

Research Article

A Framework for Estimating Long Term Driver Behavior

Vijay Gadepally,¹ Ashok Krishnamurthy,² and Ümit Özgüner³

¹The Massachusetts Institute of Technology, Cambridge, MA, USA

²The Renaissance Computing Institute, Chapel Hill, NC 27517, USA

³The Ohio State University, Columbus, OH, USA

Correspondence should be addressed to Vijay Gadepally; vijay.gadepally@gmail.com

Received 11 July 2016; Revised 16 October 2016; Accepted 27 October 2016; Published 12 January 2017

Academic Editor: William H. K. Lam

Copyright © 2017 Vijay Gadepally et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present a framework for estimation of long term driver behavior for autonomous vehicles and vehicle safety systems. The Hybrid State System and Hidden Markov Model (HSS+HMM) system discussed in this article is capable of describing the hybrid characteristics of driver and vehicle coupling. In our model, driving observations follow a continuous trajectory that can be measured to create continuous state estimates. These continuous state estimates can then be used to estimate the most likely driver state using decision-behavior coupling inherent to the HSS+HMM system. The HSS+HMM system is encompassed in a HSS structure and intersystem connectivity is determined by using signal processing and pattern recognition techniques. The proposed method is suitable for a number of autonomous and vehicle safety scenarios such as estimating intent of other vehicles near intersections or avoiding hazardous driving events such as unexpected lane changes. The long term driver behavior estimation system involves an extended HSS+HMM structure that is capable of including external information in the estimation process. Through the grafting and pruning of metastates, the HSS+HMM system can be dynamically updated to best represent driver choices given external information. Three application examples are also provided to elucidate the theoretical system.

1. Introduction

At the Center for Intelligent Transportation (CITR) at The Ohio State University, we are interested in addressing the Cyber Physical Systems (CPS) related problem of *autonomous vehicles operating safely in mixed-traffic urban environments*. Cyber Physical Systems refer to the conjoining of a system's computational and physical elements [1]. Examples of Cyber Physical Systems include, but are not limited to, robotic surgery, automated traffic control, and autonomous vehicles. CPS has the ability to remove many of the expensive and dangerous hurdles required in common human tasks. One application area for CPS is in personal transportation through the development of autonomous or self-driving vehicles which have the potential to revolutionize transportation by removing humans from the driving loop. Autonomous vehicles have the ability to address some of the largest issues associated with personal and commercial transportation. Of course, as with any new technology, if there is to be widespread adoption of autonomous vehicles, it is important

to look at the possible adoption path and determine challenges with this adoption path.

The most likely path for widespread autonomous vehicle adoption is with the gradual inclusion of either fully or partially autonomous vehicles into the current transportation system. This is in line with recent developments of car manufacturers introducing vehicles with partial autonomous capabilities. Such an environment, in which there are both autonomous and human-driven vehicles operating together, is known as a mixed-traffic urban environment (MUE). In such an environment, autonomous vehicles will interact with human-driven vehicles in their regular operation. This interaction will require that autonomous vehicles have the ability to receive information about human-driven vehicles, either through vehicle-to-vehicle (V2V) communication or through onboard sensors. The autonomous vehicle must then be able to deduce the behavior of other vehicles based solely on the information received or measured. This ability is critical to avoid and/or mitigate the threat of traffic accidents.

Such requirements are also necessary in the development of vehicles with advanced safety systems. The ability of an autonomous vehicle, based on sensor or V2V input, determining the likely behavior of a human-driven vehicle is referred to as *driver behavior estimation*.

In our model, drivers make discrete decisions at a higher level that are then followed by the lower level vehicle's continuous trajectory. Consider the discrete decision a driver approaching an intersection may make to "Turn right." While the reasoning for this decision may be due many reasons, the vehicle follows a measurable continuous path based on this decision. In this example, the vehicle may reduce velocity, have negative acceleration, use a turn signal, and change the yaw and yaw rate and eventually perform a right turn at the intersection. Our work is on the estimation of the higher level driver state made from measurements of continuous observations made on the vehicle. For the purpose of this article, the term "driver" is used to describe the combination of human operator and vehicle, and the behavior exhibited by this combination is referred to simply as *driver behavior*. In order to accurately determine driver behavior, a framework capable of describing the qualitative and quantitative nature of driver behavior estimation is needed. Extending upon a previously designed system for driver behavior estimation is the main topic for this article.

In our previously developed framework, a Hybrid State System (HSS) representation is used to provide a qualitative view of how driver decision and vehicle dynamics are related. The HSS representation has been used in a number of studies [2–5]. In our application, we use the HSS as a mechanism for the modeling and estimation of driver behavior for autonomous vehicle applications. The HSS provides an intuitive architecture where the continuous state of a vehicle and discrete driver states are represented as two layers that interact to provide an overall view of how drivers and vehicles interact. A key element of this architecture is the ability to leverage the similarity of a vehicle model that is operating under different situations. The interrelation between the higher level driver state and lower level continuous measurements is done through tools borrowed from the signal processing and pattern recognition community.

Hidden Markov Models (HMMs) [6] are used to define how changes in a driver state correspond to changes in the continuous state of a vehicle.

The quantitative aspects (HSS system interconnections) of driver behavior estimation are provided by Hidden Markov Models (HMMs) [6]. HMMs are stochastic models that can be used to determine a relationship between relatively easy-to-measure observations (continuous vehicle trajectory in our application) and the hidden states that are said to generate these observations (discrete driver states in our application). While HMMs have had considerable success in fields such as speech recognition [7], they have also been used for estimating human behavior [8] and previously in recognizing driving events [9]. In order to best model the vehicle's continuous state, we employ Gaussian Mixture Models (GMMs) for the learning of the HMMs. Estimating the driver state given a set of continuous observations is done through a pattern

recognition technique that finds the most likely driver state given a set of continuous observations. Overall, the HSS provides the qualitative architecture of the driver and vehicle coupling and HMMs define the quantitative relationship between the two layers of the architecture. We refer to this HSS and HMM combination as *HSS+HMM*.

In [10], we presented results obtained for driver behavior estimation using the HSS+HMM system. The system presented in that article performs accurate driver behavior estimation around intersections. This system was expanded to include lane change maneuvers and highway events to better describe the long term behavior of a vehicle. For the purpose of this article, we define "long term behavior" to be extended periods of driving that may include a vehicle operating at a combination of intersections, highways, and arterial roads. Previous studies such as [11–13] have only looked at certain aspects of long term driver behavior.

This article begins with an overview of the HSS+HMM system, in which we discuss the developed system along with the data collection experiment. Section 3 discusses the current system implementation along with limitations that hamper estimating long term driver behavior. Section 4 discusses a proposed theoretical extension to the HSS+HMM system along with definitions. Finally, we discuss three example implementations of the proposed system, discuss avenues for future work, and conclude the article.

2. The HSS+HMM System

The HSS+HMM system refers to the conjoining of two models, Hybrid State Systems and Hidden Markov Models. The HSS is used to describe the interaction between continuous observations made on a vehicle and the discrete set of driver decisions that generated these observations. The HSS consists of two layers. The higher layer is a discrete state system (DSS) which encapsulates discrete driver decisions; the lower layer is a continuous state system (CSS) which encapsulates continuous observations on a vehicle. This architecture can be seen in Figure 2. In our model, we assume that a driver operates within the space of states in the DSS and based on a decision made at this higher level the vehicle follows some observable continuous trajectory. Other studies such as [14] have looked at modeling systems as a HSS for applications such as autonomous vehicles. States of the DSS are modeled as a finite state machine (FSM). In our application, we assume that continuous state estimates (CSS state estimate) can be made through vehicle-to-vehicle (v2v) or onboard sensors.

Given a sequence of CSS state estimates taken from onboard sensors, determination of the higher level DSS state is done by using Hidden Markov Models and pattern recognition techniques. Using the data collection process described in Section 2.1, numerous HMMs are trained using the Baum Welch method. Details of this process are described in [10]. Essentially, we first train a number of HMM models, $\lambda_1, \lambda_2, \dots, \lambda_n$, that correspond to n different vehicle action. We then use the forward algorithm to compute the probabilities $P(O | \lambda_i), i = 1, 2, \dots, n$, for an observation sequence $\mathbf{O} = \{o_1, o_2, \dots, o_t, \dots, o_T\}$. A suitable DSS metastate is estimated

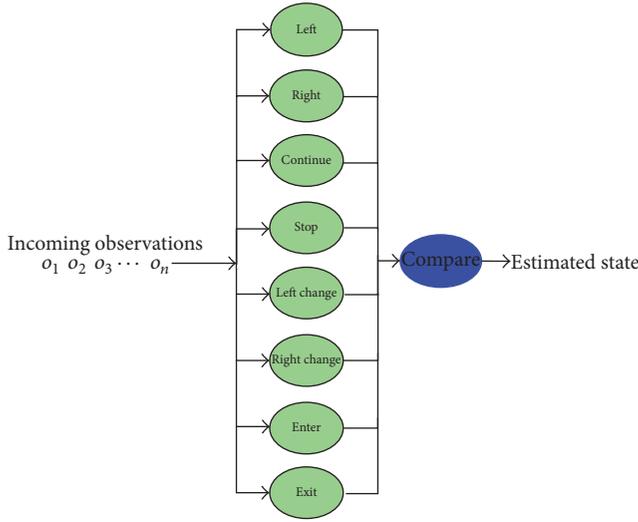


FIGURE 1: Graphical representation of possible event metastates and incoming CSS state estimates. Output estimated state corresponds to HSS+HMM state estimate.

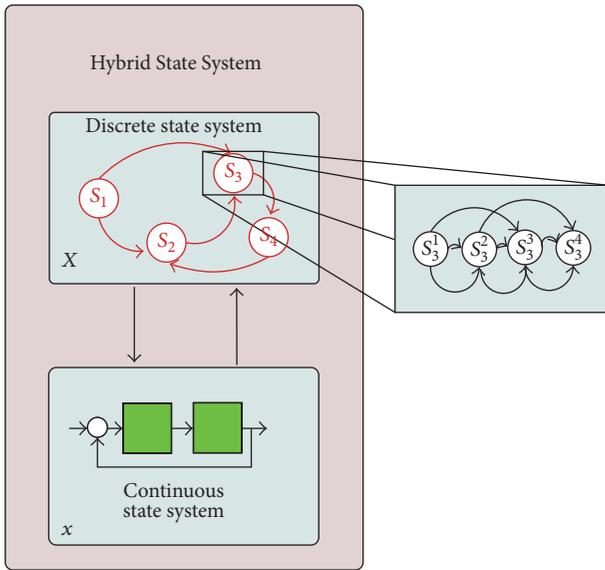


FIGURE 2: Overall HSS+HMM model. The left side of the figure describes the overall architecture of DSS and CSS. The right side indicates how each metastate of the DSS is modeled as an HMM.

by determining the probabilities $P(O | \lambda_i)$, $i = 1, 2, \dots, n$, for each λ_i . The model, λ_i , that maximizes the likelihood probability at a given time step is said to be the estimated DSS state:

$$\text{State}(t) = \arg \max_i P(o_1 \lambda_i o_2 \dots o_t) \quad i = 1, \dots, n. \quad (1)$$

This process is summarized in Figure 1. Incoming observations are measured by onboard sensors and the metastate estimate corresponds to different vehicle actions. Equation (1) is used for the final compare step.

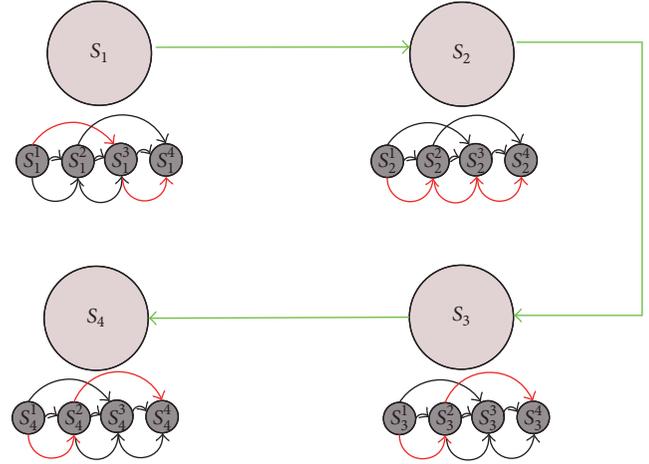


FIGURE 3: A simple 4-metastate DSS transition. In this example, the green lines correspond to metastate transitions and the red lines correspond to potential underlying state transitions.

In our model, the DSS of the HSS contains states that represent driver decisions. Metastates of the DSS are related to continuous observations via a number of HMMs. Determining which HMM most closely relates the CSS estimates with DSS estimate is done through (1). We refer to states of the DSS as *metastates*. Each *metastate* is itself an HMM that corresponds to a driver decision of interest. The right side of Figure 2 describes a situation where the S_3 metastate is an HMM with 4 hidden states ($\{S_3^1, S_3^2, S_3^3, S_3^4\}$).

Transitions between metastates of the DSS are determined by applying (1) to a given observation sequence. For the exemplary case in the introduction of a vehicle turning right at an intersection, the observation sequence may consist of multidimensional measurements of velocity, acceleration, yaw, yaw rate, and so forth. As an observation sequence is measured, we can determine the most likely DSS metastate (corresponding HMM states). As a very simple example, consider the DSS of Figure 3 that consists of four states $\{S_1, S_2, S_3, S_4\}$. The green lines of the figure correspond to metastate transitions and the red lines indicate possible transitions of the underlying HMM describing that particular metastate. In this example, traversing metastates in the order $S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4$ actually results in the transition of underlying states:

$$S_1^1, S_1^3, S_1^4, S_2^1, S_2^2, S_2^3, S_2^4, S_3^1, S_3^2, S_3^3, S_3^4, S_4^1, S_4^2, S_4^4. \quad (2)$$

Using the HSS+HMM system for real driving scenarios requires that there are a large number of trained DSS metastates. Further, we can use this training to determine the relationship between metastates. Learning the transition probabilities between metastates can then be used to predict future driver state using algorithms such as the Viterbi algorithm [17]. In the following section, we provide a brief overview of the data collection process.



FIGURE 4: OSU CITR vehicles in a convoy. The vehicle on the right was instrumented with a number of sensors and driven by study participants.

2.1. Data Collection. Training the metastates of the HSS+HMM model requires access to data of drivers operating vehicles under normal driving conditions. For the purpose of this study, we assume that any data we are leveraging for generating the CSS estimate could be measured using v2v or other onboard sensors a vehicle may have. Recruited drivers were asked to drive a 2012 Honda Accord (Figure 4) along two predetermined paths in Columbus, OH. Each path was approximately 25 miles in length and took approximately 40 minutes to complete. Minor adjustments to the route were made with each participant to allow for breaks or traffic or construction.

Our sensor fitted vehicle had a number of sensors:

- (i) *Novatel GPS Unit.* This sensor provides GPS latitude, GPS longitude, timestamp of reading, altitude, latitude and longitude estimated standard deviation, horizontal speed, and track over ground (yaw).
- (ii) *Honda Accord CAN Bus.* This sensor provides timestamp of reading, yaw rate, lateral acceleration, throttle pedal position, brake pressure, steering wheel angle, speed, engine RPM, torque converter RPM, odometer, headlights, brake lights, throttle peddle, gear, wiper, turn signals.
- (iii) *Three HD cameras.* They provide views of the front, left side, and right side of the vehicle.

Data collected from these sensors are fused into relevant feature vectors which are then used for model training. We concentrate on events of interest that occur near intersections, roads, and highways. For this purpose, we have limited the models to the following events:

- (1) Left Turn
- (2) Right Turn
- (3) Straight/Continue
- (4) Stop
- (5) Left Lane Change
- (6) Right Lane Change
- (7) Enter Highway
- (8) Exit Highway

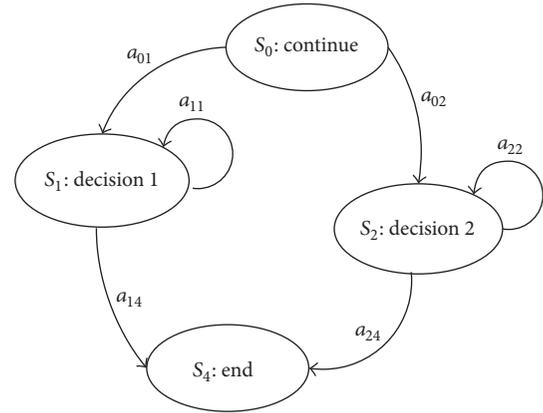


FIGURE 5: Simple DSS model with two possible metastates. “Continue” metastate refers to vehicle proceeding with no change. S_1 and S_2 could correspond to driver decisions such as stop and turning right. Notional metastate transition probabilities are denoted.

Additional experimental data was collected that validated the HSS+HMM system with respect to naturalistic driver behavior estimation. Details of the experiment and results are available in [10].

3. Long Term Driver Behavior

Consider the HSS+HMM metastate estimation stage for driver behavior estimation described in the previous section. Previous work on driver behavior estimation has concentrated on estimating the short term behavior (i.e., during a particular event or at a particular location). In many cases, there is a need to estimate or track a full vehicle driving sequence. In a full driving sequence, a driver goes through a series of events to accomplish a goal often through changing scenarios. For example, to drive to work from home, a typical sequence of driving events may be as follows: move back out of the garage, drive through city streets for a few miles, take a limited access highway for several miles, take an exit, drive through city streets, turn into a parking garage, and finally park.

3.1. Long Term Behavior Estimation Limitations. HSS+HMM system provides very good driver behavior results when compared to other possible systems. One of the limitations of the HSS+HMM system is in describing the long term behavior of a vehicle. This limitation is largely due to two reasons: (1) metastate likelihood probabilities reduce for long observation sequences and (2) the HSS+HMM system does not include external information into the estimation process.

3.1.1. Diminishing Likelihood Probabilities. The first limitation can be understood through a simple example. Consider the hypothetical DSS FSM shown in Figure 5. In this example, the DSS contains four possible metastates. The transition probabilities between metastates are given by the indicated

values. These metastates can refer to any vehicular event of interest such as turning, going straight, and changing lanes. The metastate transition matrix, A , can be expressed in matrix form as

$$A = \begin{bmatrix} 0 & a_{01} & a_{02} & 0 \\ 0 & a_{11} & 0 & a_{14} \\ 0 & 0 & a_{22} & a_{24} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3)$$

Now consider a long observation sequence in which the system enters state S_1 or S_2 for an extended period of time. In such a case, the transition probabilities (after n observations) become

$$A^n = \begin{bmatrix} 0 & a_{01} & a_{02} & 0 \\ 0 & a_{11} & 0 & a_{14} \\ 0 & 0 & a_{22} & a_{24} \\ 0 & 0 & 0 & 1 \end{bmatrix}^n = PD^nP^{-1}, \quad (4)$$

where P is a 4×4 matrix whose columns contain the eigenvectors of A and D is a diagonal matrix with the corresponding eigenvalues of A . For the above exemplary values, the eigenvalues of A are $a_{00}, a_{11}, a_{22}, 1$. With these eigenvalues D^n becomes

$$D = \begin{bmatrix} a_{00}^n & 0 & 0 & 0 \\ 0 & a_{11}^n & 0 & 0 \\ 0 & 0 & a_{22}^n & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (5)$$

Since a_{00}, a_{11}, a_{22} are all transition probabilities that are less than 1, for a large n , $a_{11}^n, a_{22}^n \rightarrow 0$ and the system will eventually transition from state S_1 or S_2 to state S_4 regardless of the observed CSS estimate. Given driving patterns which often consist of driving straight for a long period of time (think of highway driving), the unintended transition out of a metastate may occur for one in the *straight* metastate.

3.1.2. Including External Information. To understand the second limitation, consider the HSS+HMM metastate estimation stage. In this stage, continuous observations of a vehicle are compared with number of competing models (HMM metastates) to determine the most likely driver behavior metastate in the DSS that caused the continuous observed vehicle trajectory. Without additional information, the metastates used for estimating driver behavior are the same as those used for highway driving which are the same as those used for nonintersection road driving, as shown in Figure 1. As one may readily understand, the driver intention metastates used for an intersection may not be appropriate when driving on a road far away from any intersections. Fundamentally, an ideal driver behavior estimation system should be able to make use of external information in order to describe a full driving sequence. At any given time, only a limited set of actions, and thus driver behaviors, are valid. For

example, when driving on a limited access highway, making a left turn may not be possible, and when driving on a city street, exiting a highway may not be possible. Including higher level information about the surroundings in the decision making process may be able to provide insight into what is possible at a given time.

3.2. Overcoming These Limitations. The limitations discussed in the previous section can be overcome by intelligently limiting the length of sequences and including information from external sources in the driver behavior estimation process. There are a number of methods one can use.

One possible method to describe such a full driving sequence is to build some sort of *all-encompassing* metastate that contains all possible driver states under all scenarios and let a system similar to Figure 1 estimate driver behavior. Such a method will likely struggle with the drawbacks discussed with learning based classifiers in which training data needs to be carefully selected and long sequences need to be manually truncated. While such a technique may provide a solution to the first problem, it does not allow us to include external information in the driver behavior estimation process.

Another method to describe a full driving sequence could be to dynamically modify the metastates that make up the DSS. The system can also be designed to intelligently modify the observation likelihood probabilities based on external information.

Dynamically modifying the metastates of the DSS requires the ability to add or remove metastates. The aim of adding or removing metastates from the DSS is to define a DSS that better defines the metastates required to represent vehicle events given current conditions. For example, a DSS corresponding to highway driving should include metastates Exit Highway and Enter Highway, whereas a DSS corresponding to near-intersection driving will not require these metastates but instead require metastates corresponding to Left and Right Turns. At any time where the DSS changes, we can modify the likelihood probabilities.

There are numerous scenarios in which DSS modification is necessary and the driver behavior estimation system should be capable of dynamically changing the metastates that comprise the DSS.

4. Dynamically Modifying the DSS

Dynamic discrete state system (DSS) modification provides the HSS+HMM system with the ability to dynamically change the metastates that comprise the DSS. Recall that possible driver decisions are encapsulated by metastates in the DSS. In order to dynamically modify the DSS, the operations required for such modifications are adding and removing metastates. Adding a metastate to the DSS is known as *Metastate Grafting*. Conversely, removing a metastate from the DSS is known as *Metastate Pruning*. The ability to graft or prune a metastate from the DSS provides the HSS+HMM system with the ability to overcome the limitations mentioned in the previous section.

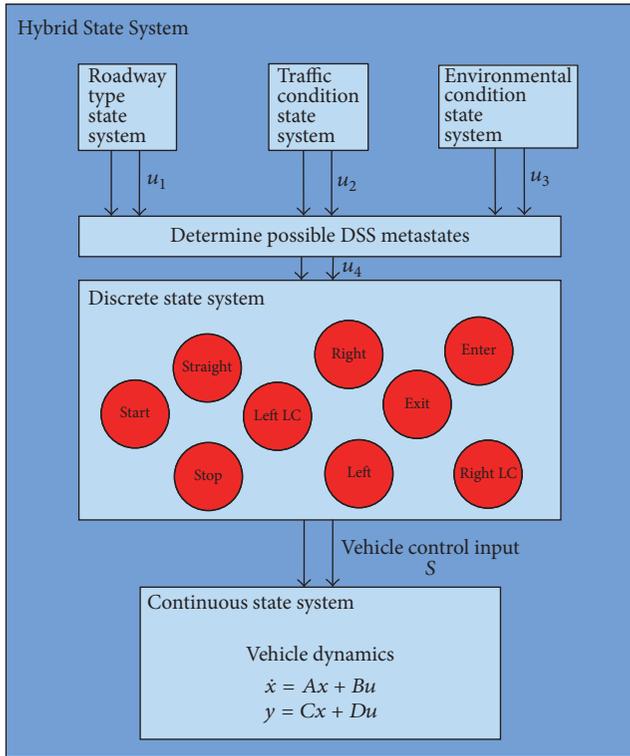


FIGURE 6: Extended HSS+HMM architecture. Observations measured on a vehicle are the result of driver decisions, road conditions, environmental conditions, and traffic conditions.

At each graft or pruning, the FSM that represents DSS metastate interconnections is changed to account for the additional or lesser metastates. Grafting and pruning of a metastate can occur at any given time but for the purposes of this study it is limited to instances when there is a change in external factors.

As a driver readily understands, different inputs can imply very different actions under different situations. For example, at an intersection, a right turn signal indicator (in addition to other observations), may lead to a Right Turn state estimate. On the other hand, a right turn signal indicator (in addition to other observations) on a highway may lead to a Right Lane Change estimate or, if one is near an exit, perhaps an Exit Highway estimate. This estimation may further change with differing traffic and environmental conditions, even when given similar observation sequences. This requires that the HSS+HMM framework be expanded to make use of such information such as the modified architecture in Figure 6.

The original formulation of the HSS, the property that discrete driver states of the DSS generate a continuous trajectory which can be observed through easy-to-measure CSS state estimates, is still maintained. As previously described, the discrete state system consists of metastates which encompass HMMs that represent driver events of interest. The CSS represents the vehicle dynamics that, given the input of the DSS, generates the observations that decisions are based on. Previously, the discrete driver decision was based entirely on the actions or behavior of a human driver. As depicted in Figure 6, in order to represent a driver's use of external

information in the decision making process, new system interconnections to the HSS depict external factors that may play into a drivers decision. Specifically, the three state systems have been added:

- (i) *Roadway Type Condition State System (RCS)*: high level information that describes the road. Roadway type states may include “highway,” “intersection,” and “nonintersection.”
- (ii) *Environmental Condition State System (ECS)*: high level environmental information. Environmental condition states may include “icy roads,” “mountainous roads,” and “wet roads.”
- (iii) *Traffic Condition State System (TCS)*: information about current or future traffic patterns, congestion, and so forth.

Each of these state systems is represented by a FSM such as that in Figure 5. State transitions within these FSMs are governed largely by external information such as GPS coordinates, traffic information, and/or weather information. While there are many other possible sources for external information, we believe that a majority of the information we process can be contained within these systems.

With the extended architecture, observations of the CSS state are still related to the DSS estimate through the HSS+HMM system. The new state system additions reflect that driver behavior metastates of the DSS can be affected by these state systems. Further, changes in the states of any of these external state systems allow DSS metastates to be change dynamically with changing external conditions. With this extended system, the estimated driver behavior is determined by observations from the CSS, and in determining likely driver decision information is included from the RCS, ECS, and TCS.

4.1. Definitions. With access to high level information (e.g., RCS states can be determined using GPS position), one can dynamically determine a relevant DSS configuration. Given a particular set of states in the RCS, ECS, and TCS a suitable DSS configuration can be used. A DSS configuration is defined to include the following:

- (1) Metastate list
- (2) Metastate FSM (i.e., transition probabilities and metastate interconnections)
- (3) External information FSM

With this definition, for changes in external conditions, the DSS dynamically changes to model a driver's decision making process. The metastate list represents possible driver behavior events of interest. Consider the DSS, \mathbf{S} , which contains metastates $S_1, S_2, S_3, \dots, S_n$. For an instance of the DSS \mathbf{S} and metastates S_i , we have the following definitions:

$$\begin{aligned} \mathbf{S} &\ni \{S_1, S_2, S_3, \dots, S_n\} \\ |\mathbf{S}| &= |\{S_1, S_2, S_3, \dots, S_n\}| = n, \end{aligned} \quad (6)$$

where $|\mathbf{S}|$ is defined size of \mathbf{S} (number of metastates in \mathbf{S}).

The task of modifying DSS metastates is referred to as *metastate modification*. Situation in which a metastate is added to the DSS is referred to as a *Metastate Grafting*. Situation where a metastate is removed from the DSS is referred to as *Metastate Pruning*.

In this context, we can now mathematically define the two types of metastate modifications: Metastate Grafting and Metastate Pruning.

4.1.1. Metastate Grafting. In a metastate graft, a metastate is added on to the DSS. Let the DSS, S , at time t be defined as $S(t) \ni \{S_1, S_2, \dots, S_n\}$, where n is the number of metastates in the DSS at time t . At time t , the size of the DSS $|S| = n$. Now, after τ time, suppose that the road, environment, and/or traffic conditions change such that a new metastate S_{n+1} needs to be grafted into the DSS. Now, $S(t+\tau) \ni \{S_1, S_2, \dots, S_n, S_{n+1}\}$.

Formally, the DSS S , is said to have been grafted on, if, for S' defined as $S \xrightarrow{S_{n+1}} S'$, $|S'| = |S| + 1$. Also, since a state has been grafted onto S , $S' \supset S$.

Additionally, if it is necessary to graft more than one state simultaneously, that is the same as serially grafting each individual state. For example,

$$S \xrightarrow{S_j, S_k} S' \equiv S \xrightarrow{S_j} S' \xrightarrow{S_k} S' \quad (7)$$

Consider a simple case where a vehicle is driving on a straight one-lane road. In this case, the DSS may contain only one metastate, "Continue." Now, if based on a change in the Road Condition State such as a change from a one-lane road to a two-lane road, two new metastates will be grafted onto the DSS, "Left Lane Change" and "Right Lane Change." Similarly, if the road condition changes to an intersection, two additional metastates, "Left Turn" and "Right Turn," can be grafted.

4.1.2. Metastate Pruning. In Metastate Pruning, a metastate is removed from the DSS based on changes in road, environment, and/or traffic conditions. Let the DSS S at time t be defined as $S(t) \ni \{S_1, S_2, \dots, S_n\}$ where n is the number of metastates in the DSS at time t . At time t , the size of the DSS $|S| = n$. Now, after τ time, suppose that the road, environment, and/or traffic conditions change such that metastate S_n needs to be pruned from the DSS. Now, $S(t+\tau) \ni \{S_1, S_2, \dots, S_{n-1}\}$.

Formally, the DSS S is said to have been pruned, if, for S'' defined as $S \xrightarrow{S_n} S''$, $|S''| = |S| - 1$. Also, since a state has been pruned from S , $S'' \subset S$.

Additionally, if it is necessary to prune more than one state simultaneously, that is the same as serially pruning each individual state. For example,

$$S \xrightarrow{S_j, S_k} S'' \equiv S \xrightarrow{S_j} S'' \xrightarrow{S_k} S'' \quad (8)$$

Consider a simple case where a vehicle is approaching an intersection. At this stage, the DSS metastates may include "Stop," "Continue," "Left Turn," and "Right Turn." After passing the intersection, the vehicle road condition changes to a single lane road and metastates "Left Turn" and "Right Turn" are pruned from the DSS.

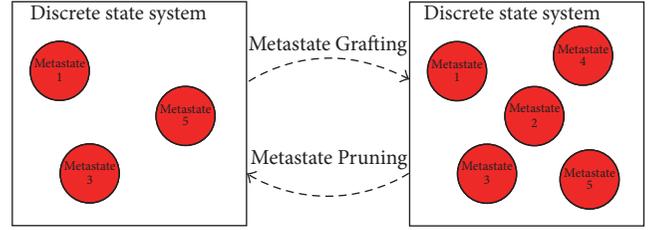


FIGURE 7: Example of metastate modification: Left \rightarrow right shows Metastate Grafting. Right \rightarrow left shows Metastate Pruning.

4.2. Procedure for Grafting and Pruning. Based on the current state or change of state of the Roadway Type System (RCS), Environment Condition System (ECS), and Traffic Condition System (TCS), a procedure for grafting and pruning the metastates needs to be defined. For the following examples, we have assumed that the ECS and TCS states are constant.

Let the state of the RCS at time t be $RC(t)$. Now, suppose that at time t' external information (such as change in GPS position) is received such that the state of the RCS becomes $RC(t')$. Additionally, suppose that the DSS at time t is $S(t) = \{S_1, S_2, \dots, S_n\}$.

If the change in the RCS requires additional state or states to be added to $S(t)$ to determine $S(t')$, this is represented as follows:

$$S(t) \xrightarrow{\text{graft}_{\text{metastates}}} S(t'). \quad (9)$$

A graft effectively increases the number of possible vehicle events (represented by metastates). Similarly, if the changes in RCS require the removal of a state or states from $S(t)$ to determine $S(t')$, this is represented as

$$S(t) \xrightarrow{\text{prune}_{\text{metastates}}} S(t'). \quad (10)$$

In this case, the pruning effectively decreases the number of possible vehicle events. There may be many cases in which $S(t)$ needs more than one graft and/or pruning. Suppose that it is necessary to graft S_{n+1} and prune S_n ; this is done through a graft followed by pruning on S .

$$S(t) \xrightarrow{S_{n+1}} S'(t') \xrightarrow{S_n} S(t'), \quad (11)$$

where $S'(t') \ni \{S_1, S_2, \dots, S_{n-1}, S_n, S_{n+1}\}$ and $S(t') \ni \{S_1, S_2, \dots, S_{n-1}, S_{n+1}\}$. With these definitions, given suitable metastates, through the process of grafting and/or pruning various metastates, one can define a DSS that contains any combination of metastates.

Using Figure 7 as an example, suppose that the DSS on the left is called S_{left} and the DSS on the right is called S_{right} . If the current DSS is S_{left} and changes to external conditions require modifying the DSS to S_{right} , the following operations will need to occur:

$$S_{\text{left}} \xrightarrow{\text{graft}_{\text{metastate2}}} S_{\text{left}} \xrightarrow{\text{graft}_{\text{metastate4}}} S_{\text{right}} \quad (12)$$

Similarly, in the reverse situation,

$$S_{\text{right}} \xrightarrow{\text{prune}_{\text{metastate2}}} S_{\text{right}} \xrightarrow{\text{prune}_{\text{metastate4}}} S_{\text{left}} \quad (13)$$

The grafting and pruning operations defined in this section have been applied to long term driving examples.

5. Application Examples

Data collected as described in Section 2.1 was used to test the extended HSS+HMM system for long term driver behavior estimation. One of the motivations for introducing high level information contained in the RCS, ECS, and TCS is to allow for driver behavior estimation of multiple consecutive driver events. In this section, we look at the current implementation of the theoretical system described in Section 4. We then describe the results obtained of implementing this system on three example driving sequences. To view the second and third examples, you will need a browser capable of playing videos from the online video hosting service YouTube. In all the videos, the state and RC estimate correspond to the vehicle the camera is in.

5.1. Current Status and Implementation. A reverse geocoder, GeoNames (<http://www.geonames.org/>), is used to offline process the GPS coordinates. This allows us to automatically determine the roadway type state for intersections, nonintersections, and highways (reverse geocoding for highway segments requires significant manual corrections). *For the purpose of these examples, it is assumed that the ECS and TCS states remain unchanged* since this information is neither available nor used. Further, RCS states are limited to the following:

- (i) Intersection RC_{int} (corresponding to DSS S_{int}): this state describes a location close to an intersection (an intersection is defined to be the location of two roads intersecting). $S_{int} \ni \{S_{Straight}, S_{LeftTurn}, S_{RightTurn}, S_{Stop}\}$.
- (ii) Highway $RC_{highway}$ (corresponding to DSS $S_{highway}$): this state describes a vehicle operating on a highway or motorway. $S_{highway} \ni \{S_{Straight}, S_{LeftChange}, S_{RightChange}\}$.
- (iii) Nonintersection RC_{road} (corresponding to DSS S_{road}): this state describes a vehicle not described by RC_{int} or $RC_{highway}$. $S_{road} \ni \{S_{Straight}, S_{LeftChange}, S_{RightChange}\}$.

The grafting and pruning operations are also designed only to occur in situations in which there is a change in the external conditions. A change in external conditions may cause a metastate to be grafted or pruned (or both or any combination thereof) to (or from) the DSS. Whenever there is a change in the composition of the DSS, the current metastate is reset to allow the system to go into any of the available metastates. For example, if the system is in an HMM state of the ‘‘Straight/Continue’’ metastate and the RCS changes, this leads to metastates being grafted and pruned from the current DSS. At this point, regardless of current metastate, the system will step out of the current metastate and redo the procedure highlighted in Section 2 to select the most likely new metastate.

As further examples, consider a couple of special cases which are mentioned for grafting and pruning operations. In the first case, suppose that the system is currently in a metastate that due to external information has been pruned. In such a case, the system will immediately exit its current metastate and go to a default Continue metastate (which

is present in every possible DSS configuration) after which the procedure outlined before is followed. For the second case, suppose that our system is currently in a metastate when a new metastate is grafted. In this case, the system will essentially reset to allow the ongoing observation to be tested against other metastates including the newly grafted metastate(s).

A limitation of the current implementation of the system is that a change in the composition of the DSS that is, a metastate grafted or pruned causes the history of the system until that point to be reset as the system *jumping* out of the current metastate. In this reset there is useful information that relates to driving patterns or external conditions that may be lost. In order to extend the methodology presented, it will be beneficial to look at statistical information related to the RCS but not presented in this study. For example, collecting information on how RCS changes affects metastate selection.

Even with these limitations to the current implementation of the extended HSS+HMM system, application examples show promise for this theoretical framework.

5.2. Example 1. Using the data collected, an example case that contained multiple consecutive vehicle events was generated. The test observation sequence contains approximately 1900 individual observations captured at a rate of 10 Hz. This 2.75-minute observation sequence starts with a vehicle approaching an intersection and stopping at a stop light. After the light turns green, the vehicle proceeds towards the intersection and turns right. After turning, the vehicle continues straight until coming to a stop at another next intersection. Figure 8 shows the estimation results obtained for this observation sequence as a function of time. The top plot shows the estimated state at a given time step. The lower plot shows the Road Condition State at time t ; $RC(t) \in \{RC_{int}, RC_{highway}, RC_{road}\}$. The lowest part of the figure shows the DSS at time t .

For $RC(t) = RC_{int}$, $S(t) = S_{int} \ni \{S_{Continue}, S_{LeftTurn}, S_{RightTurn}, S_{Stop}\}$. For $RC(t) = RC_{road}$, $RC_{road} \ni \{S_{Continue}, S_{LeftChange}, S_{RightChange}\}$.

In Figure 8, the grafting/pruning operations when the RCS changes from road to int are

$$\begin{aligned}
 S_{road} &= \{S_{Continue}, S_{LeftChange}, S_{RightChange}, S_{Stop}\} \\
 S_{int} & \\
 &= S_{road} \xrightarrow{\text{prune}}_{LeftChange} \xrightarrow{\text{prune}}_{RightChange} \xrightarrow{\text{graft}}_{Right} \xrightarrow{\text{graft}}_{Left} S_{int}.
 \end{aligned} \tag{14}$$

When the RCS changes from int to road the grafting/pruning operations are

$$\begin{aligned}
 S_{road} &= \{S_{Continue}, S_{LeftTurn}, S_{RightTurn}, S_{Stop}\} \\
 S_{road} & \\
 &= S_{int} \xrightarrow{\text{prune}}_{LeftTurn} \xrightarrow{\text{prune}}_{RightTurn} \xrightarrow{\text{graft}}_{RightChange} \xrightarrow{\text{graft}}_{LeftChange} S_{road}.
 \end{aligned} \tag{15}$$

5.3. Example 2. Figure 9 gives the results for another example of metastate modification being used in conjunction with the

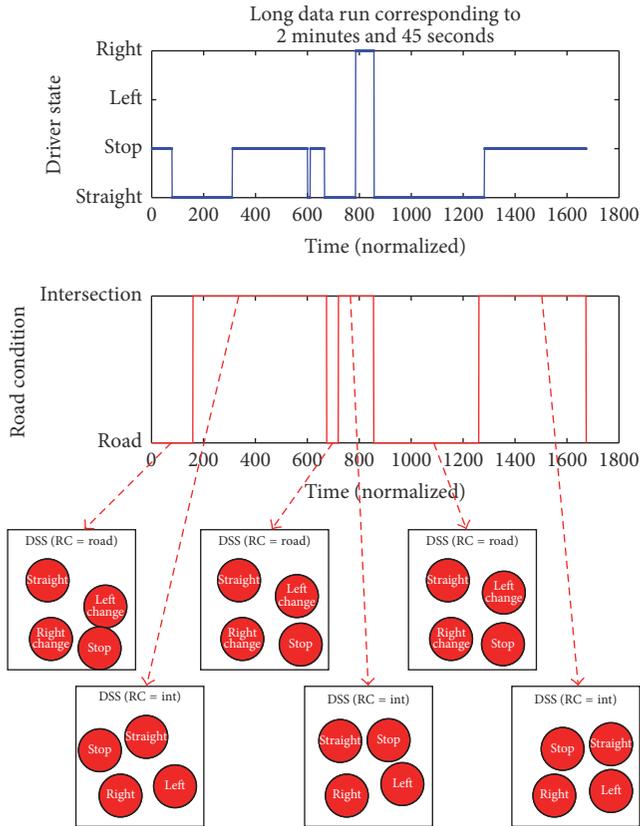


FIGURE 8: Metastate modification in conjunction used for observation sequence of length of 2.75 minutes. For varying road conditions (middle plot) metastates contained in DSS are shown in lowest part of figure.

HSS+HMM to estimate driver behavior. This observation sequence corresponds to approximately 6.5 minutes of city driving. In this time, the driver approaches approximately 10 intersections, which are highlighted in the lower plot of vehicle trajectory using Google Earth. The upper plot, similar to Figure 8, describes the vehicle state at a given time. The lower plot shows the Road Condition State for the vehicle at a given time. The lower figure shows the vehicle trajectory (white line) of the driver performing the 6.5-minute drive around Columbus, OH. The purple/blue arrows illustrate coordinates where the RCS is intersection.

The real-time driver behavior estimation of the sequence of driver behavior depicted in Figure 9 can be viewed online by using the link of Figure 10. The online video provides the front camera output of the vehicle being driven on the path shown in the lower part of Figure 9. The top part of the video gives the current driver state (event) and road condition (RC) as 2-tuple. As the vehicle moves, these values are updated using procedures previously outlined. At the beginning of the video (approximately 0:07-0:10 minutes) the driving event is incorrectly shown to be straight, whereas the vehicle is clearly turning right. This difference is due to the RC being road in which there is no Right Turn metastate. This is one example of the need to have accurate RC, TC, and EC state information. The road condition changes from road to int when the

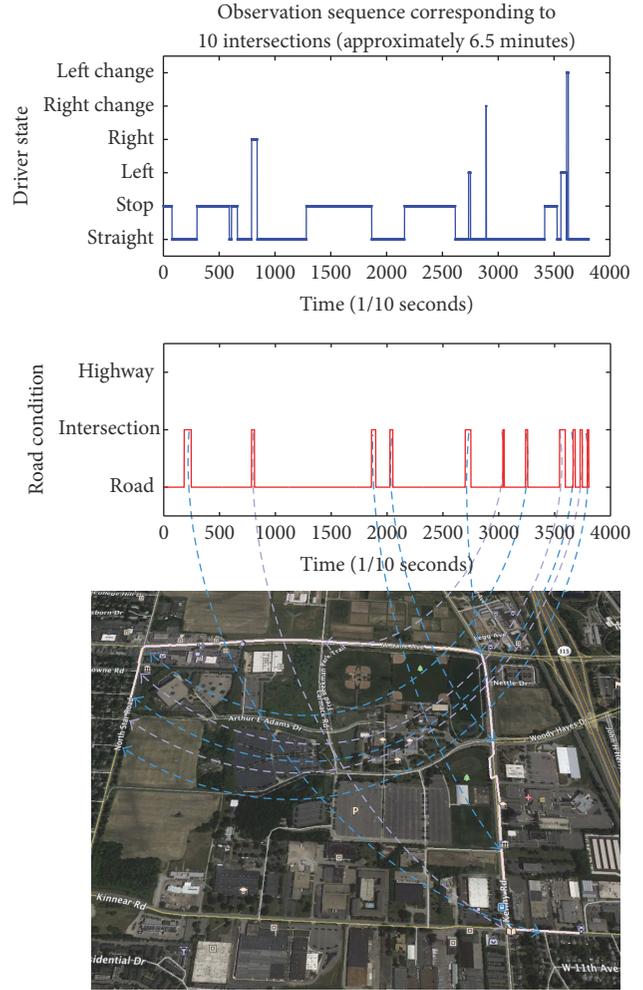


FIGURE 9: Example of driver behavior estimation using metastate modification. Blue and purple arrows highlight areas where the road condition changes.

distance to an intersection is approximately 100 feet. Starting at 1:56 minutes, a Left Lane Change maneuver is missed. This is largely due to the difficulties of estimating lane changes described in section. Apart from these two anomalies, the driver behavior estimation in the video very accurately estimates driver behavior for changing road conditions.

5.4. Example 3. The third example, which you can access by using the link in Figure 11, shows the HSS+HMM behavior estimates for a vehicle operating on the highway over approximately 3.5 minutes. In order to better represent events of interest, this video has been condensed to approximately 1.5 minutes showing only the noncontinuous events. In this time, the vehicle performs 4 Left Lane Changes and 3 Right Lane Changes. Some of the segments are prone to a slight delay in lane change recognition largely due to the challenges associated with lane change recognition discussed earlier in this article. The results are displayed in the video screen similar to the second example with the metastate estimate on the right and RC state estimate on the left.



FIGURE 10: YouTube link [15]: task grafting with changing road condition information.



FIGURE 11: YouTube link [16]: highway driving.

6. Discussion

In order to validate the results of the previous section, we used a mechanism similar to the one described in [10] to manually verify the results of the estimated state output. This manual system plays a video of the vehicle in motion and asks users to identify the start and end of an event of interest. These events of interest include left turns, right turns, stopping, and lane changes. These manual markings are then compared with the output of the HSS+HMM system described in this article. By viewing the video links provided in [15, 16], a reader can look at how close these results align with naturalistic driver estimation. As described in [10], in many cases, the HSS+HMM system estimates the “correct” state (as determined by a human observer) quicker than a human observer.

The three examples presented in this article highlight a number of challenges. First, state estimation that relies largely on continuous measurements of yaw and velocity can confuse lane changes and curves on a road or driver drifting. Our future work will include video lane markings

and turn signal indicators. Further, the HSS+HMM model presented in this article largely focuses on “good” drivers and can only estimate states based on previously observed data. For problematic drivers or unsafe road conditions, the output of our system may be incorrect. In-person driver data collection and associated safety of human subjects may not allow us to obtain training data for such conditions. We believe an interesting data collection source may be a driving simulator in which we can collect data on hazardous driving conditions or problematic driver behavior. Another challenge is in adding task grafting and pruning probabilities when transitioning between road and environmental condition changes. Ideally, each metastate grafted and pruned would have transition probabilities that vary probabilistically with combinations of road and environmental conditions.

The majority of the computation effort is taken by the training step. On a modern computer, the inference portion of the computation (estimating the state) can be done near real time. Utilizing hardware accelerators such as Graphical Processing Units (GPUs) [18] or FPGAs may potentially speed up both the training and inference computation. Further, we are looking at improving the computation performance using parallel processing on multicore or many processors as described in [19, 20]. Ideally, we would like to have an online training system that is able to take in data streaming from the sensors along with user feedback to further improve state estimation.

The results presented in this article can be applied to a variety of problems. While our discussion has largely focused on autonomous vehicles operating in a mixed urban environment, other application areas include driver safety systems and connected vehicles. The quality and speed of our state estimation techniques rely on accurate streaming sensor data and better vehicle-to-vehicle (v2v) or infrastructure-to-vehicle (i2v) communication [21] can improve quality. Future work may also incorporate more sophisticated models for traffic flow and road conditions [22].

7. Conclusions

Cyber Physical Systems have the ability to change human interaction with the world. One particular application area of Cyber Physical Systems, autonomous vehicles or self-driving cars, has the ability to revolutionize transportation and remedy many of the problems associated with the current state of personal and commercial transportation. In an expected mixed urban environment, it will be necessary for autonomous vehicles to be able to estimate the driver behavior of a human-driven vehicle. A system for estimating driver behavior has been described in [10] but is unable to track the long term behavior of a vehicle. The system proposed in this system extends upon the HSS+HMM system by including external information that shaped driver behavior by dynamically modifying the DSS. Through grafting or pruning operations, the extended HSS+HMM system is capable of emulating driver decisions in changing environments. An implementation of the system has been developed and three examples have been provided.

Disclosure

The work described in this article was performed at The Ohio State University, OH, USA.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This material is based upon work supported by the National Science Foundation Grant no. ECCS-0931669. The authors also thank Dr. Arda Kurt, Dr. Giorgio Rizzoni, Dr. Keith Redmill, Dr. Albert Reuther, and Honda R&D Americas, Ltd., for their help.

References

- [1] National Science Foundation, "Cyber-Physical Systems (CPS)," 2006, <https://www.nsf.gov/pubs/2010/nsf10515/nsf10515.htm>.
- [2] A. Kurt, J. L. Yester, Y. Mochizuki, and Ü. Özgüner, "Hybrid-state driver/vehicle modelling, estimation and prediction," in *Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems (ITSC '10)*, pp. 806–811, Madeira Island, Portugal, September 2010.
- [3] A. Kurt and Ü. Özgüner, "Hierarchical finite state machines for autonomous mobile systems," *Control Engineering Practice*, vol. 21, no. 2, pp. 184–194, 2013.
- [4] V. Gadepally, A. Kurt, A. Krishnamurthy, and Ü. Özgüner, "Driver/vehicle state estimation and detection," in *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC '11)*, pp. 582–587, IEEE, Washington, DC, USA, October 2011.
- [5] P. Liu, A. Kurt, and U. Ozguner, "Trajectory prediction of a lane changing vehicle based on driver behavior estimation and classification," in *Proceedings of the IEEE 17th International Conference on Intelligent Transportation Systems (ITSC '14)*, pp. 942–947, IEEE, Qingdao, China, October 2014.
- [6] L. Rabiner and B. Juang, "An introduction to hidden Markov models," *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986.
- [7] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [8] A. Pentland and A. Liu, "Modeling and prediction of human behavior," *Neural Computation*, vol. 11, no. 1, pp. 229–242, 1999.
- [9] D. Mitrović, "Reliable method for driving events recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 2, pp. 198–205, 2005.
- [10] V. Gadepally, A. Krishnamurthy, and U. Ozguner, "A framework for estimating driver decisions near intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 637–646, 2014.
- [11] T. M. Miller and D. H. Schuster, "Long-term predictability of driver behavior," *Accident Analysis & Prevention*, vol. 15, no. 1, pp. 11–22, 1983.
- [12] A. Várhelyi, M. Hjalmdahl, C. Hydén, and M. Draskóczy, "Effects of an active accelerator pedal on driver behaviour and traffic safety after long-term use in urban areas," *Accident Analysis & Prevention*, vol. 36, no. 5, pp. 729–737, 2004.
- [13] G. J. S. Wilde, "Social interaction patterns in driver behavior: an introductory review," *Human Factors*, vol. 18, no. 5, pp. 477–492, 1976.
- [14] A. Kurt and U. Ozguner, "Hybrid state system development for autonomous vehicle control in urban scenarios," in *Proceedings of the 17th World Congress, International Federation of Automatic Control (IFAC '08)*, pp. 9540–9545, Seoul, Korea, July 2008.
- [15] V. Gadepally, "6min30sec Observation sequence with road condition information," March 2013, <https://www.youtube.com/watch?v=RkuyYh0KSLU>.
- [16] V. Gadepally, "(2013, Mar. 17) Highway Driving-approximately 1 min 40 sec on SR315," YouTube video, 2013, <https://www.youtube.com/watch?v=IFfJyg-XvvY>.
- [17] G. D. Forney, "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.
- [18] L. Yu, Y. Ukidave, and D. Kaeli, "GPU-accelerated hmm for speech recognition," in *Proceedings of the 43rd International Conference on Parallel Processing Workshops (ICPPW '14)*, pp. 395–402, IEEE, Minneapolis, Minn, USA, September 2014.
- [19] S. Samsi, V. Gadepally, and A. Krishnamurthy, "MATLAB for signal processing on multiprocessors and multicores," *IEEE Signal Processing Magazine*, vol. 27, no. 2, pp. 40–49, 2010.
- [20] D. E. Hudak, N. Ludban, A. Krishnamurthy, V. Gadepally, S. Samsi, and J. Nehrbass, "A computational science IDE for HPC systems: design and applications," *International Journal of Parallel Programming*, vol. 37, no. 1, pp. 91–105, 2009.
- [21] M. Jerbi, P. Marlier, and S. M. Senouci, "Experimental assessment of V2V and I2V communications," in *Proceedings of the IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS '07)*, pp. 1–6, IEEE, Pisa, Italy, October 2007.
- [22] T.-Q. Tang, L. Caccetta, Y.-H. Wu, H.-J. Huang, and X.-B. Yang, "A macro model for traffic flow on road networks with varying road conditions," *Journal of Advanced Transportation*, vol. 48, no. 4, pp. 304–317, 2014.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

