# Probabilistic Situation Recognition for Vehicular Traffic Scenarios

Daniel Meyer-Delius Christian Plagemann Wolfram Burgard

***Abstract*— To act intelligently in dynamic environments, a system must understand the current situation it is involved in at any given time. This requires dealing with temporal context, handling multiple and ambiguous interpretations, and accounting for various sources of uncertainty. In this paper we propose a probabilistic approach to modeling and recognizing situations. We define a situation as a distribution over sequences of states that have some meaningful interpretation. Each situation is characterized by an individual hidden Markov model that describes the corresponding distribution. In particular, we consider typical traffic scenarios and describe how our framework can be used to model and track different situations while they are evolving. The approach was evaluated experimentally in vehicular traffic scenarios using real and simulated data. The results show that our system is able to recognize and track multiple situation instances in parallel and make sensible decisions between competing hypotheses. Additionally, we show that our models can be used for predicting the position of the tracked vehicles.**

## I. INTRODUCTION

A fundamental requirement for an autonomous system to be able to act intelligently is the continuous monitoring and understanding of the current situation it is involved in. Knowing what is going on is relevant for predicting what will happen, which in turn can be used to make informed decisions, avoid risks, and, in general, improve the performance of the system. Situation recognition, however, is not an easy task even if the state of the system or its environment can be estimated accurately. To robustly recognize the current situation at any given time, the temporal context needs to be taken into account. Additionally, the system must be able to deal with ambiguities, since there may be more than one possible interpretation, and some of them might be contradictory. Furthermore, the system needs to deal with uncertainty in the environment, sensor noise, and inaccuracies in its models. On top of this all, the system must recognize situations as they are evolving, that is, in an online fashion.

In this paper, we present a framework for modeling and online-recognition of situations. Although the framework is generic, we focus on a driver assistant application in traffic scenarios and consider situations that typically occur in highway-like driving settings. The situations detected by our current system include passing, following, and aborted passing situations.

Within our framework, the process of change is viewed as a series of snapshots, each describing the state of the system at a particular time. Based on this characterization, we speak of *situation types* and *situation instances*, where a situation instance is defined as a state sequence that has some meaningful interpretation. A situation type, on the other hand, is the set of all situation instances that are grouped under the same interpretation.

All authors are with University of Freiburg, Department of Computer Science, D-79110 Freiburg, Germany.

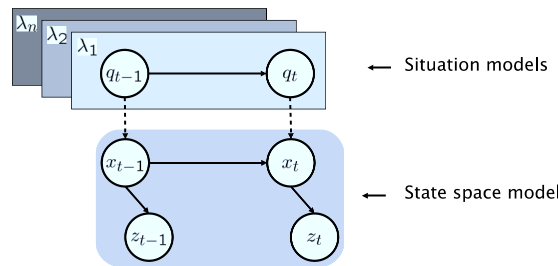


Fig. 1. Schematic representation of the proposed framework for situation modeling and recognition. Each situation type is described by an individual HMM $\lambda_i$. A situation instance exists as long as the corresponding HMM recognizes the state sequence being generated.

We take a model-based approach in which hidden Markov models (HMMs) are used for characterizing and recognizing situations. Each situation type is described by an individual HMM, which specifies the admissible state sequences that correspond to an occurrence of the given situation.

A graphical representation of the proposed framework is shown in Fig. 1. The state space model in the lower part of the figure corresponds to a dynamic Bayesian network that characterizes the system. The upper part of the figure corresponds to a layer of different situation-HMMs, which are evaluated against the estimated state $x_t$ at each point in time $t$.

The contribution of this paper is a practical approach for modeling and recognizing situations in an online setting. As mentioned above, we show how our framework can be used for characterizing typical situations in a vehicular traffic scenario, and how situation instances can be tracked while they are developing. Experimental results using real and simulated data show that our system can recognize and track multiple situation instances in parallel, and make sensible decisions between competing hypotheses. Additionally, we show that our models can be used for predicting the position of the tracked vehicles.

The remainder of this paper is organized as follows. In the next section, we review related work. Then, Section III describes our framework for situation modeling and recognition. In Section V we finally present experimental results illustrating the advantages of our technique.

## II. RELATED WORK

In the field of intelligent agents, *space* and *time* modeling has been approached using qualitative knowledge representation and reasoning, like, for example, using the *situation calculus* [1] or the *event calculus* [2]. Despite the existence of these formalisms that simultaneously represent space and time, most modern approaches combine spatial and temporal calculi. Muller's [3] spatio-temporal theory, for example, is basically a first-order axiomatization of spatio-temporal entities based on RCC [4]. Wolter and Zakharyaschev [5] combine RCC and propositional time logic [6]. Brandon *et al.* [7] also use propositional time logic but combine it with modal logic to produce PSTL (Propositional Spatio-Temporal Logic), a two-dimensional logic capable of describing topological relationships that change over time. Gerevini and Nebel [8] use Allen's interval calculus [9] to temporalize RCC.

All these approaches are mostly focused on the representation of and reasoning about spatio-temporal facts. Several authors have investigated the problem of extracting such facts from quantitative data to recognize relevant temporal configurations of those facts in an online fashion. Ghallab [10], for example, introduced the concept of a *chronicle* as a set of events and a set of temporal constraints, between those events, where the events are symbolic representations obtained from sensors. In an online fashion, the recognition system processes these events and if they match the event model of a chronicle, then an instance of this chronicle occurs. Nagel [11] describes a complete system capable of transforming sequences of video images into a natural text description of spatio-temporal developments. Nagel introduces the notion of a *generically describable situation* as a combination of a state-scheme and an action-scheme, where a state-scheme is a characterization of the state and an action-scheme specifies an action that could be performed if the conditions stipulated by the state-scheme are satisfied.

Similar to the approaches of Ghallab and Nagel, our framework describes each relevant spatio-temporal configuration using an individual model that is evaluated as the state of the system changes. However, the approaches of Ghallab and Nagel do not explicitly deal with the inherent uncertainty in the observations and actions of a system. The hidden Markov model (see Rabiner [12]) is one of the most popular probabilistic models for representing sequences of states that have structure in time. Brand *et al.* [13], for example, represent and classify sequences corresponding to T'ai Chi Ch'uan gestures using *coupled hidden Markov models*. Ghahramani *et al.* [14] use *factorial hidden Markov models* to model a collection of musical pieces of J.S. Bach. Landwehr [15] extracts different activities executed in parallel during the preparation of breakfast at home using *interleaved hidden Markov models*.

Like in the approaches mentioned above, we also use hidden Markov models to describe distributions over meaningful state sequences. However, we additionally present a complete framework for the real-time recognition of sequences that are consistent with the models. This approach is similar in spirit to the one presented by Bennewitz *et al.* [16] where a complete framework for recognizing gestures for human-robot interaction is presented. In contrast to the left-to-right

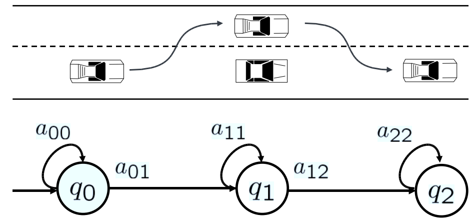


Fig. 2. A *passing* maneuver, in which the reference vehicle (square car in the middle) is being passed on its left hand side by another one. In this case, we divided the maneuver into three stages and, thus, used a 3-state HMM over abstract world states as a model (bottom).

models used by Bennewitz *et al.* we do not put restrictions on the state transitions of the model and we also discuss how recognized situations can be used to predict future developments in the scene.

## III. Modeling Spatio-Temporal Situations

Our approach to modeling the dynamics of realistic systems, such as vehicular traffic, is to assume two layers of abstraction: first, on a higher abstraction level, the so-called *situation models* describe how the system evolves over longer periods of time at a lower spatial resolution (e.g., "car A passes car B on the left") . Secondly, on a more detailed level, a *state-space model* describes the concrete interaction of the agents with the environment involving the relationship between the state $x_t$ of the system and the observations $z_t$.

Fully interweaving both abstraction layers would lead to an intractable model in all but the simplest cases. Therefore, the two layers are loosely coupled as visualized in Fig. 1, that is, the posterior state estimates in the state-space models are treated as fixed "observations" by the situation models. Concretely, as state-space model we assume a dynamic Bayesian network [17], in which the state $x_t$ and observation $z_t$ at time $t$ are characterized by a set of random variables (see Fig. 1). The state $x_t$ of the system at time $t$ is estimated from the sequence of previously obtained observations $z_{1:t} = \{z_1, \ldots, z_t\}$ using the recursive state estimation scheme [18].

On the more abstract level, a *situation instance* is defined as a sequence of states that has some meaningful interpretation. A *situation type* corresponds to the set of all situation instances that are grouped under the same interpretation. To characterize a situation type $s$, we use a continuous hidden Markov model $\lambda_s$, that describes the stereotypical state sequence corresponding to the situation type over the wide range of variations inherent to the different situation instances. A situation HMM consists of a tuple $\lambda = (\mathcal{Q}, A, B, \pi)$, where

- $\mathcal{Q} = \{q_0, \ldots, q_N\}$ represents a finite set of $N$ states.
- $A = \{a_{ij}\}$ is the state transition matrix in which each entry $a_{ij}$ represents the probability of a transition from state $q_i$ to state $q_j$.
- $B = \{b_i(x)\}$ is the observation model, where $b_i(x)$ represents the probability of observing $x$ while being in state $q_i$.

- $\pi = \{\pi_i\}$ is the initial state distribution, where $\pi_i$ represents the probability of state $q_i$ being the initial state.

Although the states $\mathcal{Q}$ in a HMM are hidden, a concrete meaning can often be associated with them [12]. In our case, we choose $\mathcal{Q}$ as a set of $N$ salient states $x$ from the state-space model. The transition probabilities $a_{ij}$ specify the admissible state sequences that correspond to an instance of that situation. Consider, for example, the passing maneuver illustrated in Fig. 2, in which a vehicle is passed on the left side by another car. We can describe this type of situation using an HMM with three states $q_1$, $q_2$, and $q_3$ where the first state $q_1$ corresponds to the passing vehicle being behind of the reference vehicle, $q_2$ corresponds to the passing vehicle being on the left, and $q_3$ corresponds to the passing vehicle being in front of the reference one. In a preliminary version of the framework presented in [19], we used a hard discretization of the state space to define the observation models $B$. In our current framework, the observation model of a situation HMM is given by a finite mixture of the form

$$b_i(x) \quad = \quad \sum_{k=1}^{K} c_{ik} \mathcal{N}(x; \mu_{ik}, \Sigma_{ik}), \tag{1}$$

where $x$ is a state in the state space model, $c_{ik}$ is the mixture coefficient for the $k$-th mixture in state $q_i$, and $\mathcal{N}(x; \mu_{ik}, \Sigma_{ik})$ is a multivariate Gaussian distribution with mean $\mu_{ik}$ and covariance matrix $\Sigma_{ik}$ for the $k$-th mixture in state $q_i$.

In order to estimate the parameters of a situation HMM $\lambda$, we use the Baum-Welch algorithm [20]. This is an approximative iterative optimization technique for maximizing the likelihood of the data. The algorithm takes an initial estimate of the parameters and greedily improves it by following the likelihood gradient. In this work, we assume that the number of states $N$ and mixtures $K$ in the observation model are fixed and given. Therefore, the initial state distribution $\pi$, the transition matrix $A$, and the observation model $B$ are the free parameters to be learned. The training data for each situation type $s$ consists of a set of observation sequences $X_s = \{x^1, \ldots, x^c\}$, where each $x^i$ is a sequence $\{x^i_1, \ldots, x^i_{t_i}\}$ of states of the state space model.

## IV. Situation Recognition

Given a set of $M$ trained situation models $\lambda_{s_1}, \ldots, \lambda_{s_M}$, and a sequence $x_1, \ldots, x_t$ of states of the state space model, our approach to situation recognition is based on evaluating the likelihood $P(x_{1:t} \mid \lambda_s)$ of the sequence for each model $\lambda_s$. This likelihood is computed incrementally using the *forward procedure* (see [12]) given as

$$P(x_{1:t} \mid \lambda) = \sum_{i=1}^{N} \alpha_t(i), \tag{2}$$

where

$$\alpha_{t+1}(j) = \left[ \sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(x_{t+1}), \tag{3}$$

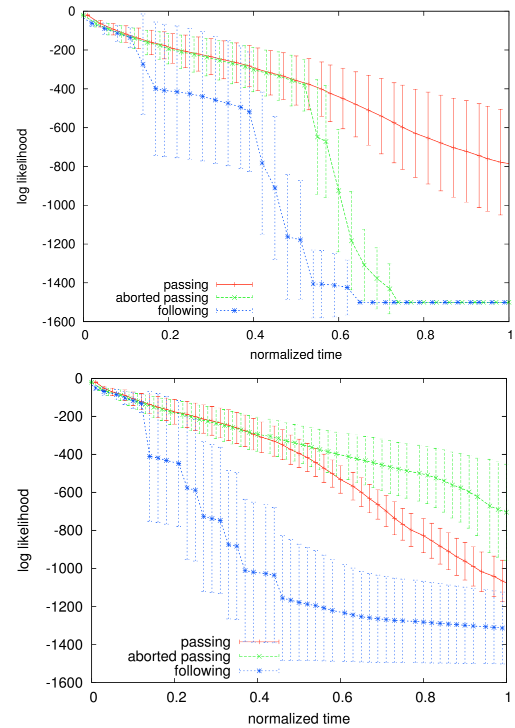


Fig. 3. Average likelihood of 10 observation sequences corresponding to a passing (top) and an aborted passing (bottom) maneuver according to the three different situation models: *passing*, *aborted passing*, and *following*.

and

$$\alpha_1(j) = \pi_j b_j(x_1). \tag{4}$$

At each point in time $t$, the framework incrementally updates the likelihoods computed at time $t-1$ independently for the different models. In certain scenarios, however, it is reasonable to consider two or more situation types as being mutually exclusive or competing. In this kind of problems, we would like to be able to select, among the competing models, the most likely one. Assuming that the stereotypical sequences of the competing situation types can be differentiated and that the learned models accurately characterize them, we can use the likelihood $P(x_{1:t} \mid \lambda_s)$ of the observation sequence given the different models $\lambda_s$ to select the one that provides the better explanation for the sequence [12].

For deciding between two competing models, we compute the posterior odds, which provides a way of evaluating evidence in favor of a probabilistic model relative to an alternative one. The posterior odds $P_{\lambda_1,\lambda_2}$ for two competing models $\lambda_1$ and $\lambda_2$ given an observation sequence $x_{1:t}$ is computed as

$$P_{\lambda_1,\lambda_2} \quad = \quad \frac{P(x_{1:t} \mid \lambda_1) P(\lambda_1)}{P(x_{1:t} \mid \lambda_2) P(\lambda_2)}, \tag{5}$$

that is, the ratio of the likelihoods of the models being compared given the data and the model priors. The likelihood $P(x_{1:t} \mid \lambda)$ of an observation sequence $x_{1:t}$ given a model $\lambda$ can be computed efficiently using the forward procedure

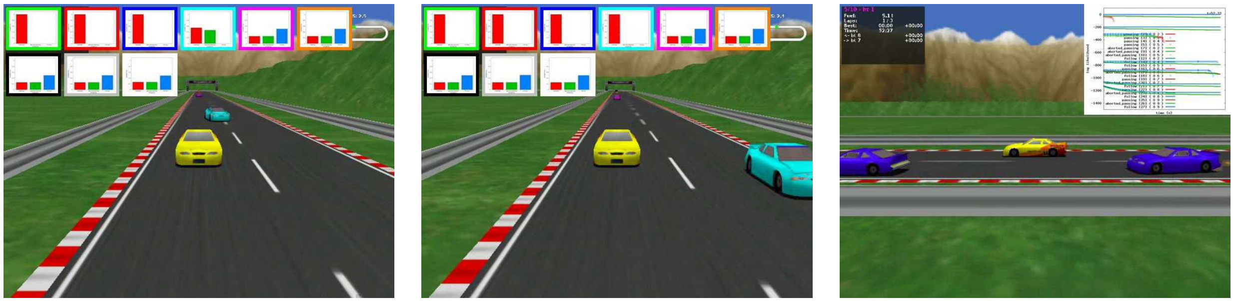

Fig. 4. The situation models were learned from a large set of simulated traffic scenarios involving multiple users using the open source driving simulator TORCS. Left: it is not yet clear whether the cyan car will actually pass the yellow one (see the forth bar chart on top). Middle: at this point, the passing maneuver was clearly identified. Right: Inspecting the evolution of situation likelihoods involving many agents.

as described above. The prior probabilities $P(\lambda)$ allow us to include information about how likely a given model is, prior to any evidence. We learn these from the training data using simple counting.

## V. Experimental Evaluation

Our framework has been implemented and tested in a vehicular traffic scenario using real data as well as a simulated driving environment. We considered three different situations that typically occur on a highway, namely *passing*, *aborted passing*, and *following*. The goal of the experiments was to show that our framework can be used to model and successfully recognize different situations instances in dynamic multi-vehicle environments.

For the experiments, the state $x_t$ of the state space model at time $t$ consisted of the relative distance $r_t^i$, relative bearing $\psi_t^i$, and relative speed $v_t^i$ of each surrounding vehicle $i$. It turned out that these features were sufficient to characterize the modeled situations, being also robust against variations in the different situation instances. In our experiments, each situation HMM consisted of 5 states with one three-dimensional Gaussian as observation model. Some HMMs could not be trained when using more states due to insufficient data, and using less states made the models too general and reduced their discriminative capacity. The initial estimates of the parameters $\pi$, $A$, and $B$ of the models where manually set by segmenting the situation stereotypical sequence into meaningful states. The Baum-Welch algorithm was then used to optimize the parameters so as to maximize the likelihood of the data.

### A. Situation Tracking

The goal of this experiment was to demonstrate that our approach can be used to successfully characterize and track different situation types. We first trained the different situation models using sequences generated in the simulation environment. The training data was generated using randomly selected speeds for the vehicles in different circuits. As test sequences, 10 passing and 10 aborted passing maneuvers were generated. Figure 3 plots the average log likelihood and standard deviation of the test sequences according to the different situation models. Note that in order to compare different instances of the same maneuver, the length of the sequences was first normalized, since different executions produce state sequences of different length.

After the approaching vehicle was detected, that is, when the approaching vehicle was within a 50 meter radius of the reference vehicle, we start computing the likelihood of the state sequence for the different situation models as described in Section IV. In the figures, it can be observed how the likelihood given a model measures how well the model explains the current state sequence. For the passing maneuvers (top plot) the *passing* situation model provides the best explanation compared with the other models. For the aborted passing maneuvers (bottom plot), however, the model does not perform as well.

For example, observe how at approximately 20% of the maneuver, as the passing vehicle starts changing to the left lane, the likelihood according to the *following* model starts to decrease. This occurs since the model expects the vehicle to remain behind and therefore ceases to provide an explanation for the observations. Similarly, at approximately 50% of the maneuver, as the passing vehicle is abeam, the likelihood according to the *aborted passing* model starts to decrease too.

The error bars in the figure capture the variance in the different executions of the maneuvers. However, this variance is greatly inflated by the normalization over the length of the sequences. This is why the bars appear so large. Specially after the likelihood of a sequence given a model falls below a certain threshold. In this experiment, we set the minimum allowed log likelihood value to -1500 for better visualization of the results.

### B. Tracking Multiple Situation Instances

A similar experiment was carried out with additional vehicles driving simultaneously to evaluate the performance and robustness of the framework. The scenario consisted of nine different vehicles passing the reference car. Every time one of the reference vehicles came within a 50 meter radius of the reference car, the three situation models where instantiated for that vehicle, and evaluated as described in Sec. IV. Once a vehicle was outside the tracking range, the associated models where discarded. The results (see Figure 4) showed that our approach can also be used in scenarios where multiple vehicles are being simultaneously tracked,


Fig. 5. Left: arrangement of two SICK laser range finders on a convertible (middle) used for gathering real data. Each laser has a field of view of 180 degrees and can detect objects as far as 80 meters with an angular resolution of 1 degree at 75Hz. The arrangement of the two lasers provided a 340 degree field of view (right).

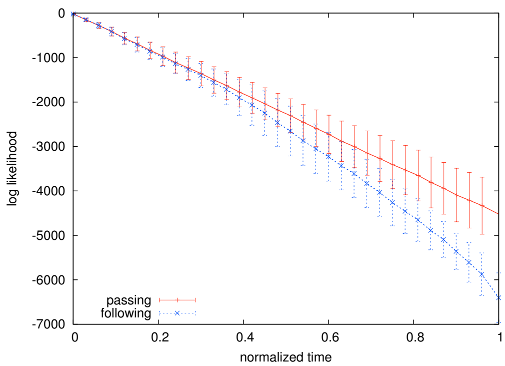


Fig. 6. Average likelihood of 5 observation sequences corresponding to a passing maneuver gathered from real data according to the *passing* and *following* situation models.

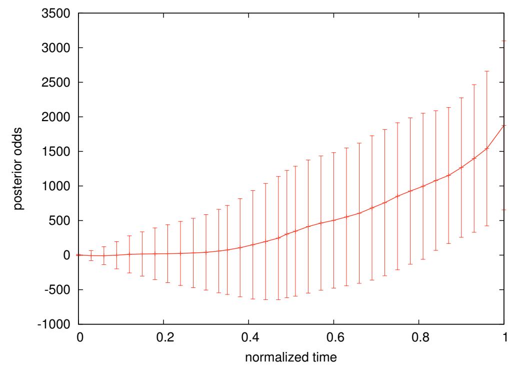


Fig. 7. Posterior odds in favor of the *passing* situation model compared against the *follow* situation model according to the observation sequence for the passing maneuver.

instantiating and eliminating multiple different instances of the situation models as vehicles appear and disappear from the state space, over extended periods of time. Since each situation instance is evaluated independently from the others, the complexity of the situation tracking algorithm increases only linearly in the number of situation types and vehicles in the state space.

### C. Real Data

The framework was also evaluated using real data. Two SICK laser range scanners were mounted on a convertible as illustrated in Fig. 5. Each laser has a field of view of 180 degrees and can detect objects as far as 80 meters with an angular resolution of 1 degree at 75Hz. The arrangement of the two lasers provided a 340 degree field of view as illustrated in the figure. Due to the blind spot in the field of view of the laser arrangement, states in which the vehicle was in front of the reference car could not be considered. Data was gathered by driving over more than 50 kilometers on highways and state roads at velocities of up to 110 km/h. Note that in this work, we concentrate only on the recognition of situation instances and do not deal with the tracking of the vehicles nor the state estimation problem. In order to extract the trajectories of the vehicles out of the real data, track initialization was done manually. After that, a Kalman filter was used for tracking the vehicles.

From the gathered data, only 14 and 8 complete tracks corresponding respectively to *passing* and *following* situations could be successfully extracted. Due to the technical limitations of the sensors together with their arrangement, many situation instances could not be captured, or were captured only partially in the data. Figure 6 plots the average log likelihood of 5 observation sequences corresponding to a passing maneuver according to the *passing* and *following* situation models trained with the real data.

### D. Model Selection

In this experiment we illustrate how the posterior odds can be used for selecting between different situation models. Figure 7 plots the posterior odds $P_{\lambda_p,\lambda_f}$ in favor of the *passing* situation model $\lambda_p$ compared against the *following* situation model $\lambda_f$ for the real data used in the previous experiment. The model priors $P(\lambda_p)$ and $P(\lambda_f)$ needed to compute the posteriors were obtained form the training data by counting the number of instances of the *passing* and *following* model.

A positive $P_{\lambda_p,\lambda_f}$ can be interpreted as evidence provided by the data in favor of the *passing* situation model. The motivation behind using the posterior odds as criterion for model selection can be observed in the results of the previous experiment, in which the likelihood of the sequence according to the model corresponding to the executed maneuver is generally higher than the likelihood according to the other models. Figure 7 also illustrates how the posterior odds can be used to make decisions between competing situation models as discussed in Section IV.

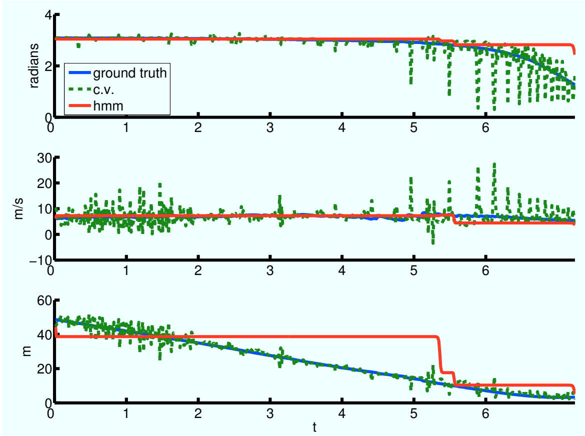


Fig. 8. 1-second prediction for a sequence corresponding to a passing vehicle according to a constant velocity model (c.v.) and our learned HMM-based *passing* situation model (hmm).

### E. Prediction

In this experiment we demonstrate how our learned models can be used for predicting the state of the system. State prediction within the HMM framework consists of computing the belief state of the HMM as in (3) but without correcting for new evidence. The predicted state $\bar{x}_{t+\Delta}$ in the state space model can then be computed as

$$\bar{x}_{t+\Delta} = \frac{\sum_{i=1}^{N} \alpha_{t+\Delta}(i)\mu_i}{\sum_{i=1}^{N} \alpha_{t+\Delta}(i)}, \tag{6}$$

where $\mu_i$ is the mean of the multivariate Gaussian distribution of the observation model for state $q_i$ (see Section III). Figure 8 plots the predicted state of a vehicle for one of the sequences gathered from real data. The sequence corresponds to a passing maneuver and the learned *passing* model was used to obtain a 1-second prediction. Our HMM-based situation models allow us to describe and also predict situations with complex dynamics. However, it must be noted that the accuracy of the prediction depends strongly on the parameters of the model. For example, using a coarse discretization of the state space may lead to a model which can produce inaccurate predictions (see bottom plot). We also compare the prediction results of our models against a simple constant-velocity model. Such a model can be quite accurate for predictions when the situation is relatively linear but propagates errors in the state estimation into the predictions. Our HMM-based models, on the other hand, are robust against these errors since the predictions are based on the learned model and not only on the current state of the system.

## VI. Conclusions

In this work, we presented a general framework for modeling and recognizing situations. We take a model-based approach in which each situation type is described by an individual HMM that specifies the admissible state sequences corresponding to an instance of the given situation. We also demonstrated the usefulness of the posterior odds as criterion for selecting between two competing situation models.

The approach has been evaluated experimentally using real and simulated data in the context of a driver assistant application in traffic scenarios and with situations that typically occur in highway-like driving scenarios. The results demonstrate that our system is able to recognize and track multiple situation instances in parallel and to make sensible decisions between competing hypotheses. Additionally, we show that our models can be used for predicting the position of the tracked vehicles.

## VII. Acknowledgments

This work has been partly been supported by the Siemens AG, Corporate Technology, Intelligent Autonomous Systems, CT IC 6 and by the German Ministry for Education and Research (BMBF) through the DESIRE project.

## References

[1] J. McCarthy, "Situations, actions and causal laws." Stanford University, Tech. Rep., 1963.
[2] R. A. Kowalski and M. J. Sergot, "A logic-based calculus of events." *New Generation Comput.*, vol. 4, no. 1, pp. 67–95, 1986.
[3] P. Muller, "A qualitative theory of motion based on spatio-temporal primitives." in *KR*, 1998, pp. 131–143.
[4] D. A. Randell, Z. Cui, and A. G. Cohn, "A spatial logic based on regions and connection." in *KR*, 1992, pp. 165–176.
[5] F. Wolter and M. Zakharyaschev, "Spatio-temporal representation and reasoning based on rcc-8." in *KR*, 2000, pp. 3–14.
[6] Z. Manna and A. Pnueli, *The temporal logic of reactive and concurrent systems.* New York, NY, USA: Springer-Verlag New York, Inc., 1992.
[7] B. Bennett, A. G. Cohn, F. Wolter, and M. Zakharyaschev, "Multi-dimensional modal logic as a framework for spatio-temporal reasoning." *Appl. Intell.*, vol. 17, no. 3, pp. 239–251, 2002.
[8] A. Gerevini and B. Nebel, "Qualitative spatio-temporal reasoning with rcc-8 and allen's interval calculus: Computational complexity." in *ECAI*, 2002, pp. 312–316.
[9] J. F. Allen, "Maintaining knowledge about temporal intervals." *Commun. ACM*, vol. 26, no. 11, pp. 832–843, 1983.
[10] M. Ghallab, "On chronicles: Representation, on-line recognition and learning." in *KR*, 1996, pp. 597–606.
[11] H.-H. Nagel, "Towards a cognitive vision system." Institut für Algorithmen und Kognitive Systeme, Fakultät für Informatik der Universität Karlsruhe (TH), 2001, electronic publication.
[12] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition." in *Proceedings of the IEEE*, vol. 77 (2), 1989, pp. 257–286.
[13] M. Brand, N. Oliver, and A. Pentland, "Coupled hidden markov models for complex action recognition," in *CVPR*, 1997, pp. 994–999.
[14] Z. Ghahramani and M. I. Jordan, "Factorial hidden markov models," *Machine Learning*, vol. 29, no. 2-3, pp. 245–273, 1997.
[15] N. Landwehr, "Modeling interleaved hidden processes," in *Machine Learning, Proc. of the Twenty-Fifth Int. Conf. (ICML 2008)*, Helsinki, Finland, 2008, pp. 520–527.
[16] M. Bennewitz, T. Axenbeck, S. Behnke, and W. Burgard, "Robust recognition of complex gestures for natural human-robot interaction," in *Proc. of the Workshop on Interactive Robot Learning at Robotics: Science and Systems Conference (RSS)*, 2008.
[17] Dean and Kanazawa, "A model for reasoning about persistence and causation," *COMPINT: Computational Intelligence: An International Journal*, vol. 5, 1989.
[18] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Prentice-Hall, Englewood Cliffs, NJ, 2003.
[19] D. Meyer-Delius, C. Plagemann, G. von Wichert, W. Feiten, G. Lawitzky, and W. Burgard, "A probabilistic relational model for characterizing situations in dynamic multi-agent systems," in *Proc. of the 31th Annual Conference of the German Classification Society on Data Analysis, Machine Learning, and Applications*, 2007.
[20] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.