Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Incorporating Environmental Knowledge into Bayesian Filtering using Attractor Functions

Andreas Alin, Martin Butz, Jannik Fritsch

2012

Preprint:

This is an accepted article published in IEEE Intelligent Vehicles Symposium (IV). The final authenticated version is available online at: https://doi.org/[DOI not available]

Incorporating Environmental Knowledge into Bayesian Filtering using Attractor Functions

Andreas Alin, Martin V. Butz and Jannik Fritsch

***Abstract*— Many automotive systems use linear approaches to track and predict other traffic participants. While this may be appropriate on highways, linear predictions do not work properly on curved roads or lane crossings. This contribution introduces a generic way for including environmental knowledge – such as the lane trajectory ahead – to anticipate yaw rate and acceleration of other traffic participants. The anticipatory knowledge is used to improve prediction in filtering tasks. It is embedded in a Bayesian framework by introducing attractors, which modify the probabilistic propagation of state estimations. The attractors model how traffic participants typically behave, given environmental knowledge such as lane information, traffic lights, or indicator lights. We demonstrate the potential of this approach by modeling the fact that vehicles usually stay in their lane. We show that given correct context information and nonlinear traffic situations, the tracking error is considerably lower compared to conventional tracking methods. In addition, we also show that the intentions of other traffic participants may be inferred by comparing actual sensory data with anticipated probability distributions, which were generated dependent on alternative attractors.**

I. Introduction

In highly dynamic environments, a behaving agent relies on accurate information and knowledge about other moving entities and the current surrounding for optimal decision making and control. Especially when driving a car, information about both the dynamics of other traffic participants and the traffic situation is highly relevant. Humans appear to use various clues and knowledge to anticipate the behavior of others. For example, when predicting future positions of a vehicle, one typically assumes that the vehicle will follow its current lane. With such anticipatory assumptions, one can compensate for inaccurate or incomplete sensory information. However, such assumptions do not come without risk. On the one hand, incorrect lane knowledge may lead to incorrect anticipations. On the other hand, unexpected behaviors of other driving participants may remain undetected, because the system may believe too strongly in its own predictions. An important task therefore is to build a probabilistic framework that combines context information with kinematic knowledge and sensor information in a maximally effective and balanced manner.

A. Alin is with the Department of Cognitive Modeling, University of Tuebingen, Tuebingen, Germany andreas.alin@uni-tuebingen.de

M. V. Butz is with the Department of Cognitive Modeling, University of Tuebingen, Tuebingen, Germany martin.butz@uni-tuebingen.de

J. Fritsch is with the Honda Research Institute, Offenbach am Main, Germany Jannik.Fritsch@honda-ri.de

Thanks to Marius Klug and Tim Fissler, students at the University of Tuebingen for the assistance.

Classic Bayesian filters, such as the Kalman filter or particle filter approaches, typically combine sensory information with kinematic constraints. In this way, the state information can be projected into the future. The higher the inaccuracy of the applied movement model, the more uncertainty and thus variance will be added to the predicted state. The combination with successive sensory information in turn yields iterative information gains and thus lower variances in the state estimations. Thus, the more accurate the applied predictive model, the lower the loss in state information.

We propose a new approach to create more accurate predictive movement models. The approach utilizes state-dependent context information, thus differentiating the movement model state- and context-dependently. The context information is incorporated into the system by deriving attractors, which specify potential target locations for the considered traffic participants. Given an attractor and a current state estimate, which both consist of position and velocity estimates in our implementation, a likely trajectory can be derived to reach the attractor. In consequence, the movement model probabilistically follows the trajectory. We show that the resulting movement model can outperform movement models that do not incorporate such attractors, especially in nonlinear motion scenarios such as lane following in a curve or lane changes.

The paper is organized as follows: Related work is reviewed in Section II. Next, an overview of the proposed approach is given in Section III. The basic concept of Bayesian filtering with context fusion and the algorithmic approach for defining attractors and influencing the movement model is introduced in Section IV. Results demonstrating the improvement of the filtering process and an exploratory study on behavior detection with the proposed approach are presented in Section V. The paper ends with a summary and conclusions.

II. Related Work

Various papers use bottom-up, sensory-driven approaches to improve their movement models of other vehicles. Barth and Franke [1], [2], for example, use a traditional tracking approach and introduce an additional estimation of the yaw rate by measuring the position from tracked optical flow vectors. In [2] they added the estimation of the yaw rate change, further improving the estimations in intersection scenarios with potential occlusions. The approach is very well-suited to handle intersection scenarios in real-world situations. However, drawbacks arise due to the more error-prone indirect derivation of the yaw rate estimations. More-

over, curves on the road are usually formed like clothoids, where radii continuously change so that the yaw rate is not sufficient for an accurate prediction of future motion.

An alternative approach has been proposed by Althoff et al. [3], where the road network and therefore the behavior of the vehicles is highly deterministic. The system can be used to predict object states from the viewpoint of a fixed observer in a local known area, disregarding potential motion of the observer (that is, ego-motion). It will thus work in traffic monitoring applications like the European INTERSAFE 2 program crossing assistant [4]. However, vehicles leaving the road are not considered and the approach is used for prediction, but not for filtering. Thus, while the approach may be useful in fixed intersection scenarios, more dynamic scenarios, which may include ego-motion, seem not to be immediately applicable.

Brechtel et al. [5] and Gindele et al. [6] use dynamic Bayes nets to predict trajectories of traffic participants based on contextual information. They use scripted behaviors generated a-priori for their predictions. Behaviors and local situational context determine the motion model. Our approach is highly similar in that we also propose to utilize the context for trajectory derivations and thus motion model adaptations. However, we propose to apply a grid-based Bayesian filter, rather than a particle filter, to approximate the involved probability density functions. Furthermore, Gindele et al. use scripted behaviors generated a-priori for their prediction and apply these behaviors to the full probability density function. We propose to further particularize the application of the derived potential behaviors to the individual grid-points in our grid-based Bayesian filter. In consequence, our approach may apply multiple potential behaviors to different parts of the distributed state estimate, allowing the generation of more distributed predictions and thus the more accurate modeling of potential motion trajectories. Moreover, we propose to utilize an attractor concept to derive likely driving trajectories of other traffic participants on the fly.

III. System Overview

In order to track a traffic participant, we use a Bayesian filter. The Kalman filter and particle filter approaches are particular Bayesian filter examples. We use a grid-based Bayesian filter, which was shown to outperform simple Kalman filtering in non-linear motion scenarios [7]. The general design we introduce here is also applicable to particle filters. However, the grid regularity promises computational advantages, which we expect to exploit in the future.

While Kalman filter, particle filter, and grid-based Bayesian filter have the mathematical model in common, they differ in the way the state of the observed vehicle is represented [8]. In design, we represent the relative position and velocity of an observed vehicle from an ego-vehicle perspective. The state variable X of an observed vehicle can be described by $X = (pos, ||\mathbf{v}||, \omega(\mathbf{v}))$, with $||\mathbf{v}||$ representing the velocity and $\omega(\mathbf{v})$ the direction.

To represent the internal belief of the filter, a probability density function over all system states is necessary. The

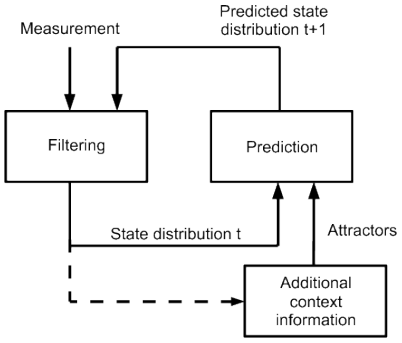


Fig. 1. A schematic view of the proposed Bayesian filter with context fusion algorithm

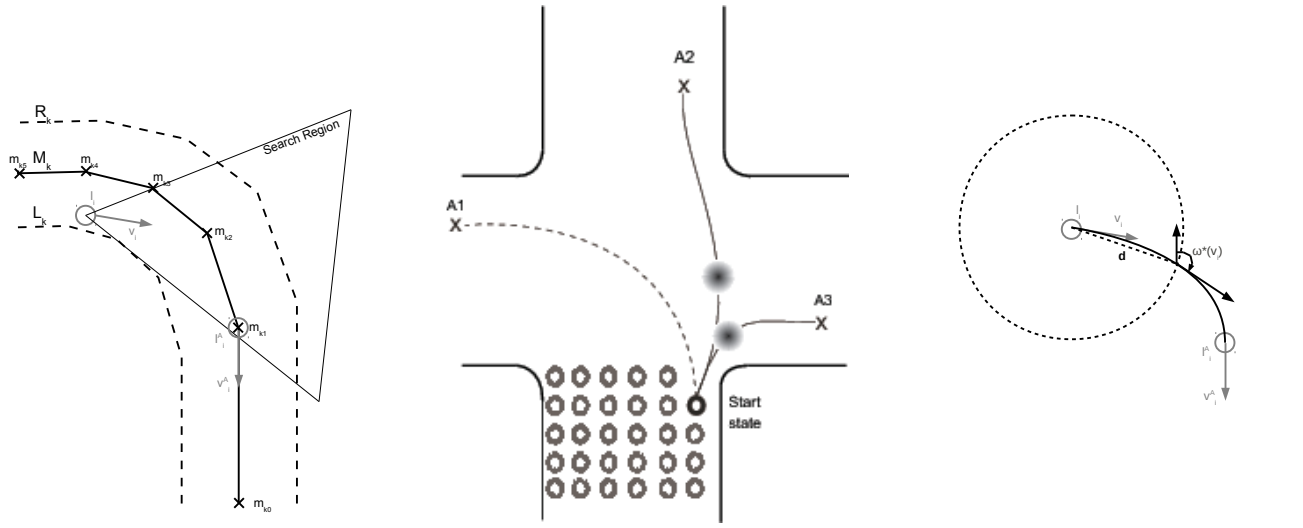
Kalman filter, for example, uses an expectation value with Gaussian noise around one particular state value. If the probability density function is not uni-modal, however, the Gaussian approximation may not fit sufficiently well, so that multi-modal distributions are needed. Therefore we use the grid-based Bayesian filter, which encodes the probability density function of the internal state in a distributed manner.

We distribute the estimate of the state variable X of a vehicle over n uniformly arranged grid nodes with grid node i at position $(l_x, l_y)_i = \mathbf{l}_i$. Each grid node contains a probability $p_{i,t}$ that the observed vehicle is at the nodes' position $\mathbf{l}_i$, given the latest measurement at time t and the grid state from the last time step. Additionally, each grid node i specifies velocity $||\mathbf{v}_i||$ and direction $\omega(\mathbf{v}_i)$ estimates, specifying the most likely velocity and heading direction given the vehicle is located at $\mathbf{l}_i$.

The velocity and direction knowledge is important to predict the position change of the vehicle from one time step to the next. The motion model is implemented by projecting the probability mass $p_{i,t}$ of each grid node i to the surrounding nodes using the velocity and direction estimates. Details of this probability propagation can be looked up in [7]. In this paper, we propose to adjust velocity and direction estimates by applying yaw rates and accelerations, which we derive from the context. An attractor concept is introduced to estimate the most likely driving trajectory of the monitored vehicle, given its current state estimate $(\mathbf{l}_i, ||\mathbf{v}_i||, \omega(\mathbf{v}_i))$ and the current traffic context. From the trajectory estimate, yaw rates and accelerations are derived. Fig. 1 depicts the basic steps of the applied Bayesian filter including the incorporation of additional context information, which is detailed below.

IV. Attractor-Based Context Fusion

As outline above, our grid-based Bayesian filter with context fusion algorithm derives for each grid point location $\mathbf{l}_i$ an attractor, a resulting trajectory to reach the attractor for the grid point i, and it consequently adds yaw and acceleration estimates to the velocity $||\mathbf{v}_i||$ and direction $\omega(\mathbf{v}_i)$ estimates in the grid point. We now first specify the attractor concept for flexibly deriving yaw and acceleration estimates. Next, we show how these estimates are incorporated into the forward motion model.



(a) The algorithm chooses an attractor point from a set of reachable points.

(b) Splines between one node and alternative attractor locations are shown. A1 is filtered out since it does not fulfill (2) and (4).

(c) The spline algorithm combines the node's velocity with the node's attractor velocity to compute ω^*

Fig. 2. Illustrations of crucial steps in the attractor algorithm

A. Attractor Algorithm

Various contexts may be considered with the proposed attractor-based context fusion concept. In this work, we focus on deriving trajectories based on lane information as a first example.

In particular, we specify an attractor function AF, which generates an attractor location $\mathbf{l}_i^A$ as well as a heading direction $\omega(\mathbf{v})_i^A$ and velocity $||\mathbf{v}_i^A||$ given a state estimate in a particular grid node i and the current context c of the road and surrounding traffic, that is:

$$\begin{pmatrix} \omega(\mathbf{v}_i^A) \\ ||\mathbf{v}_i^A|| \\ \mathbf{l}_i^A \end{pmatrix} = AF(||\mathbf{v}_i||, \omega(\mathbf{v}_i), \mathbf{l}_i, c) \tag{1}$$

The general attractor function may be realized by individual algorithms dependent on the attractor type.

In the following, we consider an AF algorithm for lane following. The AF algorithm is detailed in Alg. 1 and illustrated in Fig. 2(a). Each lane k consists of a left lane border L_k, a right lane border R_k, and the lane center M_k. Lane borders are represented with polygons in our input data and are updated in each time step by a virtual lane sensor.

The algorithm considers the lane k in which the grid point i is located in. Next, it generates the set B_k of all points $\mathbf{m}_{kj}$ that lie on the lane center M_k and whose connection to the grid point location $\mathbf{l}_i$ do not intersect with the lane borders. Points $\mathbf{m}_{kj} \in M_k$ may, for example, be equidistantly distributed on M_k. Next, it verifies if the points in the considered set are sufficiently close in distance and angle to the starting state at grid node i. Finally, the algorithm returns the attractor point farthest away from the grid node in that set. Note that the algorithm does not consider velocity changes at this point. Fig. 2(a) illustrates the algorithm in a curve exemplary for one grid point with indicated estimates for velocity and heading direction.

Algorithm 1 Attractor function algorithm considering lane context.

for all lanes k **do**
  if $\mathbf{l}_i$ within lane $k \in K$ **then**
    $B_k \leftarrow \{\mathbf{m}_{kj} | \mathbf{m}_{kj} \in M_k \wedge \overline{\mathbf{m}_{kj}\mathbf{l}_i} \cap L_k = \emptyset \wedge \overline{\mathbf{m}_{kj}\mathbf{l}_i} \cap R_k = \emptyset\}$
    $B_k \leftarrow \{\mathbf{b}_j | \mathbf{b}_j \in B_k \wedge \mathbf{b}_j$ fulfills (2,3) and laneDirectionAt($\mathbf{b}_j$) fulfills (4)$\}$
    $\hat{\mathbf{b}} \leftarrow arg\max_{\mathbf{b}_j \in B_k}(||\overline{\mathbf{b}_j\mathbf{l}_i}||)$
    return (laneDirectionAt($\hat{\mathbf{b}}$), $||\mathbf{v}_i||$, $\hat{\mathbf{b}}$)
  end if
end for

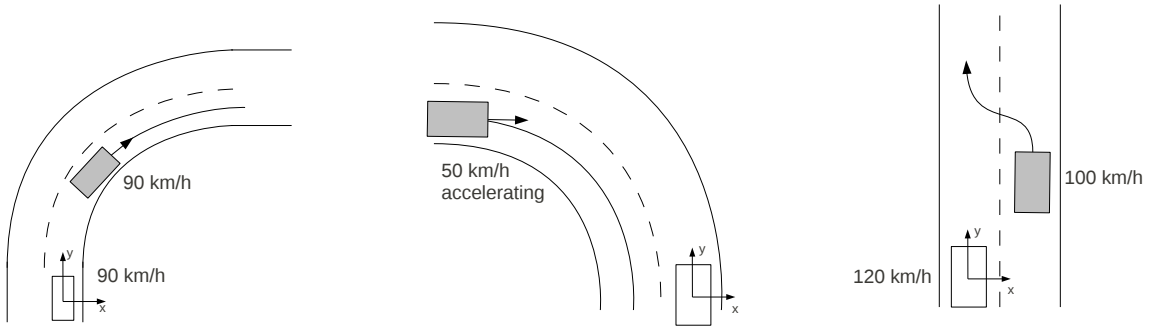
The attractor state reachability assertion considers if the attractor state may be reached from the current state in a reasonable way. Such reachability assertions may consider additional factors, such as information about the road level, e.g. A-road, B-road, inner city, setting a looking-forward distance d_{max} and maximum change in the yaw rate σ_{max} based on the road conditions. In effect, attractor points are only considered within the area around the grid node i that are sufficiently close given the estimated velocity $||\mathbf{v}_i||$ at grid point i as specified in (2) and whose aimed-at heading direction $\omega(\mathbf{v}_i^A)$ does not differ too strongly from the current heading direction $\omega(\mathbf{v}_i)$ in grid point i (4):

$$(||\mathbf{v}_i||) \cdot \Delta t \leq d_{max}, \tag{2}$$

$$|atan2(\omega(\mathbf{v_i}), \overline{m_{kj}l_i})| \leq \sigma_{max} \cdot \Delta t \tag{3}$$

$$|\omega(\mathbf{v}_i) - \omega(\mathbf{v}_i^A)| \leq \sigma_{max} \cdot \Delta t, \tag{4}$$

where σ_{max} specifies the maximum change of the yaw rate and d_{max} specifies the maximum distance change allowed in a time window Δt.



(a) Scenario 1: Vehicle is tracked in a curve by a following vehicle.

(b) Scenario 2: Oncoming vehicle passes in a curve from.

(c) Scenario 3: Slightly slower vehicle cuts in the lane in front.

Fig. 3. Scenario settings

B. Bayesian Filter with Context Fusion

In order to incorporate the lane knowledge in form of the specified attractor function, splines are generated between $\mathbf{l}_i$ and $\mathbf{l}_i^A$, considering start direction $\omega(\mathbf{v}_i)$ and attractor direction $\omega(\mathbf{v}_i^A)$. A similar spline technique was proposed elsewhere [6]. The validity of each potential spline is tested, potentially removing unrealistic trajectories. Fig. 2(b) illustrates this process in an intersection scenario.

A spline is generated and used to adjust the heading direction estimate $\omega(\mathbf{v_i})$ of each grid point i in the following way:

$$d = ||\mathbf{v}_i|| \cdot \Delta T / (\mathbf{l}_i^A - \mathbf{l}_i) \tag{5}$$

$$\omega^*(\mathbf{v_i}) \leftarrow (2\mathbf{l}_i - 2\mathbf{l}_i^A + \omega(\mathbf{v}_i) + \omega(\mathbf{v}_i^A))d^3 + (-3\mathbf{l}_i + 3\mathbf{l}_i^A - 2\omega(\mathbf{v}_i) - \omega(\mathbf{v}_i^A))d^2 + \omega(\mathbf{v}_i)d + \mathbf{l}_i \tag{6}$$

The direction change is derived by probing the spline position at distance d around the start node (5). Here, d is an approximation of the ratio between the drivable path length during a time step ΔT and the length of the spline generated. We approximate it by using the velocity $||\mathbf{v}_i||$ and the distance between $\mathbf{l}_i$ and $\mathbf{l}_i^A$. This approximation fits well for splines with small curvature. Next, this distance d is used to derive the expected resulting heading direction $\omega^*(\mathbf{v_i})$, computing the tangent on the spline at distance d. Fig. 2(c) illustrates this process. Note that we do not consider velocity changes at this point, but such changes may easily be incorporated in future work.

V. Experimental Evaluations

In the following evaluations we consider three scenarios: two curved road scenarios, the first one with the ego-vehicle following another vehicle and the second one with opposing vehicles; in the third scenario a slightly slower vehicle cuts in the lane of the ego-vehicle. While we use the first two scenarios to evaluate the potential of the attractor concept for handling nonlinear motion, the third scenario is used to illustrate the potential of the approach for detecting unexpected and potentially dangerous events.

All experiments were conducted within the freely available TORCS open racing car simulation environment [9], which provides measurements that are used as ground truth data for the evaluation framework. The data was generated by a driving controller following the lane center. We added noise to these measurements to simulate real sensors. In particular, we simulated radar sensors with radial noise of 0.09rad standard deviation and a distance deviation of $0.02m$ times the distance of the sensed car. All results reported in this paper were averaged over 100 runs with identical trajectories but independent noise in the measurements.

A. Scenario 1: Follow into curved road segment

The ego vehicle is following another vehicle in a curved road segment. Both vehicles are driving with a velocity of 90 km/h (see Fig. 3(a)). The results are shown in Fig. 4 comparing the results from a Bayesian filter without context information with the Bayesian filter with context information. The graph shows that the error can be reduced by using relevant context information. The remaining error is mainly caused by the discrete steering behavior of the TORCS controller of the ego-vehicle. Fig. 5 confirms the advantage of the lane following attractor plotting the likelihood at the ground truth, which is larger when the attractor is applied. The results show that the attractor-based adaptation of the motion model can handle radial motion well, even without representing or approximating yaws explicitly.

B. Scenario 2: Passing Opposing Vehicle

We also tested the algorithm in an inner-city like curve scenario, with the opposing vehicle driving with 50 km/h, while our vehicle is static (see Fig. 3(b)). This is comparable with the settings used in [1] and [2]. This setting is more challenging because there are not many time steps in which the other vehicle is observable. Additionally the relative velocities are higher. As a result, there is not much time for the Bayesian filter to calibrate its internal velocity estimates, since the filter is always initialized with zero velocities in all grid nodes.

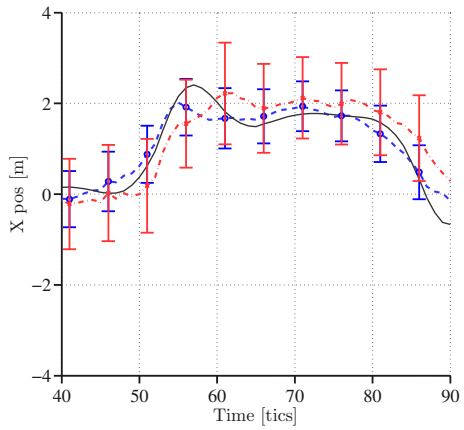


Fig. 4. The x position of the observed vehicle relative to the ego-vehicle over time. The small solid line is the ground truth position, the blue dashed line is the Bayesian filter with context information, and the dotted red line is the Bayesian filter without context information as a baseline.

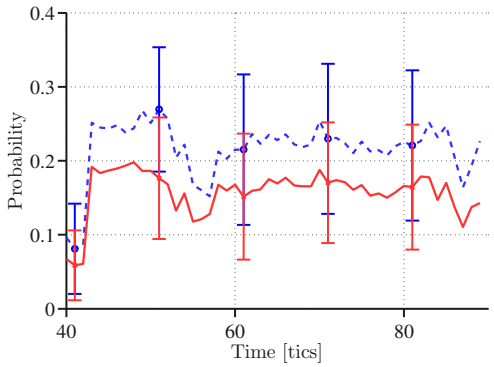


Fig. 5. The probability the grid filter assigns to the ground truth position. The dashed line is produced by the Bayesian filter with context information, the solid line is given by the Bayesian filter without context information as a baseline.

The evaluation shows that the attractor benefits are smaller than in scenario 1. This has several reasons. Firstly, the lane information was not available during the first seven time steps, since the lane detector is not able to see behind curves. Secondly, with the observed vehicle approaching our ego-vehicle, the radar sensor noise is becoming smaller, so that the attractor model influence decreases. Moreover, the sensory noise has a smaller effect given higher relative velocities.

The results in Fig. 6 show that again the additional attractor concept improves the likelihood of the predictions. As explained, first, the attractor cannot apply yielding no difference when compared to the results without attractor. After about seven time steps, however, a difference becomes visible and stays visible until about the 33rd time step. At this point, the vehicle is rather close to the ego-vehicle, which minimizes the sensory noise of the simulated radar sensor. The distance to the ego-vehicle over time and the respective estimates without and with attractor concept are shown in Fig. 7 for comparison.

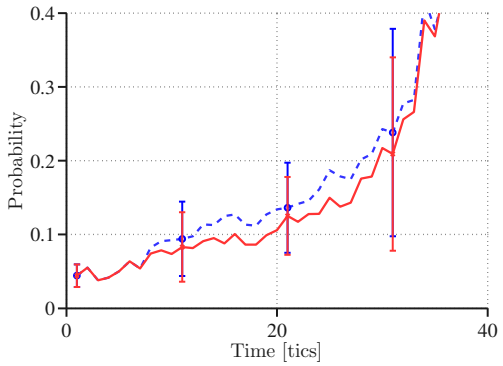


Fig. 6. The probability the grid filter assigns to the ground truth position. The dashed line is produced by the Bayesian filter with context information, the solid line is given by the Bayesian filter without context information as a baseline.

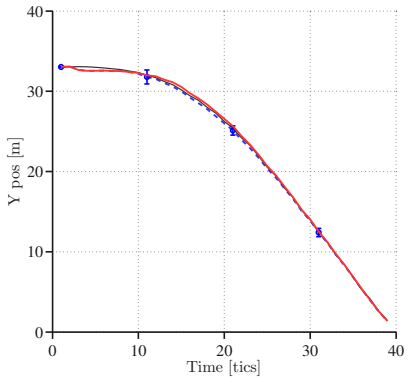


Fig. 7. The y position of the observed vehicle relative to the ego-vehicle over time. The small solid line is the ground truth position, the dashed line is the Bayesian filter with context information, the red solid line is the Bayesian filter without context information as a baseline.

C. Scenario 3: Cut-in Lane Event

In this scenario, we actively compare the predictions of the model without attractor concept with the one with attractor concept, considering the lane information. The active comparison allows us to build a detector for a lane change. If the measurement fits better with the kinematic model than the attractor-based lane following model, the system can be assumed to be ”surprised”. Note that in contrast to the often used Interacting Multiple Models [10] no mixing and weighting between the models takes place here but the likelihood of the alternatives are simply compared.

We evaluated the lane change detector with a cut-in scenario on a freeway with two lanes for each direction (see Fig. 3(c)). The ego vehicle drives with 120 km/h on the left lane. The observed vehicle drives on the right lane maintaining 100 km/h while cutting into the left lane in front of the ego-vehicle. This is a frequently seen risky situation, in which the ego-vehicle needs to brake quickly. A detector for such an event may support a distracted driver.

We use an attractor function, which models that the observed vehicles stays in its lane. The belief in the observed measurement drops quickly when the lane change begins, since the measurement does not comply with the prediction

model. By comparing this with the pure kinematic model (see Fig. 8), an anticipatory lane change detector can be created. We assume an attractor model violation, when the Bayesian filter without attractor model yields better estimation values three times in a row.

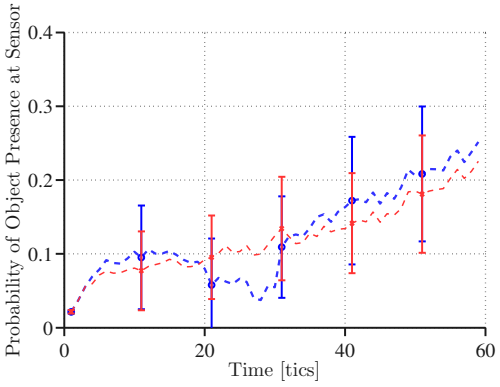


Fig. 8. The probability the grid filter assigns to the measured position. The blue line is produced by the Bayesian filter with context information, the thinner red line is given by the Bayesian filter using the kinematic model only.

To evaluate the anticipatory detector, we decreased the angular noise in the radar sensor to 0.04 and to 0.02 rads for comparison purposes. In Fig. 9 it can be seen that the detector detects the lane change events rather well (for 0.02rads). The actual lane crossing occurs after 30 time steps. With increasing noise levels, more false positives occur (with 0.04rads, gray solid line). Nonetheless, we are confident that the approach has high potential. When the sensory noise can be kept at a reasonably low level, lane change events can be anticipated reliably while the monitored vehicle approaches the border of a lane.

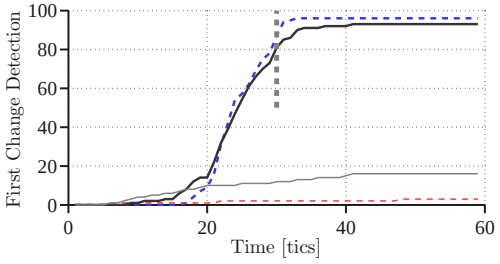


Fig. 9. The integrated time-point of the first lane change detection in the lane changing scenario (thick) versus the lane change detection in a non-lane-changing scenario with same noise sequence (thin) showing the false positive results. Solid lines measured with radial sensor noise of 0.04rad, and dashed lines with 0.02rad. The vertical line marks the time at passing of the lane marking.

While these results are encouraging, the detector was based on successive sensor measurements alone. We are currently improving the detector further by considering the expected measurement noise distribution, that is, the sensor model in combination with the current sensor measurement. If sensory noise is high, more false positives can be expected when disregarding the measurement noise distribution, for example, simply by accidentally perceiving measurements on the right and left side of the lane. Thus, we currently develop a Bayesian lane change detection mechanism.

VI. Summary and Conclusion

We have demonstrated how to improve tracking with a grid-based Bayesian filter with context fusion. The results show that this approach may constitute an alternative to or may complement approaches that try to derive yaw rates from sensory information. We showed that context information can be used to infer potential driving trajectories, which in turn induce anticipatory yaw rates into the system. As a result, tracking performance is increased even without incorporating explicit, bottom-up derived yaw rates.

Moreover, it was shown that dangerous events can be detected in advance by contrasting the estimated likelihood of successive sensory measurements without attractors from the one with attractors. When the Bayesian filter without attractors yields better plausibility values than the filter with attractors given successive sensory measurements, the behavior of the monitored vehicle is unexpected. The experimental results show that such unexpected behavior can be detected *before* a dangerous situation is actually in place. In our example, a lane change was anticipated before it occurred. As a result, preemptive or warning actions may be issued in time.

At the moment, we are evaluating the proposed methodology in other scenarios. We are also generating other attractor functions, for example, for handling turning maneuvers as well as for adapting to expected velocity changes. Finally, the technology is currently also evaluated on real-world automotive data.

References

[1] A. Barth and U. Franke, "Where Will the Oncoming Vehicle be the Next Second?" in *Proceedings of the IEEE Intelligent Vehicles Symposium*. Eindhoven: Springer, June 2008, pp. 1068–1073.

[2] ——, "Tracking oncoming and turning vehicles at intersections," in *Intelligent Transportation Systems, IEEE Conference on*, Madeira Island, Portugal, 2010, pp. 861–868.

[3] M. Althoff, O. Stursberg, and M. Buss, "Stochastic reachable sets of interacting traffic participants," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2008, pp. 1086–1092.

[4] C. Shooter and J. Reeve, "Intersafe-2 architecture and specification," in *Intelligent Computer Communication and Processing, 2009. ICCP 2009. IEEE 5th International Conference on*, aug. 2009, pp. 379 –386.

[5] S. Brechtel, T. Gindele, J. Vogelgesang, and R. Dillmann, "Probabilistisches Belegtheitsfilter zur Schätzung dynamischer Umgebungen unter Verwendung multipler Bewegungsmodelle," in *Proceedings of the 21th Fachgespräch Autonome Mobile Systeme*, 2009, pp. 49–56.

[6] T. Gindele, S. Brechtel, and R. Dillmann, "A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments," in *IEEE Intelligent Transportation Systems Conference*, Madeira Island, Portugal, September 2010.

[7] A. Alin, M. V. Butz, and J. Fritsch, "Tracking moving vehicles using an advanced grid-based bayesian filter approach," *IEEE Intelligent Vehicles Symposium (IV)*, pp. 466–472, 2011.

[8] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, USA: MIT Press, 2006.

[9] (2007) TORCS, the open racing car simulator. [Online]. Available: http://torcs.sourceforge.net/

[10] J. Burlet, T. D. Vu, and O. Aycard, "Grid-based localization and online mapping with moving object detection and tracking," INRIA, Research Report RR-6276, 2007. [Online]. Available: http://hal.inria.fr/inria-00167687/en/