Advanced multiple objects tracking by fusing radar and image sensor data - Application on a case study.

Fabio Tango  
Advanced Driver Assistance Systems Department  
Centro Ricerche Fiat  
Strada Torino, 50 – 10043 Orbassano, Italy  
fabio.tango@crf.it

Eric Richter, Ullrich Scheunert, and Gerd Wanielik  
Professorship of Communications Engineering  
Chemnitz University of Technology  
Reichenhainer Str. 70, 09126 Chemnitz, Germany  
{eric.richter, ullrich.scheunert, gerd.wanielik}@tu-chemnitz.de

Abstract – The aim of this paper is to present a multiple object tracking data fusion technique, which fuses radar, image, and ego vehicle odometry. The data are fused at a high level, which leads to reliable and stable tracking results providing also additional features as width estimation and the detection of stationary objects. A “real” application of these algorithms is illustrated on a specific use-case: the SASPENCE system, developed inside the Integrated Project PeReViNT. The principle results show that such a data-fusion technique is actually able to improve the SASPENCE performances, especially in terms of extension of operative scenarios, a basic issue for the system functionality.

Keywords: Data fusion, Tracking, Extended Kalman filtering, Estimation, Image processing, Radar.

1 Introduction

The Preventive and Active Safety Applications project (PeReViNT), which is part of the Sixth Framework Programme, contributes to the safety goals of the European Commission (EC) [13]. PeReViNT addresses multiple functional fields and covers most areas of active safety; in particular: Safe Speed and Safe Following, Lateral Support and Driver Monitoring, Intersection Safety, and Vulnerable Road Users as well as Collision Mitigation. The majority of these functions are characterized by using perception strategies based on multi-sensor platforms and multi sensor data fusion. Hence, the strategy of PeReViNT was to initiate a cross-functional subproject called Profusion2 in order to streamline and to develop the subject of multi sensor data fusion in some greater degree of depth and in a more systematic approach as compared to the primarily function-driven subprojects. Thus, the role of Profusion2 is to define certain standards and develop fusion algorithms, which should be used by the functional subprojects, after gathering the requirements from them. An additional rationale of Profusion2 is motivated by the observation that a variety of national and international research projects are devoted to the development and improvement of active and preventive safety systems, but all of them are affected by the limited performance and even by deficiencies of the currently available sensor platforms. Since PeReViNT is considered as the core of the e-Safety research and development initiative, it has been obvious to embrace a cross-functional subproject that adopts a variety of challenges and open issues in the field of multi-sensor perception.

In this context, an example of application of such Profusion2 algorithms to one PeReViNT function is well represented by the activity carried out in the SASPENCE project, on which a multiple object tracking technique, fusing Image and Radar data, has been implemented.

This paper is structured as follows. After this introduction, next chapter presents an overview of the SASPENCE system. Chapter 3 describes the main issues of the data-fusion system as developed in Profusion2 project and implemented on SASPENCE demonstrator car. The chapters 4 and 5 give more details of this multi-tracking algorithm. The main results are illustrated in chapter 6, together with the description of the experimental phase carried out. Finally, the paper is finished with short conclusions.

2 System overview

The sub-project SASPENCE (Safe Speed and Safe Distance) is part of the Integrated Project PeReViNT, and in particular, it belongs to the Safe Speed and Safe Following area.

In Europe, a considerable part of lives are lost in traffic accidents due to inappropriate vehicle speed (e.g. in curves) or distances, factors which are well known to be one of the major causes of accidents on European roads. In fact, speeding increases the risk of injuries and death very rapidly, also at speeds that slightly exceed the proper velocity value for a given situation. In this context, the 39-months project SASPENCE aims at developing and evaluating an innovative system able to perform the Safe Speed and Safe Distance concept, that means to aid the driver in avoiding accident situations related to excessive speed or too short distances [14][15].

799
2.1 System Architecture

In this paragraph, the system functions are described, together with the main modules and components, which provide the functionality of SASPENCE. There are four main functions constituting the SASPENCE application [16]:

1. Warning for excessive speed related to the situation, specifically for an obstacle ahead or for approaching a curve too fast
2. Warning for too short distance from the obstacle ahead
3. Advice for approaching too fast a landmark (such as pedestrian crossing, railway crossing, etc.)
4. Information about speed limits and related violation

The functional modules of the SASPENCE system are shown in Figure 1.

Figure 1: Functional architecture of the SASPENCE system

2.2 System Implementation

The SASPENCE system is installed in two demonstrator vehicles: a “Fiat Stilo Multiwagon” for CRF and a “Fiat Stilo Hatchback” for TRW, as shown in Figure 2 (where the CRF car is illustrated). The engine compartment is instrumented with the mechanical components of the active accelerator pedal, and the data processing units and interfacing components are installed in the boot of the vehicle. Both vehicles use exactly the same system architecture, but vary slightly in the implementation of the HMI [17]. In addition to the visual warning display (double modality: set of LEDs on odometer and central display on the dashboard) and the haptic accelerator, the CRF vehicle uses audible warnings, whereas the TRW vehicle is equipped with a vibrating seatbelt.

The first step after a new observation at time \( k \) arises is to compensate the ego motion that took place between the last occurred observation \( 1 \) and the current one \( k \). The object’s state vector is

\[
\mathbf{x}_o = \begin{pmatrix} x_o & y_o & \theta_o & v_o & \omega_o & d_o \end{pmatrix}^T,
\]

where \( (x, y) \) is the position and \( \theta \) the orientation of the object in the ego vehicle coordinate system. Furthermore, \( v \) denotes the relative velocity, \( \omega \) the yaw rate, and \( d \) the width of the object.

The first step after a new observation at time \( k \) arises is to compensate the ego motion that took place between the last occurred observation \( \hat{y}_{k-1} \) and the current one \( \hat{y}_k \) (cp. Section 5.2).
After this, a new state $\tilde{x}_o^*$ with the covariance $P_o^*$ can be predicted for each object $o \in O$ as stated in section 5.1. These predicted states are transformed into the according observation space

$$\tilde{y}_o = \begin{pmatrix} \tilde{r}_o \\ \tilde{\phi}_o \\ \tilde{v}_o \end{pmatrix}$$

$$\tilde{y}_o = \begin{pmatrix} x_o \\ y_o \\ d_o \end{pmatrix}$$

where $r$ is the distance, $\phi$ the angle, and $v$ the relative velocity measured by the radar. $(x, y)$ denotes the position and $d$ the width of the object given in image coordinates (see section 5.3 for details). By applying the non-linear transformation functions $h_I h$ from section 5.3 to the predicted state $\tilde{x}_o^*$ and their Jacobians

$$\frac{\partial R/I h}{\partial x_o}$$

$$U^* = R/I J_{o}^* P_{o}^* R/I J_{o}^T$$

In case of a new radar observation, a gating takes place to remove outliers from the observation list: If the probability $p$ calculated by the density function

$$p\left(\tilde{y}^* | \tilde{y}^* \right) = \frac{1}{2\pi^2 |\det(U^*)|} e^{-\frac{1}{2}(\tilde{y}^* - \tilde{y})^T (U^* + R)^{-1} (\tilde{y}^* - \tilde{y})}$$

is smaller than a certain threshold $p_{\text{min}}$, the observation is removed from the observation list and used to initialize a new object ($R$ denotes the radar measurement covariance). A one-to-one assignment is applied to the remaining observations and the filter list using a simple nearest neighbor method; filters to which an observation was assigned are updated using the standard EKF equations [5] and [6].

Image observations are treated in another way: At first, it is verified whether an already created object is inside the ego vehicle’s future path, which usually means that the object is in front of the ego vehicle. This is done for two reasons: firstly, for reducing processing time in order to keep the system real time capable; secondly, to allow a hierarchical classification as described in section 4. If an object exists, a rectangular region of interest (ROI) is created around the estimated object position, which is transformed into the image space by (18). Within this ROI, a vehicle-like shape is searched and if found, a more precise rectangular image observation is created. This observation can be incorporated into the filter using the standard EKF equations [5] and [6].

Finally, a simple object confidence level value is increased if an observation was assigned to a filter and decreased if not, so that old and potentially not any more existing objects can be deleted.

![Diagram](image-url)

**Figure 3: Data fusion system architecture**

### 4 Image processing

There exist various vision methods for detecting the back of a vehicle, such as symmetry-based ([10], [11]) or margin, texture, and symmetry characteristics based approaches [12]. Therewith, it is often needed to combine extracted features from different levels in a geometric and logical way, which can be done as described below.

As new objects are created by non-assignable radar observations only, the image processing can operate on already existing objects. Thus, a Region of Interest (ROI) is defined around the current estimation in the image space. This ROI is supposed to enclose the whole object.
A **local orientation coding** (LOC) [9] operator is used to find distinctive image features like edges and line like structures (see Figure 4).

![Figure 4: Image features by LOC filter processing (yellow points)](image)

A Hough Transform [7] is used afterwards to detect horizontal structures in the data created by the LOC operator (see Figure 5).

![Figure 5: Extracted line features as a result of Hough transformation after LOC processing](image)

These features were chosen because passenger cars and trucks have dominating horizontal features seen from the back. The single lines are combined to higher order primitives, in this case to the bounding rectangle of the combined horizontal lines. The aim is to identify the bounding rectangle of the vehicles back, which is in this example represented by stacked horizontal line features.

To find the higher order structures a hierarchical detection and classification procedure is used [2], which introduces the **Weighted modified Hamacher operator**

\[
\mu(x) = \left[\mu_1(x), \mu_2(x), \ldots, \mu_i(x)\right]^T
\]

has. Each

\[
\mu_i(x) = \left(1 + \left(\frac{|m_i - x_i|}{c_i}\right)^d\right)^{-1}
\]

represents a one dimensional potential function. The system presented in this paper uses two of them, one for the distance or degree of overlapping of two lines and one for evaluating the height to width ratio of a resulting rectangle.¹

The combination of multiple line segments is evaluated by assigning a membership-value \(\mu^D\), which expresses the assignment to the class “good matching (overlapping) horizontal lines” in terms of the overlapping of multiple lines (in relation to the line widths, see Figure 5: e.g. line 2 does not overlap with line 1; line 1 and 5 do overlap to almost 100%). This is done for each line combination in the image. The particular feature was chosen because all horizontal lines at the vehicles back are in the best case overlapping to nearly 100%. By assigning membership values, the quality of the overlapping can be expressed. The second feature that is evaluated is the bounding rectangle of the combined line sets. A certain height/width ratio is treated as strong evidence for the object hypothesis vehicle. The usage of the Weighted modified Hamacher operator allows a step by step testing and assigning components to a higher level feature or object, an example can be seen in Figure 6. For details on Hamacher operator and feature combination see [8].

![Figure 6: Feature evaluation and combining using the Weighted modified Hamacher operator](image)

The membership value \(\mu^{(1)}\) in Figure 6(b) is the final result for the combination of the lines 1, 4 and 5. The bounding box (blue) of these lines is chosen to be the

¹ The parameters position \(m\), width \(c\) and sharpness \(d\) of these functions have been chosen by evaluating the distribution of the image features of interest in a set of test images.
candidate with the highest confidence in the hypothesis
vehicle since the evidence pointing towards it with
\( \mu^{(1)} = 0.997426 \) and the weight \( n = 3 \) is the highest
compared to other combinations e.g. like the one in Figure
6(a).

5 Filtering techniques

The well-known Extended Kalman Filter (EKF – [5],
[6]) is used to estimate the state of the nonlinear system
by predicting the state \( \hat{x}_{o_{k}} \) and its covariance \( P_{o_{k}} \) using
the nonlinear state transition \( \hat{x}(t + T) \) (see section 5.1) and
correcting this prediction by incorporating observations
(see section 5.3) which may come from different sensors.
As both, the state transition and the observation models
are described by nonlinear functions, linearization is
needed for both in order to be able to transform uncertain-
ties, resp. covariances. This linearization is one of the
main drawbacks of the EKF, as it often leads to an under-
estimation of the covariances [1], which has to be consid-
ered in setting the filter parameters.

5.1 The CTRV motion model

As tracking of moving objects means to estimate their
position, velocity, and moving direction, a suitable
movement model has to be used which describes the dy-
amic behavior of the object as best as possible.
The Constant Turn Rate and Velocity (CTRV) model
[3], [4] as a curvilinear one is used to model the move-
ment of the objects tracked by the system. Thus, a move-
ment at constant turn rate and constant velocity is as-
sumed, which implies a constant radius
\[ R = -\frac{v}{\omega} = \text{const.} \] (10)

This leads to the following state transition:
\[ \hat{x}(t + T) = \hat{x}(t) + \int_{0}^{T} \hat{x}(t) dt \] (11)

\[ \hat{x}(t + T) = \begin{bmatrix} \frac{v}{\omega}(\sin(\omega T + \theta(t)) - \sin(\theta(t))) + x(t) \\ \frac{v}{\omega}(-\cos(\omega T + \theta(t)) + \cos(\theta(t))) + y(t) \\ \omega T + \theta(t) \\ v \\ \omega \\ d \end{bmatrix} \] (12)

5.2 Ego motion compensation

In order to compensate the ego motion
\[ \hat{x}_{e} = (x_{e}, y_{e}, \theta_{e})^{T}, \] (13)
it is estimated within a separate EKF by the usage of
odometry observations velocity and yaw rate and applied
to existing objects \( o \in O \)

\[ \hat{x}_{o \text{ compensated}_{k-1}} = f(\hat{x}_{o_{k-1}}, \hat{x}_{e_{k}}), \] (14)

where \( f \) denotes the nonlinear coordinate system
transformation. As the ego motion \( \hat{x}_{e_{k}} \) is an estimation,
too, its covariance \( P_{e_{k}} \) needs to be incorporated into the
compensated object state, otherwise the system could tend
to underestimate the object’s covariance. This is done by
derivation time discrete covariance matrices by linearization
\[ J_{o} = \frac{\partial \hat{x}_{o}(t + T)}{\partial \hat{x}_{o}} \] (15)
and using them to transform the uncertainty of the object
into the new ego motion coordinate system:
\[ P_{o \text{ compensated}_{k}} = J_{o} P_{o_{k-1}} J_{o}^{T} + J_{e} P_{e_{k}} J_{e}^{T} \] (16)

It should be noted that this only approximates the real
distribution due to the linearization.

5.3 Observation models

The transformation into the observation spaces intro-
duced by (2) and (3) is done by using the according trans-
formation function:
\[ r h(\hat{x}_{o}) = \begin{pmatrix} \sqrt{x_{o}^{2} + y_{o}^{2}} \\ \arctan \frac{y_{o}}{x_{o}} \\ v_{o} \end{pmatrix} \] (17)

\[ i h(\hat{x}_{e}) = \begin{pmatrix} a_{11} x + a_{12} y + a_{14} \\ a_{13} x + a_{15} y + 1 \\ a_{16} x + a_{17} y + a_{18} \\ a_{19} x + a_{20} y + a_{21} \\ a_{22} x + a_{23} y + a_{24} \\ a_{25} x + a_{26} y + 1 \end{pmatrix} \] (18)

\[ a_{mn} \] can be obtained from the calibration matrix
\[ A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix} \] (19)

6 Results

The quality of the perception system has been evalu-
ated in several tests [18]. Therefore, images from one and
two lane roads in urban and highway scenarios were
labeled manually in order to produce Ground Truth data.
These Ground Truth data has been compared with the
estimated lateral position by data fusion and radar only,
resp. It has been proven that the lateral position error
using data fusion of radar & image is much smaller than
using radar only. As an additional feature of the image
processing, the width of objects is estimated in high qual-
ity, too. Furthermore, stationary objects can be detected
with a small FAR and MAR (respectively False Alarms Rate and Missing Alarms Rate). Real-time constraints\(^2\) are met by the perception system for all scenarios tested.

As shown in Figure 7 and Figure 8, the presented system is able to fuse the data from radar and image sensor to estimate the position, direction, and width of objects in front of the ego vehicle. Vehicles can be detected by the image processing as well as trucks or buses (see Figure 7). Furthermore, the system is capable to detect stationary objects in front of the ego vehicle; an example is given in Figure 8.

![Figure 7: Radar and image tracked truck (red rectangle). The yellow ellipse represents a radar only tracked object.](image1)

![Figure 8: Detected stationary object in a high traffic scenario. Additionally, no false alarm is produced by the bridge, lanterns etc.](image2)

7 Conclusion

In this paper, we presented a data fusion system, which combines radar and image observations at high level. As an additional feature of the image processing, the system is able to estimate the width of an obstacle and to distinguish stationary objects. An approach for a step-by-step testing and assigning components to a higher level feature has been shown using the Weighted modified Hamacher operator. It has been shown that by applying this approach, reliable and stable tracking results can be achieved. Of course, the authors are also aware of some possible limitations, which have to be investigated. Among others, the issue of robustness of the proposed multi-sensor processing has to be taken into account; it can cope with critical environment conditions (sun, rain, fog, etc.) and reflections on roads, fixed cars, etc. In particular, for the former, we think that they shouldn’t impair the whole system functionality, since not both the sensors are affected in the same way by environmental conditions (and this is a further advantage to use a multi-sensor fusion approach); for example, Camera is sensitive to fog, but not Radar. Under this point of view, the authors expect that, in such critical conditions, the system can have a limited functionality (e.g. a reduced operative range) but not a complete loss of effectiveness. Dedicated experi-

\(^2\) Using an Intel Centrino Duo 2GHz processor

References


