Multisensor Data Fusion Strategies for Advanced Driver Assistance Systems

Mahdi Rezaei Ghahroudi[1] and Reza Sabzevari[2]
[1]Department of Computer Engineering, Islamic Azad University of Qazvin
[2]Member of Young Researchers' Club (YRC)
I.R. Iran

1. Introduction

Multisensor data fusion and integration is a rapidly evolving research area that requires interdisciplinary knowledge in control theory, signal processing, artificial intelligence, probability and statistics, etc. Multisensor data fusion refers to the synergistic combination of sensory data from multiple sensors and related information to provide more reliable and accurate information than could be achieved using a single, independent sensor (Luo et al., 2007). Actually Multisensor data fusion is a multilevel, multifaceted process dealing with automatic detection, association, correlation, estimation, and combination of data from single and multiple information sources. The results of data fusion process help users make decisions in complicated scenarios. Integration of multiple sensor data was originally needed for military applications in ocean surveillance, air-to air and surface-to-air defence, or battlefield intelligence. More recently, multisensor data fusion has also included the non-military fields of remote environmental sensing, medical diagnosis, automated monitoring of equipment, robotics, and automotive systems (Macci et al., 2008).

The potential advantages of multisensor fusion and integration are redundancy, complementarity, timeliness, and cost of the information. The integration or fusion of redundant information can reduce overall uncertainty and thus serve to increase the accuracy with which the features are perceived by the system. Multiple sensors providing redundant information can also serve to increase reliability in the case of sensor error or failure. Complementary information from multiple sensors allows features in the environment to be perceived that are impossible to perceive using just the information from each individual sensor operating separately. (Luo et al., 2007)

Besides, driving as one of our daily activities is a complex task involving a great amount of interaction between driver and vehicle. Drivers regularly share their attention among operating the vehicle, monitoring traffic and nearby obstacles, and performing secondary tasks such as conversing, adjusting comfort settings (e.g. temperature, radio.) The complexity of the task and uncertainty of the driving environment make driving a very dangerous task, as according to a study in the European member states, there are more than 1,200,000 traffic accidents a year with over 40,000 fatalities. This fact points up the growing demand for automotive safety systems, which aim for a significant contribution to the overall road safety (Tatschke et al., 2006). Therefore, recently, there are an increased number of research activities focusing on the Driver Assistance System (DAS) development in order

to reduce the driver's workload and prevent driving accidents and several types of safety systems have therefore been proposed to help lessen the danger and assist the driver (Hsieh et al., 2007). Current technology field of the automotive industry focuses on the development of active safety applications and advanced driver assistant systems (ADAS) instead of passive safety systems. Passive safety systems such as seatbelts and airbags provide protection in the case of collision; more recently however, active safety systems have been introduced to help the driver avoid collisions in the first place. Nowadays, systems such as lane departure warning and rear-end collision avoidance have been introduced (Amditis et al., 2006); (Mammar et al., 2006). These active safety systems are required to interact much more with the driver than passive safety systems, creating a closed loop between driver, vehicle, and the environment. Examples of such systems could be found in the Laboratory for Intelligent and Safe Automobiles (LISA) (Trivedi et al., 2007). An ADAS shall support the driver in his/ her task to drive the vehicle providing additional information or warning when encountering any dangerous situation. The system should be equipped with various types of sensors to observer the environment around the vehicle. For example, radar and laser scanners are sensors used to measure distance and velocity of objects, and video cameras are used to detect the road surface and lane markings or to provide additional visual information (Leohold, 2004). This aspire a reduction or at least an alleviation of traffic accidents by the means of collision mitigation procedures, lane departure warning, lateral control, safe speed and safe following measures (see figure 1).
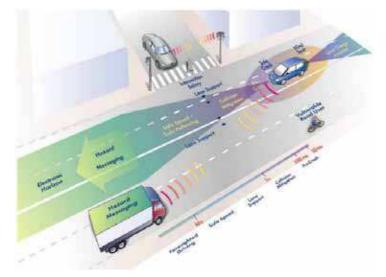

Fig. 1. Safety zone around the vehicle by implementation of multi sensory (PreVENT, 2006)

This chapter describes data fusion concepts, an applicable model, paradigm of multisensor fusion algorithms, current sensor technologies and some applications such as object tracking, identification and classification and a providence view on next-generation car safety and driver assistance systems. The later applications are particularly suitable to provide an overview of multisensor data fusion starting from the plain detection of multiple objects around a given host vehicle to inferring: the position and speed of possible obstacles, the type of objects in the road environment, the relative movement, and distribution of

obstacles over a given area, the early detection of a possible collision, possible suggestions for prompt and effective countermeasures (e.g., sudden braking, steering wheel adjustment, etc.). In this chapter, a new framework of active ADAS is proposed. The framework implements a data-fusion architecture of multiple sensors, including several key functionalities in ADAS, and related applications for different driving conditions and vehicle's surround as well as drivers state monitoring to detect the amount of driver's fatigue and his/ her inattention. Our proposed ADAS consists of functionally isolated sensors to observe the environment around the vehicle and the vehicle inside situation. Commonly used sensors in an ADAS are included in the framework and the multi-sensor selection is formulated as an optimal programming problem and implemented in a two relevant simulator. The Ultimate goal of proposed ADAS is to create safety zones around vehicles by the means of embedded multisensor data fusion systems that sense environment, so detects the type and significance of impending dangers. Depending on the nature of the threat, an active and preventive safety system will inform, warn, and actively assist the driver to avoid an accident or mitigate its possible consequences.

2. Requirements of advance driver assistance systems

Many existing robotics technologies apply to intelligent assistance driving; however, much research works neglect the preview of the driver and driver response delay. Moreover, the behavior of high-speed vehicles differs greatly from other robots. To obtain safe driving, a driver should be in the center of the "safety analysis" not on the center of driving process; that is one of the key differences of our method than other researches. Driver's response delay, together with other factors, restricts the driving path of a vehicle. In proposed model not only we use multiple and optimal sensor fusion technique but also apply the driver's behavior and degree of his/ her alertness in overall control of the host vehicle in a real time manner through a supervisor unit (Rezaei Ghahroudi & Fasih, 2007). The supervisor unit is a decision unit that may sometimes directly interpose the actuators to control the vehicle in dangerous situations.

The other key difference in this method, unlike other researches that focused in one particular application, is that we believe in order to control the vehicle similar to an expert and sober driver, the ADAS systems should not work independently. Instead, after sensor data fusion and decision making, it should run several driving assistance system simultaneously for reliable and robust control of the vehicle. It means in a real word, several actuators should take an appropriate action simultaneously, while most of current DAS systems on the market just carry out one single system at a time; for example Automatic Cruise Control (ACC) or Park assistance each of them act individually. For better understanding, imagine the following example. Encountering an unexpected obstacle in the road will cause heavy braking by all of the vehicles in all of the lanes; this may bring about deviation of other vehicles in an unpredictable way. In such situations, in addition to decreasing speed, the system should correct its direction (steer angle) to mitigate from accident or lessen the scathes and in ideal state, preventing incidence (Figure 2).

In figure 2 there are four successive steps come one by one. At First because of e.g. falling a tree or collapse of hillside (unpredictable obstacle) on the road, the entire lines blockaded and consequently all the vehicles should stop immediately. Suddenly ACC system of the host vehicle (Green car) detects that the foreside vehicle (blue car) has stopped (step1). Then emergency braking system starts to decrease the speed of the vehicle (step2). During step

two, side ultrasonic sensors detect a car on the left side that is coming close and close (step 3), so the supervisor unit decides to intervene and correct the steer angle to escape from the left car (silver car) and correct its path (step 4).
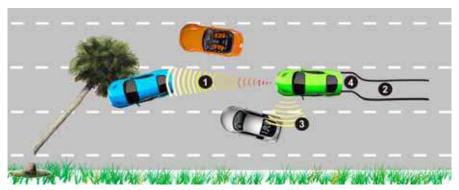

Fig. 2. Schematic of a typical accident in real world

It means, in such cases it needs to act simultaneously several driver assistance systems such as emergency braking system, steer angle correction, lateral collision warning system, front monitoring and keep in lane monitoring. Now let's discuss the design and development of a multi sensor selection in order to cover most of essential vital driver assistance systems in a dynamic scenario. In order to cover different driving conditions, a set of sensor should be established for identifying the driving environment.

3. Optimal sensor selection and multi sensor assembly

In a typical scenario, a combination of vehicles in different position of the road are facing various driving conditions such as driving straight, turning, overtaking a vehicle, meeting pedestrians, etc. For these different conditions, the ADAS equipped vehicle need different sets of sensors to detect environment-related information and determine a correct driving condition. The decision on selecting a proper set of object-detecting sensors should be made based on the capability of available sensors and real-time driving condition. Now, In order to formulate and simulate the selection of object-detecting sensors with respect to various driving situations, the sensors should be capable of evaluating *Driver commands* (steer angle setting, backing, changing lane, turning a corner and overtaking a vehicle), *Relative Vehicle's Velocity*, *Traffic Flow* (Low or Dense), and *Driver's behavior* (Observant, sleepy, drowsy, aggressive, using cell phone, etc.), (Hsieh et al., 2007).

The combination of these parameters will be used to reflect a proper diving situation encountered by the driver. So we need an optimal selection of some appropriate sensors to monitor all these four factors (Kaempchen and Dietmayer, 2003). But, which sensor is better and optimal? Image sensors have some drawbacks, such as low ability of sensing depth and advantage of higher ability of discrimination than LIDAR and RADAR. Radar shows limited lateral spatial information because it is not available at all, the field of view is narrow, or the resolution is reduced at large distances. Although LIDAR has a wide view field that solves part of the previous problems, there are other problems such as low ability of discrimination, clustering error, and recognition latency. These restrictions of the different sensor types explain the attention given to sensor fusion in research on object detection and

tracking (Cheng et al., 2007). According to advantages and drawbacks of mentioned sensors and various real driving situations, we implement an optimal selection of state of the art and most typical sensors that can be classified into four different types: Radars (R_L, R_S), Laser scanner (L), Camera Sensors (C) and Ultrasonic Sensor (U). In this project, 16 object-detecting sensors among four types of sensors are considered to be assembled on the test vehicle (Figure 3)
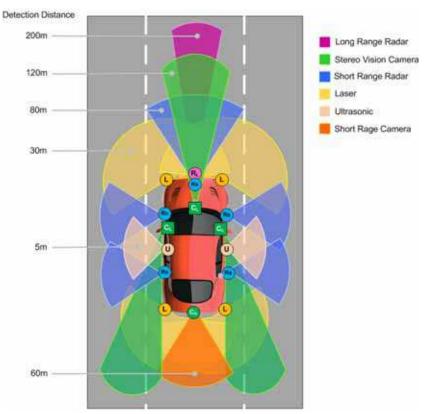

Fig. 3. Sensor placement with full coverage, redundancy and sufficient overlapping area

The sensor network assembly consists of one Long Range Radar-LRR (2nd generation long range radar by Bosch) mounted for front monitoring *(R_L)*, five Short Range Radar sensors (from M/ A-COM / Tyco Electronics) four of them in both sides and one in the front *(R_S)*. Four laser scanners (from IBEO GmbH) with broad range and wide viewing angle, two of them in front and the others for rear side *(L)*, three CMOS cameras INKANSC640PG by Aglaia GmbH (Amditis et al., 2006) two of them in side mirrors for rear and blind spot coverage and one in the middle of the front windscreen to face forward **(C_L); A** short range monocular camera for backside *(C_S)*, and finally two ultrasonic on both sides *(U)*. The placement of these object-detecting sensors and the main characteristics, such as detecting distance and the field of view, for these object-detection sensors can be seen in figure 3. All the sensors transmit data through a CAN-interface (Wahl & Doree, 2000) to the perception layer.

The placement of these 16 object-detection sensors are based on the following six main functionalities required in a state of the art ADAS. The six key functionalities are adaptive cruise control (ACC), lane departure warning (LDW), lane change assistant (LCA), rear view (RV), lane keeping assistance (LKA) and Emergency Braking System (EBS).
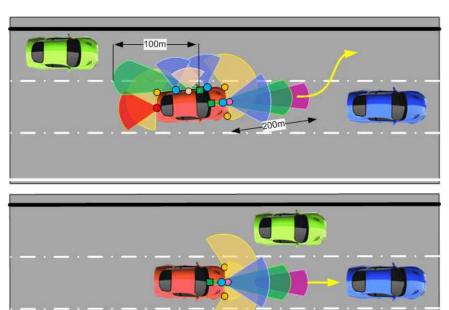

Fig. 4. Active sensors for lane change and overtaking (Top) and vehicle following with safe distance and safe speed according to ACC set point (Bottom).

These functionalities are required to match seven scenarios of straight, back, follow a vehicle, overtake a vehicle, lane change, turn corner, and emergency stop.

- Driving straight is the forward driving on a lane without leaving the lane.
- Driving back is the backward driving without leaving the lane.
- Following a vehicle is the driving accompanied a vehicle which driving in the front and keeping a proper relative distance and velocity.
- Overtaking a vehicle is cutting in a road by overtaking a vehicle which is driving in the front and there are left overtaking and right overtaking when driving on a road.
- Lane changing is the straight driving but changing the lane to another lane. Turning corner is to make a turn when driving on a road including intersection and forked road and there are left turning corner and right turning corner.
- Emergency stop is reducing the speed with a logical deceleration in order to eliminate an unexpected obstacle.

Figure 4 shows two sample scenario and active sensors for *Overtaking* (Top) and *ACC and vehicle following* (Bottom) schemes.

4. Driver assistance framework based on multi sensor data fusion

After implementing sensor assembly, a practical framework needs to be designed. We assume a driver assistance system as a vehicle–driver–environment interactive closed-loop system; moreover, we focus on not only the current situation of vehicle and driver, but also the future situation by predicting the potential collision probability distribution. In this framework, onboard sensors provide real-time information about drivers, traffic environment, and vehicles. How to configure these sensors is closely related to the application domain. For example, for multisensor ACC systems, radar and camera often suffice, but for pedestrian protection systems, an infrared sensor is essential to robust detection under various weather conditions.
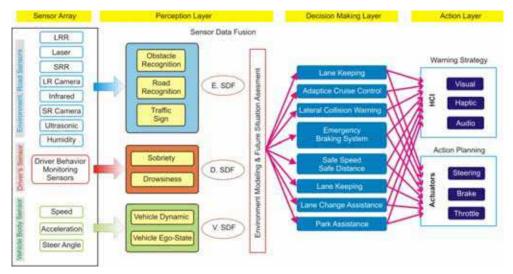

Fig. 5. Overall Architecture of Driver Assistance System based of sensor data fusion

As shown in figure 5 on the basis of various factors, an integrated interactive road safety analysis framework is introduced, where the system consists of the following modules:

1- Sensor Layer: including onboard environment monitoring sensor, inside driver monitoring sensor and vehicle body sensor e.g. vehicle ego-state and vehicle dynamic module. In General, the external sensors capture object appearance, range, and sound outside a vehicle, a vision based system monitor and record driver's behavior, and the interior sensors collect vehicle state, such as speed, acceleration, and steering angle. Figure 6 and 7 describe characteristics of two main typical sensors required for this system, LASER and RADAR respectively (Schneider, 2005), (Langheim et al., 2001)

The main technical characteristics of a typical Long Range Radar sensor are:
- Frequency: 76-77 GHz
- Range: 1-150 m
- Search Area: 12°
- Speed measurement precision: < 0.2 km/ h
- Angular Precision: < 0.3°

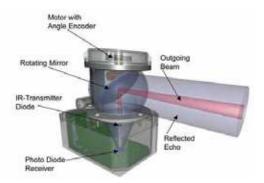Fig. 6. Bosch 2nd Generation Long Range Radar Sensor Unit

Fig. 7. ALASCA XT, the last Laserscanner from IBEO Automobile Sensor GmbH

The main characteristics of the current laser sensor are:
- Scan Frequency: 12.5 Hz or 25 Hz
- Horizontal Viewing angle: up to 240° field of view
- Effective Distance range (at reflectivity 5 %): up to 40 m
- Standard deviation of measured distance: +/ - 5 cm
- Eye-safe laser (laser class 1)

2- Perception Layer: this layer aims to give a realistic representation of the environment to the applications; it can be seen as an intermediate layer between the sensorial sub-system and the system applications. It includes sensor fusion of previous layer in order to environment modeling, overall sensor fusion, and future situation assessment.

The role of the perception layer is to:
- Carry out a perception enhancement of the external scenario, even independently on any given application
- Describe the environment and the traffic scenario (obstacles and host vehicle dynamics, road geometry, etc.) in a formal way.
- Support the specific needs of the ADAS system in the reconstruction of the environment
- Act as an intermediate layer between sensors and functions, defined by the I/O protocols and interfaces.

The main functions of environment modeling and sensor fusion are to sense and recognize obstacles, lanes, pedestrians and traffic signs, etc.; where sensor information of different kinds is fused to model the environment. For the future situation assessment module, future road safety situations are assessed by combining traffic rules, vehicle dynamics, and environment prediction. Since the safety distance varies with the speed of a host vehicle, we adopt preview time rather than safety distance as the measurement of safety response. The safety response time is given as (Cheng et al., 2007):

$$T_0 = \frac{d_r + d_v + d_s}{v} \tag{2}$$

Where d_r is the distance required to respond to the nearest object due to driver response delay, d_v is the distance to slow down, d_s is the safety distance between the host vehicle and obstacles, and v is the speed of the host vehicle.

3- Decision making agents layer: as its names implies, perform decision carrying out one or several various driver assistant depending the status of previous layers. For example to make decision about "safe distance" in following a vehicle , it refers to the decision about determining a distance in meters that needs to be maintained to a vehicle ahead in order to avoid a collision in case the vehicle suddenly brakes. Safe-distance decision depends on a number of factors such as vehicle type, reaction time, braking distance, road surface and traffic conditions, driver awareness, etc. Over the years, different criteria have been suggested to determine a safe following distance for driving. Often cited is the two seconds rule illustrated in Figure 8 (Irion et al, 2006).

Decision-making agents have two functions: first to generate warning strategies for warning systems, such as route guide systems, haptic alert and warning display devices, and the secondly to plan the actions of the actuators to control the path of the host vehicle. As we know, the parameters of a vehicle model are affected by many factors and vary with the outside environment. Therefore, real-time learning algorithms using fuzzy logic have been applied to solving lateral and longitudinal control problems for a long time and have shown good performance (Cheng et al., 2007).

Fig. 8. Safe Distance decision by applying two seconds rule

4-Action Layer: Including two primary sections HMI and Actuators. This layer is responsible for handling the output from all the different applications in order to carry out appropriate warning and intervention strategies and then, to pass the information, suggestions or alarms to the driver through a given HMI. An in-vehicle HCI module

presents visual, sound and haptic warning information from the decision module and the path planning module to the driver. On the other hand action planning affects on actuators to prevent collisions and mitigate the dangerous. These are our final objective in order to approach to safe driving.

5. Sensor fusion processing

In this section we focus on data fusion and some common terminologies in literatures (Strobel & Coue, 2004) that take place in the perception layer. In general, sensor data fusion means processing steps which take the sensor data of several single sensors and combine the information from these sensors to a common and a certainly better result comparing with the outcome of each single sensor's processing could provide. This effect of getting better or more adequate results out of sensor data fusion compared to single sensor processing is called 'synergy' or 'synergetic effect'. Figure 9 shows the process of 'multi sensor processing', starting with the sensor data acquisition. Next, the sensors processing, divided into several tasks, as 'Calibration, 'Feature Extraction', 'Object Detection', etc., begins to analyze the sensors data and, in the end, serves the application with a more or less detailed model of the environment.

Before the sensors acquisition task is taking place, the sensors should be calibrated one by one in both time and space. That means, opposed to single sensor, a multi sensor system has to be synchronized, or the data acquired has to be time-aligned. Next, the sensors have to be 'space-aligned', meaning that displacements (in a mathematical sense) between the different (spatial) sensor coordinate systems have to be determined. Figure 9 gives a non-exhaustive exemplary overview on the tasks necessary for sensor data fusion processing.

In multi sensory, there are sequences for following the 'level of abstraction'. The calibration, time-alignment and reconstruction tasks, for example, take place at low level, while the 'objects detection' and the 'obstacles classification' are quite high level tasks within the processing. In some points of view, these classifications called early and late sensor data fusion, as well.
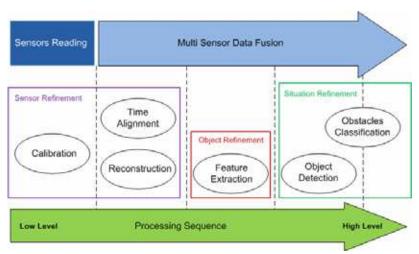

Fig. 9. Sensor Fusion Processing

From now, interest is focused on to sensor data fusion taking place exclusively within the perception layer. Generally, current trends in sensor data fusion inside the perception layer are classified in three sections of "Sensor refinement", "Object Refinement" and "Situation Refinement" (Strobel & Coue, 2004).

5.1 Sensor refinement

Sensor Refinement itself covers three tasks of "Calibration"," Time Alignment" and "Reconstruction of Sensor Models". All of them try to approach best observations from sensors. On the whole, the sensor refinement consists of the lowest, respectively the earliest processing steps. Hence, it is builds the interface between sensor and sensor processing system, too.

Generally speaking, because of its natural complexity, the low level processing tasks have not been inspected as detailed as processing steps which are taking place within the object and situation refinement. Actually compared to the object and the situation refinement level, these low level tasks have to cope with a huge amount of raw sensor data. In addition, due to the quite different sensor characteristics, imaging, ranging, etc. the raw sensor data may appear completely different as well.

In principle, the problem is obvious; in recent approaches most of the tasks taking place within the sensor refinement has been examined and to some extent even solved for single sensor processing in a rather long term period. But then, the same results have been applied for multi sensor systems. So, even in a multi sensor system each sensor is treated as a single sensor. On the other hand, these tasks should be easier to cope with and could be solved more accurate while more information has been considered during processing. So, on the low level, we have got all the information needed to tackle them in this sense. In the following, we discuss sub-sections of sensor refinement block with some well chosen examples of interest that are employed to present this fact.

5.1.1 Calibration and time alignment

Calibration refers to the process of determining the relation between the output of measuring instrument (here multi sensors) and the value of the input quantity or attribute according to a standar measurement. The problem with time-alignment is something different. In general, multi sensor systems are not synchronized, i.e. each sensor aquises its measuring at a different and specific time. Actually it would be difficult or useless to establish a multi sensor processing and the perception of the environment based on measurements from different sensors taken at different times; because they probably show contradictory states of the environment. Hence, a multi sensor system has to be time aligned. Within the 'SAVE-U' project (www.cordis.europa.eu) (Marchal et al., 2003), for example, the calibration of a multi sensor system consisting of a coloured camera (resolution: 640 x 480 pixels) and a far infrared sensor - at a resolution of 320 x 240 pixels - has been examined and implemented. Based on a manual selection of 20 corresponding points in two images, one from the coloured camera and another from the far infrared sensor, showing the same scenario the coordinate systems of both, the camera and the far infrared sensor are aligned; That means, a transformation which maps the coordinates of the camera to the coordinate system of the far infrared sensor, respectively vice versa, is derived from these corresponding points. On the calibration problem for single sensors and single sensor systems, e.g. for monocular stereo cameras and panoramic cameras, some efforts

have been taken already (Li et al, 2003) (Huang et al., 2006). Especially for stereo cameras even automatic calibration methods have already been proposed. But these existing methods still provide no reasonable solutions for the automotive sector.

5.1.2 Sensor modeling

Another group of problems to cope with on the sensor refinement level concerns the sensor models and a common representation and description of raw data. In automotive applications, the term 'sensor model' is understood as a model of sensor's data and how the sensor information matches the environment, including both model of the sensor's data and model of the vehicle's environment. This is mostly done individually for some types of sensors, e.g. for RADAR sensor probabilistic models are widely accepted, but not for a multi sensor system 'as a whole'. Until now, no common model was found which is able to represent all the different types of sensor data emerging in the field of preventive safety applications on these low levels e.g. radar, laser, images, etc.

5.2 Object refinement

The objective of the object refinement level is to interpret the multiple sensor 'observations', subsets of the sensors' raw and slightly preprocessed data, as individual objects or as parts of the environment observed (e. g. pedestrians, other vehicles, buildings, guard rails, trees, etc.). The input of this level consists of the output of the sensor refinement, called observations, which should be interpreted correctly. As Figure 10 illustrates 'observations' coming from several sensors (illustrated as ellipses in different colours) - represented in a common model as claimed above - are interpreted as objects (for example other vehicles or pedestrians) during the object refinement.
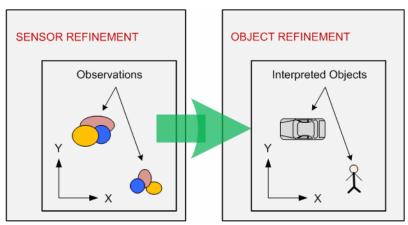

Fig. 10. Correlation between input-output of sensor refinement and object refinement

Hence, the output of the object refinement level generally consists of a list of objects with specific parameters or attributes (e. g. their position, their velocity, their height …). The key functions of 'object refinement' are, among others, considered to be feature extraction, (single or multi) object tracking, 'observation-to-track' association and track maintenance (in the case of multi object tracking), as well as object classification (Strobel & Coue, 2004).

5.2.1 Feature extraction

Generally speaking, feature extraction could be regarded as the process of deducing 'features' or eye-catching properties, like segments of sensor data corresponding to the same object in the environment for example, from the sensor data. Concepts or techniques which are basically used for feature extraction could be segmentation, contour extraction, etc.

The following example shows feature extraction via segmentation in LIDAR sensor data:

In order to segmentation of the environment, first we have to perform *clustering*. The readings, e.g. from a LIDAR are subdivided into sets of neighbor points (Figure 11, left) by taking the proximity between each two consecutive points of the scan into account. A cluster is hence, a set of measures (points of the scan) close enough to each other, which due to their proximity; probably belong to the same object (Mendes et al., 2004). The segmentation criterion is based on two consecutive measurements of laser emitter, r_k, r_{k+1}; we can conclude that they belong to the same segment if the distance between them fulfils the following expression:

$$R_{k,k+1} \leq L_0 + R_{min} \cdot \frac{\tan\alpha . \sqrt{2(1-\cos\theta)}}{\cos\left(\dfrac{\theta}{2}\right) - \sin\left(\dfrac{\theta}{2}\right)} \tag{3}$$

Where $R_{min} = \min \{R_k, R_{k+1}\}$, $R_{k+k+1} = |R_k - R_{k+1}|$ and θ is angular resolution of Laser sensor. α was introduced to reduce the dependency of segmentation with respect to the distance between the Laser Emitter and the object, and L_0 to handle the longitudinal error of the sensor. If $L_0 = 0$, then α represents the maximum absolute inclination that an object's face can have to be detected as a unique segment (Figure 11*)*. Continuing the line fitting (Mendes et al., 2004) and considering the whole near lines as a single object we would be able to determine the object's type, comparing with a model-based database. In that step using a technique called "Identity estimation", relies on special classification algorithms that are used to recognize an object on the basis of some significant extracted features-the shape or
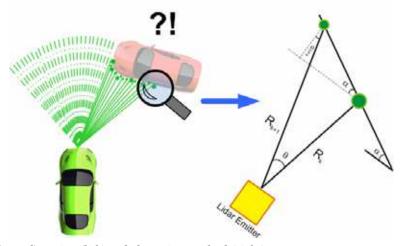

Fig. 11. Laser Scanning (left) and clustering method (right)

patterns- of various vehicles detected on a road, for example. In car safety applications, these kinds of algorithms can also be employed to construct a list of the objects (e.g., a tree, a motorcycle, a road sign, etc.) surrounding the host vehicle.

Figure 12 (left side) shows an example of observations, namely raw data (a list of points) coming from a scanning LIDAR at an angular resolution of 0.5 ° from a parking.
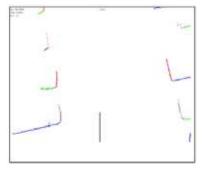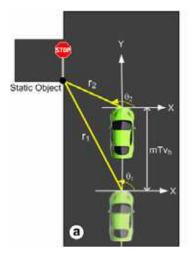

Fig. 12. LIDAR raw data (left side) and extracted features on the right side (provided by INRIA Rhone-Alpes)

In Figure 12 (on the right) we see the result from the first step of this feature extraction: the segmentation of raw LIDAR data in a set of segments (all marked in different colours). This is done using 'Split and Merge' (SaM) techniques as proposed in (Lee, 2001). During the 'splitting' step, the set of observations is recursively subdivided and a new segment is associated to each subdivision. Then, during the 'merging' step, segments lying closed to each other are merged.

Now, we have to decide which segments belong to the same object, a vehicle or a building close to the parking, and to estimate the position as well as the orientation of the object. This could be done using the Hough transform (Duda & Hart, 1972), (Tardos et al., 2002) under the hypothesis that the object could be represented as a rectangle. Then, a probability histogram on the object position and its orientation is constructed using the Hough transform, each mode of this probability distribution corresponding to one object (a vehicle or a building). But, as the vague term 'feature' already indicates, for feature extraction numerous procedures exist, depending only on the concrete definition of the term 'feature': features could be segments of corresponding sensor data, contours, prominent points, etc, for example. And at the end, as much as the meaning or the definition of the term 'feature' varies throughout the applications new or slightly adapted methods have been proposed to perform the feature extraction and to tackle the newly stated problem in these cases. In addition to object detection, we have to determine the absolute velocity of the moving object around the vehicle.

Normally the vehicle speed is measured by an encoder coupled with axle. But in sensor based system we can perform a vehicle speed estimation using two observations from a static object during a period of T. If we assume (r_1, θ_1) and (r_2, θ_2) as two observations like figure 13 left, then v_h, speed of host vehicle is:

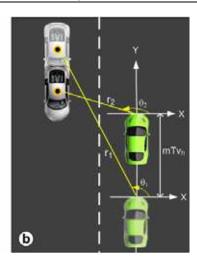$$v_h = \frac{r_1^2 + r_2^2 - 2r_1r_2 Cos(\theta_1 - \theta_2)}{mT} \tag{4}$$

Fig. 13. Speed estimation approach

Where m is the number of the consecutive frames and is generally ≥ 2 for improving the accuracy of velocity. Here, we assumed that over a small interval of time mT, the driving direction of the host vehicle is consistent with the *Y*-axis in the Cartesian coordinates system *XOY*. Now after determining the speed of vehicle host, we can define the coordinate system of two segments at position P_0=(x_0, y_0) and P_1=(x_1, y_1), while:

$$x_0 = -r_1 \cos \theta_1 \qquad , \qquad y_0 = r_1 \sin \theta_1 \tag{5}$$

$$x_1 = -r_2 \cos \theta_2 \qquad , \qquad y_1 = r_2 m T v_h + r_2 \sin \theta_2 \tag{6}$$

5.2.2 Sequential estimation and filtering problems

Estimation algorithms generally return the values of some quantitative entity parameters or attributes that are particularly significant for the application considered (R. Goodman et al., 1997). For instance, in car safety and driver assistance systems, estimations could be made for kinematic parameters (e.g., the position and the relative velocity) of the objects observed outside the host vehicle. To improve the estimation of the objects' attributes, or the objects' state (from objects detected around the vehicle), it is helpful to use several time-discrete estimations of the same objects determined at different times. The best would be if we could model the evolution of the objects' attributes in time. This is basically a filtering problem - called the 'sequential estimation problem'. One of the most famous filter used in the automotive sector to cope with these problems is called 'Kalman filter' (see (Kalman, 1960) and (Welch & Bishop, 2001) for details) and its extended version, the 'extended Kalman filter'. This probabilistic filter is based on a model of the evolution of the object's attributes in time, called the dynamic model, and on a sensor model (Strobel & Coue, 2004). Here, the concept is to determine the estimation of the objects' state at a given time k+1 in two stages: a prediction stage uses the dynamic model and the estimation of the state at time k to determine a prediction of the state of the object at time k+1. Then an update stage confronts

this prediction with the new sensor observations at time k+1 and updates the objects' attributes, accordingly.
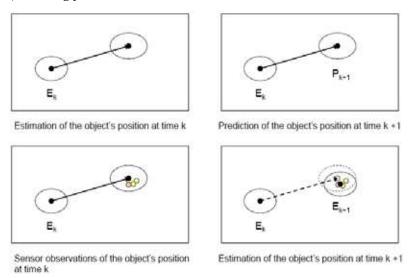

Fig. 14. An Example of Prediction scheme used in the 'Kalman filter'

Figure 14 shows the general prediction scheme of the 'Kalman filter' in the case of a two dimensional problem. Indeed both, the 'Kalman' and the 'extended Kalman filter' are simply efficient implementations of the general 'Bayesian filter' (compare (A.H. Jazwinski, 1970), for example)) - with strong constraints to linearity of the dynamic model and 'Gaussian noise' in the case of 'Kalman filter'. But more recent implementations of the 'Bayesian filter', namely the so called 'particle filters' (compare (A. Doucet et al., 2000), (S. Arulampalam et al., 2002)) become more and more popular in these fields, as they allow an efficient approximation of at least all 'Bayesian' filtering problems. On the other hand, 'Bayesian filters' are not the only technique used to come up with the 'sequential estimation problem'. For example, solutions based on the fuzzy logic are proposed as well (see Ggruyer, 1999)), but they use the same prediction scheme (see Figure 14).

6. Fuzzy fusion methodology

Here we propose a fuzzy logic approach as sensor fusion, just for ACC driver assistance system which is applicable on other driver assistance system in a similar manner. Then a new filtering method is performed -different from typical Kalman and Bayesian approaches- to reach more desirable results. The fuzzy logic system is carried out by *Fuzzy*TECH simulator and the filtering has performed by MATLAB R2007b.

As you see in figure 4 (Bottom), to follow a vehicle by an ACC system we must keep safe distance by measuring the front vehicle distance to the host vehicle. On the other hand, four types and five sensors among total 16 sensors should be considered, each of which with different coverage area and may be infected by some environments noise, consequently with deferent measurements regarding the position of front vehicle. In such situation, we fuse all sensor data by fuzzifying them so determine a near real distance (Figure 15).
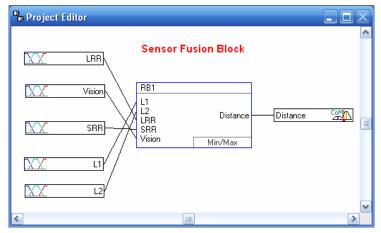
Fig. 15. Sensor fusion scheme using fuzzy logic method

Structure of fuzzy logic system:

The initial system structure is defined by input and output variables with their linguistic terms. Linguistic variables are components of fuzzy logic systems that "transform" real, crisp values, here from sensors, into linguistic values. The output also defined by some linguistic variables but finally should be defuzzified in to real output value (Figure 16).
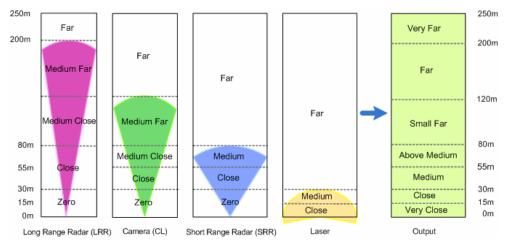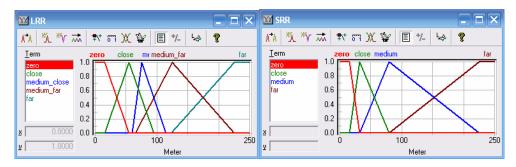

Fig. 16. Linguistic variable definition for sensors (inputs) and fused data (output)

In addition to the linguistic variables, a preliminary rule set must also be created for the system prototype. In this system, each of sensor measurements are matched to some membership functions and then a distance rule block is defined as output (Figure 17). Here are two sample rules according to figure 16:

- IF (LRR=Medium Far AND Vision, SRR, L1, L2 = Far) THEN Distance = Far
- IF (LRR=Close AND Vision=Medium Close AND SRR=Medium AND L1, L2=Far) THEN Distance = Above Medium
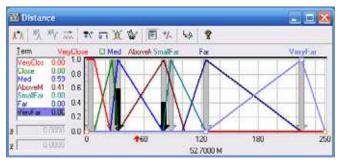
Fig. 17. Some of Membership functions for sensors (LRR, SRR) and output of sensor fusion (Distance)

Then a sample input data entered as the distance of the front vehicle with various speed and acceleration to examine the robustness of the system. The results was very interesting, after several modification and improvement of membership functions with Min-Max Aggregation operator of *Fuzzy*TECH Simulator, finally a satisfactory following by the host vehicle is extracted. Let's have a brief description on figure 18. In this graph, Blue curve is the real position of front vehicle with different acceleration rate at different times. The Red Curve is estimated distance according to multisensor data fusion using fuzzy approach. As can be seen from figure 18 in the area with multi detecting sensors, (e.g. in distances < 50m) we saw more fluctuations, But in far distance (distance >100m) we saw better following, even with just a single LRR sensor! The reason is very clear; because the nature of different multi sensors (In lower distances), they feed the system a little bit different measurements and data, which will cause some fluctuations in overall distance detection. But in general, this is more reliable than a single sensor in far distance, despite a little fluctuation. The worst deviation found in this stage was about ± 5.26 meters.

7. Moving average filter

In Continuation, it is tried to improve red (fusion) curve by keeping both reliability of multi sensory in lower distance and reducing the fluctuations in multisensor areas. In this stage a filtering method is applied as it follows. A slight improvement in computational efficiency can be achieved if we perform the calculation of the mean in a *recursive* fashion. To illustrate this, consider the following development: Suppose that at any instant k, the average of the latest n samples of a data sequence, x_i, is given by:
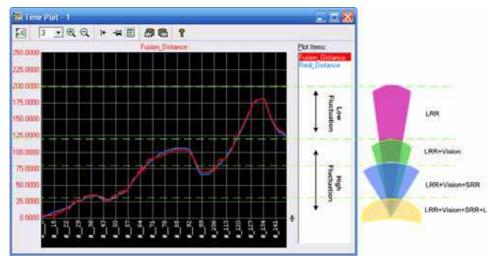
Fig. 18. Host vehicle following based on sensor data fusion

$$\overline{x}_k = \frac{1}{n} \sum_{i=k-n+1}^{k} x_i \tag{7}$$

Similarly, at the previous time instant, *k-1*, the average of the latest *n* samples is:

$$\overline{x}_{k-1} = \frac{1}{n} \sum_{i=k-n}^{k-1} x_i \tag{8}$$

$$\text{Therefore,} \quad \overline{x}_k - \overline{x}_{k-1} = \frac{1}{n}\left[\sum_{i=k-n+1}^{k} x_i - \sum_{i=k-n}^{k-1} x_i \right] = \frac{1}{n}\left[x_k - x_{k-n} \right] \tag{9}$$

which on rearrangement gives:

$$\overline{x}_k = \overline{x}_{k-1} + \frac{1}{n}\left[x_k - x_{k-n} \right] \tag{10}$$

This is known as a **moving average** filter because the average at *k*th instant is based on the most recent set of *n* values. In other words, at any instant, a **moving window** of *n* values is used to calculate the average of the data sequence (see Figure 19).

When used as a filter, the value of $\overline{x}_k$ is taken as the filtered value of x_k. The expression is a recursive one, because the value of $\overline{x}_k$ is calculated using its previous value, $\overline{x}_{k-1}$, as reference. This is always the case, regardless of the number of data points (*n*) we consider, calculating the current filtered value requires the use of $\overline{x}_{k-n}$, i.e. the measurement *n* time-steps in the past. This means that:

1. the filtering cannot be initiated reliably until *n* measurements have been made, and

2.   We need to store the value of $\overline{x}_{k-n}$ which, depending on the way the algorithm is coded, may require up to n storage locations.
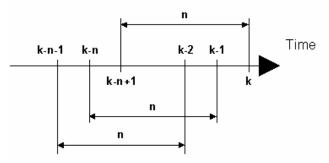

Fig. 19. Moving Window of n Data Points

Additionally, the technique places equal emphasis on all data points. Thus a value in the past will have the same influence as a more current measurement when calculating the filtered signal. This may be a desirable feature when the mean value of the measurement is almost constant, but not when the vehicle moves at various acceleration rates. These problems can however, be reduced by generating the filtered value in a slightly different manner. Actually, in dynamic systems, such as forward vehicle monitoring, the most current values tend to reflect better the state of the process. A filter that places more emphasis on the most recent data would therefore be more useful. Such a filter can be designed by following the procedure used in developing the moving average filter. As before, the starting point is the mean expressed as:

$$\overline{x}_k = \frac{1}{n} \sum_{i=k-n+1}^{k} x_i \tag{11}$$

But in this case, consider also the mean with one additional point

$$\overline{x}_{k+1} = \frac{1}{n+1} \sum_{i=k-n+1}^{k+1} x_i = \frac{1}{n+1} \left[x_{k+1} + \sum_{i=k-n+1}^{k} x_i \right] \tag{12}$$

Since $\displaystyle\sum_{i=k-n+1}^{k} x_i = n\overline{x}_k$,

therefore,

$$\overline{x}_{k+1} = \frac{1}{n+1} \left[x_{k+1} + n\overline{x}_k \right] = \left(\frac{1}{n+1} \right) x_{k+1} + \left(\frac{n}{n+1} \right) \overline{x}_k \tag{13}$$

By shifting the time index back one time-step, we obtain the corresponding expression for $\overline{x}_k$ as:

$$\bar{x}_k = \left(\frac{1}{n+1}\right)x_k + \left(\frac{n}{n+1}\right)\bar{x}_{k-1} \tag{14}$$

To simplify the notation, let $\alpha = \left(\frac{n}{n+1}\right)$, which implies that $(1-\alpha) = \left(\frac{1}{n+1}\right)$. We can write the filter as:

$$\bar{x}_k = \alpha\,\bar{x}_{k-1} + (1-\alpha)x_k \tag{15}$$

This expression is Exponentially Weighted Moving Average Filter. When used as a filter, the value of $\bar{x}_k$ is again taken as the filtered value of x_k. Notice that now, calculation of $\bar{x}_k$ does not require storage of past values of x, and that only 1 addition, 1 subtraction, and 2 multiplication operations are required. The value of the **filter constant**, α, dictates the **degree of filtering**, i.e. how strong the filtering action will be. Since $n \geq 0$, this means that $0 \leq \alpha < 1$. When a large number of points are being considered, $\alpha \to 1$, and $\bar{x}_k \to \bar{x}_{k-1}$. This means that the degree of filtering is so great that the measurement does not play a part in the calculation of the average! On the other extreme, if $n \to 0$, then $\bar{x}_k \to x_k$ which means that virtually no filtering is being performed. The Exponentially Weighted Moving Average filter places more importance to more recent data by discounting older data in an exponential manner (hence the name). This characteristic can be illustrated simply by describing the current average value in terms of past data.

For example, since $\bar{x}_k = \alpha\,\bar{x}_{k-1} + (1-\alpha)x_k$, Then

$$\bar{x}_{k-1} = \alpha\,\bar{x}_{k-2} + (1-\alpha)x_{k-1} \tag{16}$$

Therefore,

$$\bar{x}_k = \alpha\,\bar{x}_{k-1} + (1-\alpha)x_k = \alpha[\alpha\,\bar{x}_{k-2} + (1-\alpha)x_{k-1}] + (1-\alpha)x_k \tag{17}$$

i.e.

$$\bar{x}_k = \alpha^2\,\bar{x}_{k-2} + \alpha(1-\alpha)x_{k-1} + (1-\alpha)x_k \tag{18}$$

But

$$\bar{x}_{k-2} = \alpha\,\bar{x}_{k-3} + (1-\alpha)x_{k-2} \tag{19}$$

Therefore,

$$\begin{aligned}\bar{x}_k &= \alpha^2[\alpha\,x_{k-3} + (1-\alpha)x_{k-2}] + \alpha(1-\alpha)x_{k-1} + (1-\alpha)x_k\\ &= \alpha^3\,\bar{x}_{k-3} + \alpha^2(1-\alpha)x_{k-2} + \alpha(1-\alpha)x_{k-1} + (1-\alpha)x_k\end{aligned} \tag{20}$$

If we keep on expanding x terms on the right hand side, we will see that the contribution of older values of x_i are weighted by increasing powers of α. Since α is less than 1, the contribution of older values of x_i becomes progressively smaller. The weighting on x_i may be represented graphically by the following plot (Figure 20):
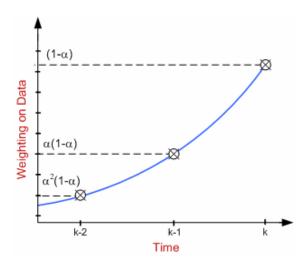
Fig. 20. Exponential weighting effect

What this means is that in calculating the filtered value, more emphasis is given to more recent measurements. By applying this approach to red curve of figure 18 and using MATLAB, we obtain more satisfactory results than before (very close to real distance) in overall fusion system (Figure 21)
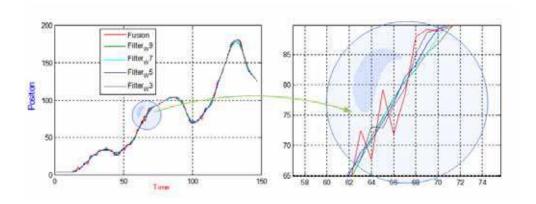

Fig. 21. Applying moving average filter with different windows sizes of 3,5,7,9

According to several experiments and as can be seen from figure 21, moving average filter with windows size of 5 met better following and more smoothing on our fusion graph. Figure 22 shows Fusion graph before (red) and after applying the filtering approach (green).
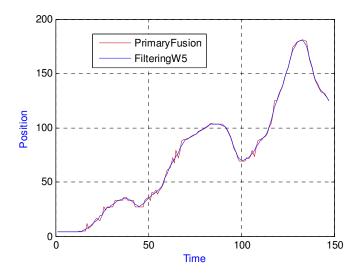
Fig. 22. Sensor Data Fusion before and after applying moving average filter

8. Future works

For future works, it could be continued with the last two sections of fusion processing (Figure 9) called situation refinement. During situation refinement it is tried to understand the *meaning* of the current situation around the vehicle: questions like "is this group of objects some meters in front of the vehicle a group of vulnerable pedestrians willing to cross the street?" or they are some bicyclists?; "Will the traffic light ahead turn to red in a few seconds?". These could be done by considering a priori knowledge and additional environment information coming from digital maps, vehicle-to vehicle as well as vehicle-to-infrastructure communication, etc.

And finally, *Trajectory Prediction,* could be done by taking the history of an object's state parameters and attributes; for example its position and its respective speed in the past, and try to predict the object's state in the near term future.

9. Conclusion

As the cars play an important and wide spreading role in transportation systems of all countries, the chance of having accident is also greatly increased. A number of cars, automobiles, bicycles and also pedestrians may involve in such accidents which sometimes cause miserable disasters. This fact attracts increasing numbers of researchers to work on driver assistance systems to be installed on modern automobiles. Such systems involve lots of considerations ranging from technical to cultural issues. One of interesting technical challenges on this field is the way that sensors communicate with each other and

the control unit and also the methods to collect, interpret and blend data come from these sensors, which is considered in the fields of sensor networks and sensor fusion techniques.

In this literature, the technologies of driver assistant systems and emerge of these systems are discussed. Also key features for an ADAS are introduced in addition with common problems and challenges that will appear in the implementation procedure. Furthermore a generalized framework for selecting proper sensors and mounting them on vehicle body in addition with a comprehensive methodology for employing sensors, initializing them and fusion of data from such sensors are introduced to be suitable for lots of applications and working environments.

Finally, a modern approach based on fuzzy logic and a filtering improvement is discussed to work as the multi-sensor data fusion core, which is implemented in FuzzyTech and MATLAB and the related experimental results are illustrated within the text.

10. References

Amditis, A.; Floudas, N.; Polychronopoulos, A.; Bank, D.; Broek, B & Oechsle, F. (2006), Sensor Data Fusion for Lateral Safe Applications, *13th World Congress & Exhibition on Intelligent Transport Systems and Services*, 12 October 2006, London, UK.

Cheng, H.; Zheng, N.; Zhang, X.; Qin, J & Wetering, H. (2007), Interactive Road Situation Analysis for Driver Assistance and Safety Warning Systems: Framework and Algorithms, *IEEE Transactions on Intelligent Transportation Systems,* Vol. 8, No. 1, March 2007

Duda, R. O. & Hart, P. E. (1972), Use of the Hough transformation to detect lines and curves in pictures, *Communications of the ACM,* Vol. 15 , Issue 1, pp. 11-15, January 1972

Hsieh, Y.; Lian, F. & Hsu, C. (2007), Optimal Multi-Sensor Selection for Driver Assistance Systems under Dynamical Driving Environment, *Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference,* pp. 696-701, Sept. 30 - Oct. 3, 2007, Seattle ,WA, USA

Huang, F; Wei, S.K & Klette, R. (2006), Calibration of Line-based Panoramic Cameras, *Computational Imaging and Vision*, pages 55-84, Springer, Dordrecht, 2006

Irion, J; Flament, M.; Mäkinen, T.; Polychronopoulos, A. et al. Specifications and architecture for PReVENT safety functions, *Sixth framework program of European Commission*, Contract number FP6-507075, Version 1.0, March 2006

Kaempchen, N. & Dietmayer, K. (2003), Data synchronization strategies for multi-sensor fusion, *10th World Congress on Intelligent Transportation Systems*, Issued No. 2250, November 2003, Madrid, Spain

Kalman, R. E. (1960), a New Approach to Linear Filtering and Prediction Problems, *Transaction of ASME Journal of Basic Engineering*, 1960

Langheim, J; Buchanan, A.; Lages, U. & Wahl, M. (2001), CARSENSE – New environment sensing for advanced driver assistance systems, *2001 IEEE Intelligent Vehicle Symposium (IV 2001)*, pp. 89-94, May 13-17, 2001, Tokyo, Japan

Langheim, J; Henrio, J & Liabeuf, B. (1999), ACC Radar System « Autocruise » with 77 GHz MMIC Radar, *ATA*, Florence, 1999.

Lee, K. J. (2001), Reactive navigation for an outdoor autonomous Vehicle, *Technical Report, University of Sidney,* December 2001, Australia

Leohold, J & Schmidt, C. (2004), Communication Requirements of Future Driver Assistance Systems in Automobiles, *Proceedings of 2004 IEEE International Workshop on Factory Communication Systems,* pp. 167-174, Sept. 2004, Germany

Li, F.; Zang, Q.; Klette, R. (2003), Accuracy Improvement in Camera Calibration, *International Conference of Image and Vision Computing*, pages 338--342, Massey University, 2003, New Zealand

Luo, Ren C.; Chou, Y. & Chen O. (2007), Multisensor Fusion and Integration: Algorithms, Applications, and Future Research Directions, *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation*, pp. 1986-1991, August 5 - 8, 2007, Harbin, China.

Macci, D.; Boni, A.; Cecco, M. & Petri, D. (2008), Multisensor Data Fusion, *IEEE Instrumentation & Measurement Magazine,* Part 14, pp. 24-33, June 2008

Marchal, P.; Gavrila, D.; Latellier, L.; Meinecke, M; Morris, R. & Töns, M. (2003), SAVE-U: An innovative sensor platform for Vulnerable Road User protection, *Intelligent Transport Systems and Services*, 16-20 November 2003, Madrid, Spain

Mammar, S.; Glaser, S. & Netto, M. (2006), Time to Line Crossing for Lane Departure Avoidance: A Theoretical Study and an Experimental Setting, *IEEE Transactions on Intelligent Transportation Systems,* Vol. 7, No. 2 , pp. 226-241, June 2006

Mendes, A.; Bento, L. & Nunes, U. (2004), Multi-target Detection and Tracking with a Laserscanner, *2004 IEEE Intelligent Vehicles Symposium*, pp. 796-801, June 14-17, 2004, Parma, Italy

Rezaei Ghahroudi, M. & Fasih, A. (2007), A Hybrid Method in Driver and Multisensor Data Fusion, Using a Fuzzy Logic Supervisor for Vehicle Intelligence, *2007 International Conference on Sensor Technologies and Applications (SENSORCOMM2007)*, pp. 393-398, October 14-20, 2007, Valencia, Spain

Schneider, M. (2005), Automotive Radar – Status and Trends, *Proceedings of the German Microwave Conference (GeMiC 2005),* Hildesheim, pp. 144-147, April 5-7, 2005, Ulm, Germany

Strobel, T.; & Coue, C. (2004), Compendium on Sensor Data Fusions, *Profusion, Sixth Framework Programme of PreVENT Project*, Germany, 2004

Tardos, J. D.; Neira, J; Newman, P. M. & Leonard, J J (2002), Robust mapping and localization in indoor environments using sonar data, *The International Journal of Robotics Research,* Vol. 21, No. 4, 311-330, April 2002

Tatschke, T.; Park, S.; Amditis, A.; Polychronopoulos, A.; Scheunert, O. & Aycard, O. (2006), ProFusion2 – Towards a modular, robust and reliable fusion architecture for automotive environment perception, *Jürgen Valldorf, Wolfgang Gessner (Ed.): Advanced Microsystems for Automotive Applications 2006 (AMAA),* pp. 451-469. Springer, Berlin, Germany

Trivedi, M. M.; Gandhi, T. & McCall, J. (2007), Looking-In and Looking-Out of a Vehicle: Computer-Vision-Based Enhanced Vehicle Safety, *IEEE Transactions on Intelligent Transportation Systems*, VOL. 8, NO. 1, pp. 108-120, March 2007

Wahl, M. & Doree, G. (2000), A CAN application layer for an experimental real time obstacle detection study, *Proceedings of 2000 IEEE intelligent Transportation Systems*, pp. 276-281, October l-3, 2000, Dearborn (MI), USA

Welch, G. & Bishop, G. (2001), an Introduction to Kalman Filter, *Course 8 of SIGGRAPH 2001*, By ACM, Inc., August 21, 2001, Los Angeles, CA, USA

Sensor and Data Fusion

Edited by Nada Milisavljevic

Data fusion is a research area that is growing rapidly due to the fact that it provides means for combining pieces of information coming from different sources/sensors, resulting in ameliorated overall system performance (improved decision making, increased detection capabilities, diminished number of false alarms, improved reliability in various situations at hand) with respect to separate sensors/sources. Different data fusion methods have been developed in order to optimize the overall system output in a variety of applications for which data fusion might be useful: security (humanitarian, military), medical diagnosis, environmental monitoring, remote sensing, robotics, etc.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Mahdi Rezaei Ghahroudi and Reza Sabzevari (2009). Multisensor Data Fusion Strategies for Advanced Driver Assistance Systems, Sensor and Data Fusion, Nada Milisavljevic (Ed.), ISBN: 978-3-902613-52-3, InTech, Available from:
http://www.intechopen.com/books/sensor_and_data_fusion/multisensor_data_fusion_strategies_for_advanced_driver_assistance_systems

INTECH
open science | open minds