26326 - Fax: +49 731 50 26301
E-mail: {Nico.Kaempchen, Klaus.Dietmayer}@e-technik.uni-ulm.de

SUMMARY

Advanced driver assistance systems (ADAS) have increasing demand for several sensor systems. Fusing multi-sensor data enlarges the field of view and increases the certainty and precision of the estimates. A crucial part of a fusion system is the data association, which requires data synchronization. The major synchronization strategies for data fusion are discussed and contrasted with respect to their usability in ADAS.

INTRODUCTION

Recent projects on driver assistance systems have focused on applications such as Pre-Crash (Chameleon) [1], ACC Stop&Go (Carsense) [2] and recognition of vulnerable road users (PROTECTOR). These advanced driver assistant systems (ADAS) have increasing demand for several sensor systems, which are complementary but also redundant. Much research has therefore been focused on high-level multi-sensor fusion [2, 3, 4, 5]. The aim of such a step is to provide a fused description of the traffic scene surrounding the vehicle, which is relevant for ADAS, but not specific for a certain application. This fusion system incorporates the data of the diverse sensors into a single description. Thus the field of view of a single sensor is enlarged, the certainty and precision of the estimates is increased and additionally this system design is economically efficient, as different applications share a set of sensors.

A crucial part of such fusion systems is the data association. In order to update the object states of the environment description, the sensor data must be associated with the environment description. Correct association requires a synchronization of the sensor data and the associated object state. Different synchronization strategies will be discussed with respect to the common automotive sensors and applications.

SENSOR FUSION ARCHITECTURE

Established advanced driver assistant systems, such as lane departure warning (LDW) and automatic cruise control (ACC) have a common architecture, as shown in Figure 1a. The application is implemented for a special sensor configuration and therefore depends on these sensors. With an increasing number of applications which share a set of sensors, the sensor dependent part of the software would exist several times, i.e. found in each application (see Figure 1b). Thus emerges a strong motivation to separate the sensor specific part of the system and use it as an interface between the sensors and the applications (see Figure 1c). The sensor fusion module is responsible for collecting measurements, interpreting them sensor specifically and incorporating them into a unified, consistent description which is then forwarded to the applications. The sensor fusion must be in part sensor specific in order to obtain a maximal profit of each sensors measurement.

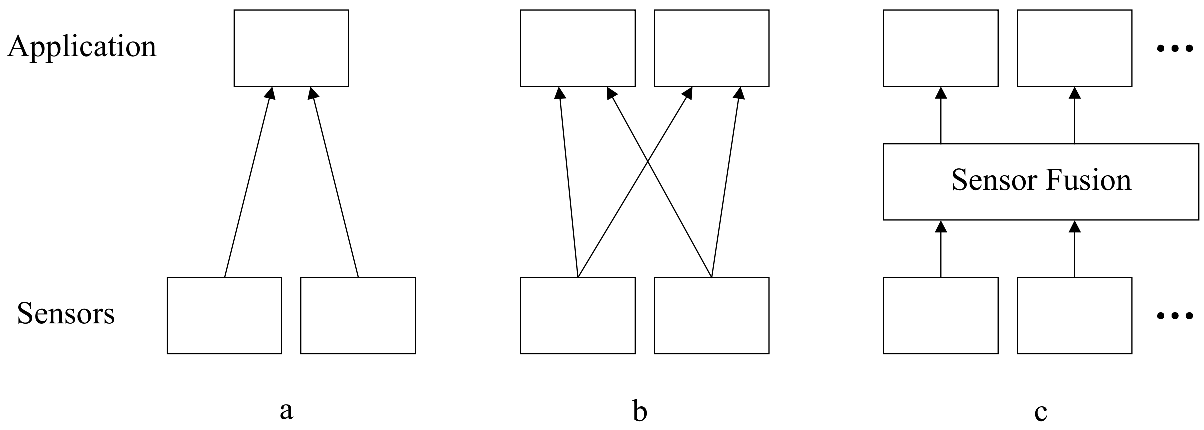

Figure 1: Driver assistant system architectures. (a) A single application which depends on one or several sensors. (b) Several applications each based directly on one or several sensors. (c) Possible future advanced driver assistant system architecture with common sensor fusion layer, thus separating the application from sensor specific implementations.

A general architecture of a sensor fusion system is contains the basic components shown in Figure 2 [6]. The environment description is composed of a set of objects, each of which is defined by an object state. There are different object models for trucks/busses, passenger cars, (motor) bikes, pedestrians and for stationary objects. Additionally the road is modeled as well as the ego-vehicle.

The objects of the environment description are predicted in order to synchronize them with the incoming sensor data. The object states are transformed into the parameter space of the sensor data by the use of inverse sensor models and the association is performed. In case of an established association the sensor data is integrated into the object state. An object management handles the generation and deletion of objects in the environment model.

As mentioned, the sensor fusion is not independent from the sensor's characteristics. The sensor specific modules are the inverse sensor model and the association. These are the only components of the sensor fusion layer which need to have some knowledge of the sensor's characteristics. The integration and prediction may be performed using a Kalman-filter.

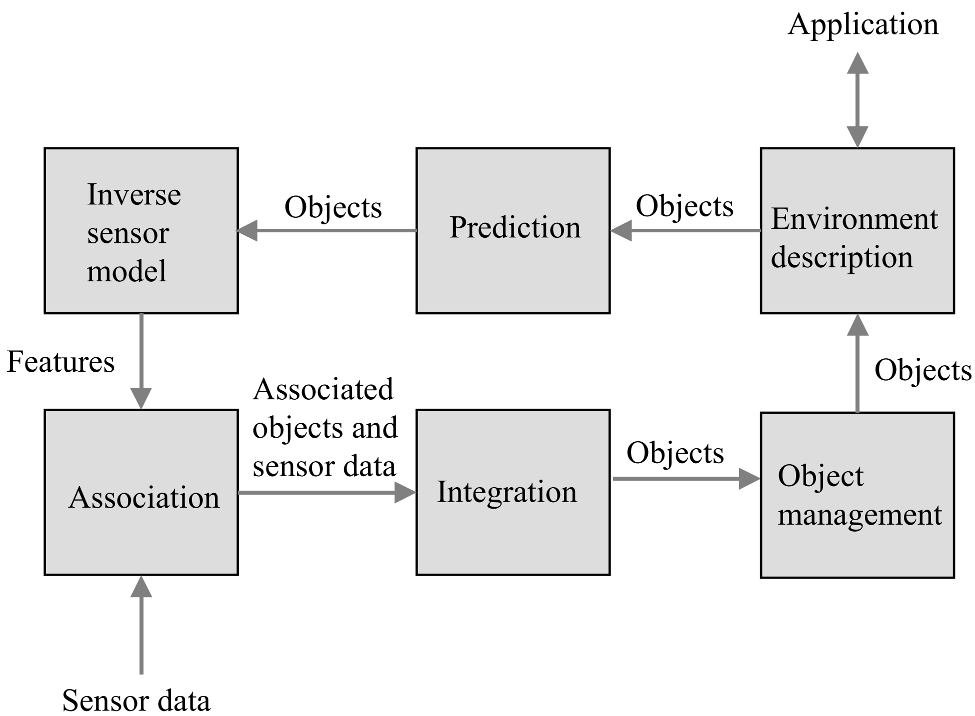

Figure 2: General architecture of a sensor fusion system which works as an interface between sensors and applications.

SYNCHRONIZATION

Sensor fusion combines several sensor measurements in order to obtain an enhanced object state estimation. The association and integration of sensor data requires its synchronization with the environment description. The environment description is predicted for the time of measurement. Thereafter the sensor data can be associated and the object states updated. In order to obtain a precise synchronization a sufficiently accurate global time for all sensors and the fusion system is necessary.

In order to obtain a time-consistent object state estimation the measurements have to be integrated in the order, they were acquired. This is a crucial constraint for a sensor fusion system. Depending on the synchronization strategy, measurements have to be buffered thus introducing a latency into the fusion system. For very time-critical ADAS such as Pre-Crash or automatic emergency brake, the overall system latency is of high interest. The advantage of such ADAS is drastically reduced if their reaction time is too long with respect to the high speed of passenger cars and therefore short time to contact in extreme situations. The aim is to minimize the systems latency which in our case is the latency introduced by the sensor fusion process. Different synchronization strategies lead to varying worst-case sensor fusion latencies which are discussed in the remainder of this paper.

Measurement frequency and latency

Assuming a general multi-sensor configuration a sensor fusion system has to cope with different and varying sensor measurement frequencies, the sensors measurement latency as well as asynchronous measurement times. The measurement frequency f_{sensor} is defined by the number of times the sensor delivers data to the sensor fusion system per second. While Laserscanners perform measurements at a constant frequency, due to a constantly rotating head [7], a vision system might perform a new measurement once it processed the old sensor data. If the processing time varies with respect to the sensor data, the measurement frequency changes over time. The measurement latency $T_{L,sensor}$ is defined by the measurement acquisition time, the pre-processing time and the latency introduced by the communication transfer time before the sensor fusion system receives the measurement (Figure 3). The latter can be neglected if the high-level sensor measurements are of small size compared to the transfer rate of the network. The reference for the measurement time is the beginning of the data acquisition. Figure 3 shows the general case of a sensor measurement process where the pre-processing may be performed parallel to acquisition and transfer. In addition to the measurement latency $T_{L,sensor}$, the sensor is characterized by its measurement cycle duration $T_{C,sensor}$ which is defined by the measurement frequency:

$$T_{C,sensor} = \frac{1}{f_{sensor}} \quad (1)$$

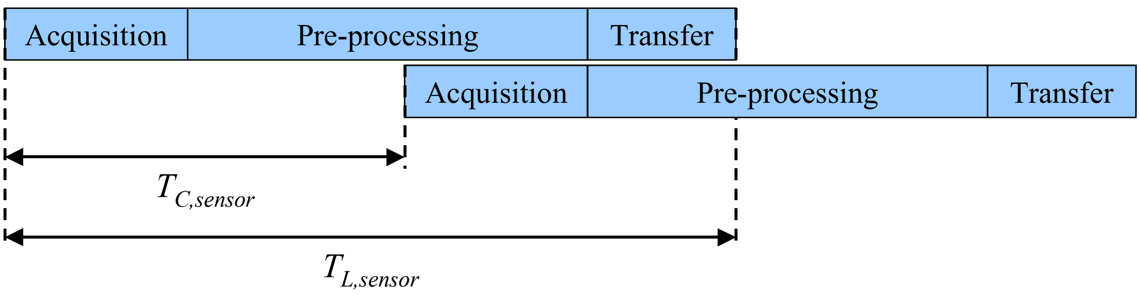

Figure 3: The measurement latency $T_{L,sensor}$ is composed by the acquisition time, the pre-processing and the time for the data transfer to the sensor fusion system. The sensors measurement cycle duration is $T_{C,sensor}$.

In general $T_{L,sensor}$ and $T_{C,sensor}$ are varying over time. As for time-critical ADAS the worst-case sensor fusion latency is of interest, we define for the remainder of this paper $T_{L,sensor}$ as the worst-case measurement latency and $T_{C,sensor}$ as the worst-case measurement cycle duration of a sensor. We assume that the sensors provide a worst-case latency and cycle duration which is necessary for time-critical ADAS.

Sorting in the non-deterministic configuration

Consider a sensor configuration depicted in Figure 4 with a computer vision system (CV), a Laserscanner (LS) and a long-range radar (R) which have an overlapping field of view . Each sensor takes a raw data measurement, pre-processes the data and propagates the result to the sensor fusion system. The sensors are working asynchronously, there is no knowledge about future measurement times or the sensor latencies, nor is the measurement frequency considered to be constant. This configuration is therefore referred to as the non-deterministic approach.

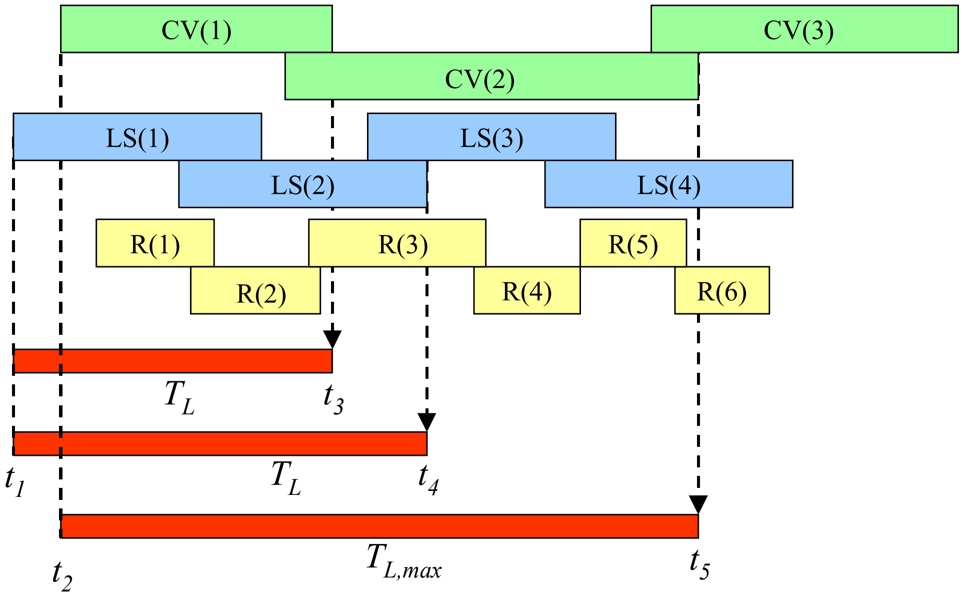

Figure 4: Worst-case latency $T_{L,max}$ introduced by the measurement sorting of the sensor fusion system in the non-deterministic configuration.

As recursive filters such as the Kalman-filter only allow for a prediction into the future, the measurements which are integrated into the Kalman-filter have to be in sequence. This however imposes an undesired constraint on the example in Figure 4. Before any measurement may be integrated into the Kalman-filter, the sensor fusion system must have received at least one measurement from each sensor (CV(1), LS(1), R(1)). At the real time t_3 the oldest measurement (LS(1)) can be integrated, thus updating the environment description to time t_1. Considering the sensor fusion calculation time T_{comp}, the environment description which is propagated to the application has an overall latency $T_{L,env}$ of

$$T_{L,env} = T_L + T_{comp} \qquad (2)$$
$$T_{L,env} = t_3 - t_1 + T_{comp}$$

In the following we will focus on the latency introduced by the sorting of the measurements T_L.

Update time	A priori latency T_L	A posteriori latency T_L	Integrated measurements
t_3	-	$t_3 - t_1$	LS(1)
t_4	$t_4 - t_1$	$t_4 - t_2$	CV(1)
t_5	$t_5 - t_2$	-	R(1), LS(2), R(2), CV(2)

Table 1: Latencies in the non-deterministic configuration.

Continuing the example in Figure 4 at time t_4 the next update can be performed, as the sensor fusion has by then stored at least one measurement per sensor. Until this update, the latency T_L has increased constantly up to $t_4 - t_1$ (a priori latency in Table 1). The sensor measurement CV(1) is integrated and the latency reduced to $t_4 - t_2$ (a posteriori latency in Table 1). Before the next update at t_5 the a priori latency is $T_L = t_5 - t_2$. In this example the worst-case latency

$T_{L,max}$ is reached, depending in the non-deterministic configuration on the maximal worst-case sensor latency max($T_{L,sensor}$) and the worst-case measurement cycle duration max($T_{C,sensor}$):

$$T_{L,max} = \max(T_{L,sensor}) + \max(T_{C,sensor}) \quad (3)$$

The aim is to minimize this worst-case latency in order to minimize the overall sensor fusion latency.

Deterministic configuration

By opposition to the non-deterministic case, the sensors are now considered to be able to provide the measurement times of at least one preceding measurement. In general this deterministic configuration is implicitly given if the sensors measure at constant frequencies. However even a sensor with a varying frequency can easily provide information about the next measurement time once its pre-processing is finished (see Figure 3). The sensor could submit the next measurement time to the fusion system together with the actual measurement.

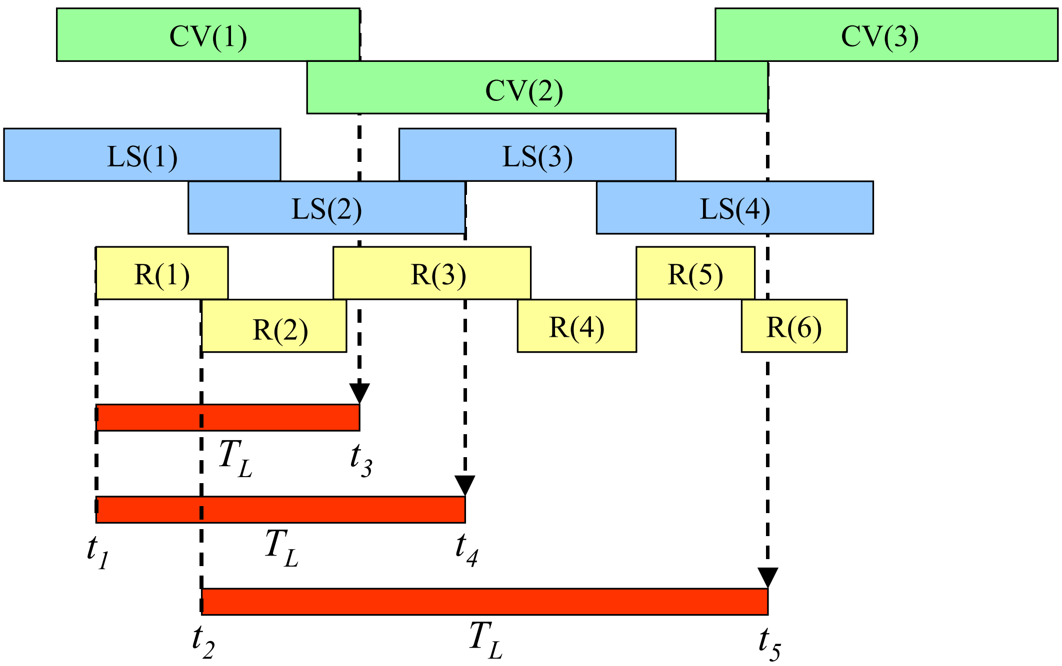

Figure 5: Latencies T_L introduced by the measurement sorting in the deterministic case.

In the example depicted in Figure 5 the measurements CV(1), LS(1), R(1) and R(2) are all received by the sensor fusion system by the time t_3. Therefore in the deterministic configuration, the measurement times of CV(2), LS(2) and R(3) are known. This enables the fusion system to directly integrate the radar measurement LS(2) and R(2) at time t_4, as it is known that CV(2) will be measured subsequently. In the non-deterministic configuration the sensor fusion system would have to wait with the integration of LS(2) and R(2) until at least CV(2) was received in order to be sure to meet the preliminary of timely ordered measurement integration. Table 2 summarizes the a priori and a posteriori latencies of the example in Figure 5 for the deterministic case.

Update time	A priori latency T_L	A posteriori latency T_L	Integrated measurements
t_3	-	$t_3 - t_1$	LS(1), CV(1), R(1)
t_4	$t_4 - t_1$	$t_4 - t_2$	LS(2), R(2)
t_5	$t_5 - t_2$	-	CV(2), R(3), LS(3), R(4)

Table 2: Latencies in the deterministic configuration.

From the example above it seems obvious that the mean latency could be reduced with deterministic working sensors. However for time critical applications not the mean, but the worst-case latency $T_{L,max}$ is most important. It is given by the maximal worst-case sensor latency $\max(T_{L,sensor})$ and the least worst-case sensor cycle duration $\min(T_{C,sensor})$:

$$T_{L,max} = \max(T_{L,sensor}) + \min(T_{C,sensor}) \tag{4}$$

The resulting overall latency introduced by the fusion system is given in Equation 2.

Synchronous sensors

A third configuration considers the sensors to be synchronized. As shown in Figure 6 this implies a synchronization on the slowest sensor which is in this case the video sensor (CV). The approach is deterministic as all measurement times have to be known in advance.

Even if actual automotive sensors seldom offer a synchronization, it is feasible and therefore of theoretical interest in this comparison of synchronization strategies.

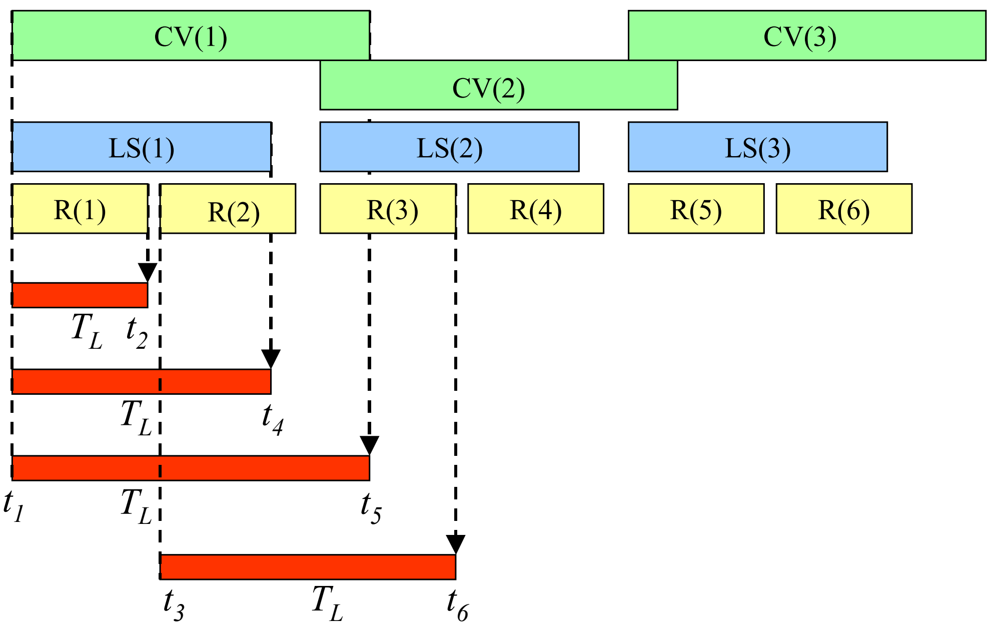

Figure 6: Latencies T_L introduced by the measurement sorting in the configuration with synchronous sensors.

Consider the example in Figure 6 which is summarized in Table 3. The measurements R(1) and LS(1) can immediately be integrated after their reception as CV(1) is measured at the same time as LS(1) and R(1). Only the integration of R(2) is delayed until CV(1) is received in order to meet the preliminary of timely ordered integration of measurements.

Update time	A priori latency T_L	A posteriori latency T_L	Integrated measurements
t_2	-	$t_2 - t_1$	R(1)
t_4	$t_4 - t_1$	$t_4 - t_1$	LS(1)
t_5	$t_5 - t_1$	$t_5 - t_3$	CV(1), R(2)
t_6	$t_6 - t_3$	-	R(3)

Table 3: Latencies in the configuration with synchronous sensors.

The worst-case latency $T_{L,max}$ is given by the maximal worst-case sensor latency max($T_{L,sensor}$) and the latency of the sensor with the least worst-case latencies min($T_{L,sensor}$+$T_{C,sensor}$):

$$T_{L,max} = \max\left(\max(T_{L,sensor}), \min(T_{L,sensor} + T_{C,sensor})\right) \tag{5}$$

COMPARISON

Three different measurement synchronization strategies were presented. For time critical ADAS the worst-case latency introduced by a sensor fusion system is of major importance. As shown the overall worst-case latency of the environment description which is propagated to the application strongly depends on the synchronization strategy.

However the best strategy depends on the performance of each sensor in the multi-sensor set. Table 4 summarizes two extreme sensor scenarios. In the first configuration there is one sensor in the set which has a very high measurement frequency and at least one sensor with half of that measurement frequency. In this case the synchronous sensor approach is significantly preferable as it reduces the latency introduced by the measurement synchronization by approximately 50% with respect to the non-deterministic case. The deterministic approach performs slightly worse because of its additional minimal worst-case sensor latency.

	$T_{L,max}$, one slow sensor	$T_{L,max}$, equally fast sensors
Non-deterministic	$\max(T_{L,sensor}) + \max(T_{C,sensor})$	$T_{L,sensor} + T_{C,sensor}$
Deterministic	$\max(T_{L,sensor}) + \min(T_{C,sensor})$	$T_{L,sensor} + T_{C,sensor}$
Synchronous sensors	$\max(T_{L,sensor})$	$T_{L,sensor} + T_{C,sensor}$

Table 4: Comparison of different synchronization strategies for two cases: a) there is one sensor with a high measurement latency and at least one sensor with half of that measurement latency; b) all sensors work at equal measurement latencies.

In the other extreme sensor configuration case all sensors are considered to have an equal worst-case measurement latency and frequency. As shown in Table 4 all approaches exhibit the same latency introduced by the measurement synchronization. In this case the non-deterministic approach is probably preferable as it imposes the least constraint on the sensors. Additionally the asynchronous approaches exhibit a smoother object trajectory as measurement are integrated at different times. In the synchronous configuration all sensors deliver measurements at the same periodic times.

CONCLUSIONS

Different synchronization strategies for sensor data fusion are discussed and their features compared. Their usability is contrasted with respect to the application in future advanced driver assistance systems.

REFERENCES

[1] Kay Fuerstenberg, Pierre Baraud, Gabriella Caporaletti, Silvia Citelli, Zafrir Eitan, Ulrich Lages, and Christoph Lavergne, “Development of a Pre-Chrash sensorial system — the CHAMELEON Project,” in *VDI Berichte 1653: Fahrzeugkonzepte f ¨ur das 2. Jahrhundert Automobiltechnik 2001*, Wolfsburg, Germany, 2001, pp. 289–310.

[2] Dr. Jochen Langheim, A.J. Buchanan, Ulrich Lages, and M.Wahl, “CARSENSE—New environment sensing for advanced driver assistance systems,” in *Proceedings of the IEEE Intelligent Vehicle Symposium 2001*, Tokyo, Japan, 2001, pp. 89–94.

[3] C. Stiller, J. Hipp, C. Rössig, and A. Ewald, “Multisensor obstacle detection and tracking,” *Image and Vision Computing*, vol. 18, pp. 389–396, 2000.

[4] Domninique Gruyer, Cyril Royere, and Veronique Berge-Cherfaoui, “Credibilist multi-sensor fusion for the mapping of dynamic environment,” in *Fusion 2000*, Paris, France, July 2000.

[5] A. Vukotich and Alexander Kirchner, “Sensor fusion for driver-assistance-systems,” in *Elektronik im Kraftfahrzeug*, Baden-Baden, Germany, 2001.

[6] K.C.J. Dietmayer, A. Kirchner and N. Kaempchen, “Fusionsarchitekturen zur Umfeldwahrnehmung für zukünftige Fahrerassistenzsysteme,“ in *Fahrerassistenzsysteme*, Springer Verlag, Germany, 2003, accepted

[7] K.Ch. Fuerstenberg, K.C.J. Dietmayer, S. Eisenlauer, and V. Willhoeft, “Multilayer Laserscanner for robust Object Tracking and Classification in Urban Traffic Scenes,” in *Proceedings of ITS 2002, 9th World Congress on Intelligent Transportation Systems,* Chicago, USA, 2002.