2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
Banff Center, Banff, Canada, October 5-8, 2017

Driver Intention Estimation via Discrete Hidden Markov Model

Seifemichael B. Amsalu, Abdollah Homaifar and Ali Karimoddini
Electrical and Computer Engineering
Department
North Carolina A&T State University
Greensboro, NC 27411 USA
Email: samsalu@aggies.ncat.edu, {homaifar, akarimod}@ncat.edu

Arda Kurt
Electrical and Computer Engineering
Department
Ohio State University
Columbus, OH 43210 USA
Email: kurt.12@osu.edu

***Abstract*—In this paper, driver intention estimation near a road intersection is presented, using discrete hidden Markov models (HMM) and the Hybrid State System (HSS) framework as basis. The development of Advanced Driver Assistance Systems (ADAS) has assisted drivers in many driving scenarios and resulted in safe driving. Developing techniques to estimate driver's intention leads to the advancement of ADAS. As a large number of accidents occur near road intersections, estimating the intention of a driver at an intersection is vital. The methods developed in this paper can be applied in ADAS to take appropriate measures in reducing accidents. The driver decisions are depicted as a Discrete State System (DSS) at a higher level and the continuous vehicle dynamics are depicted as a Continuous State System (CSS) at a lower level in the HSS framework. In the proposed technique, the vehicle's continuous observations including speed and yaw-rate, are used to estimate the driver's intention at each time step. In this work, the speed and yaw-rate are discretized in such a way that the important features about the driver's intention such as "go straight," "stop," "turn right," or "turn left" at the intersection, are abstractedly represented in the form of symbols. Naturalistic driving data, which is collected using a vehicle fitted with sensors, is used to train and test the developed model. The results from the proposed approach show high accuracy in estimating the driver's intention at a road intersection.**

I. INTRODUCTION

A mixed-traffic environment arises due to the advent of autonomous vehicles where semi-autonomous, autonomous and human driven vehicles must collaborate for safe traffic flow. To achieve safe driving and road safety, effective prediction and estimation of the drivers' intention is required. The techniques developed to estimate drivers' intention are crucial in ADAS development for automated driving. The technological innovations in communication, computation and sensing provides the means to utilize the techniques developed for drivers' intention estimation in safety features such as cruise control and anti-lock braking systems in order to mitigate road accidents [1].

The goal of this work is developing driver intention estimation models to predict driver's state at road intersections. Ohio State University (OSU) researchers had proposed a framework for multi-agent model development [2], [3]. It simulates the drivers' attention, perception, control and cognition behaviors from the changing inputs of the environment by using different symbolic or mathematical methods [3], [4]. This paper proposes a technique that estimates the intention of a driver approaching a road intersection which is controlled by a traffic signal. The technique is evaluated using a naturalistic driving data. The objective of the work is to use a set of continuous observations of the vehicle such as its speed and yaw-rate to estimate the driver's intention to turn left, turn right, stop or go straight at a road intersection according to the traffic signal indicator. The variations in the observations that arise due to the drivers' different driving behavior are taken into consideration during the classification process. Here, the driver and vehicle combination is called as a "driver" and the vehicle trajectory resulting from the combination is called as the "driver behavior" [5]. The Hybrid-State System (HSS) framework [6] is the base of the overall work. In this work, the discrete hidden Markov model (HMM) is used as the mathematical method to relate the higher level discrete-states (driver decisions) and the lower level continuous-states (continuous observations) in the HSS framework.

In this study, estimation problem of a driver's intention near a road intersection is studied using discrete HMM and the HSS framework. HMM has been applied to many applications including speech recognition, bioinformatics, finance, computer vision, etc [7], [8], [9], [10]. Assuming, the driver's decisions that affect the vehicle trajectory are governed by the Markov process, HMM is used to represent the stochastic process that results the continuous vehicle observations using the Markov chain intuition. Here, the vehicle's continuous observations including speed and yaw-rate are discretized and combined into symbols to train the HMM. The HMM based on the Gaussian mixture model for continuous observations is presented in [11], [12] using the observations speed, yaw-rate and accelerations. However, in this work, we conclude that the use of acceleration to differentiate different driving maneuvers is not significant near a road intersection. The accuracy of the proposed approach in estimating the driver's intention is improved as compared to the work in [5], [11], [12].

The problem of driver behavior estimation has been studied extensively for different driving scenarios. Graphical models are developed for different driving scenarios with a focus on driver's performance in [13]. Experimental data is used to train the models and traffic contextual information is used to estimate the maneuvers. In [14], an HMM based method is proposed for cognitive model of human driving behavior during normal and emergency lane changes. The authors used a driving simulator data to train the models. In [15], discrete HMMs are used to recognize driving events. The authors trained the models using speed, lateral and longitudinal acceleration from a vehicle driven in an ordinary driving environment. In [11], driver intention estimation at a road intersection using HMM that uses a Gaussian Mixture Model (GMM) and Support Vector Machine (SVM) are proposed. The models are trained with naturalistic driving data. The authors extend their work to improve the recognition performance of the proposed approach using HMM trained with Genetic Algorithm (GA) in [12].

The organization of the paper is as follows. In Section II, the driver intention estimation problem and the proposed method is described. This section also briefly reviews the background on HSS and HMM. In Section III, the collection and analysis of the data are explained. The results of the proposed approach is presented in Section IV. Finally, the paper conclusion and future work is presented in Section V.

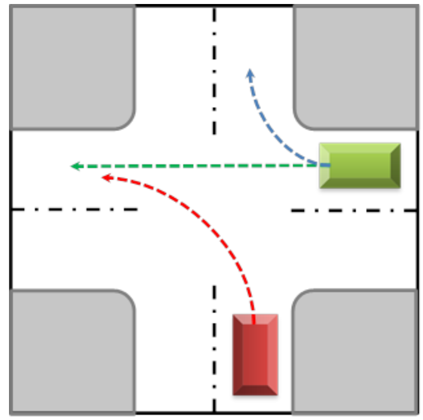


Fig. 1: Intersection driving scenario example. The green car has the right of way. The red car must know the green car's intention to go straight or turn right in order to turn left safely.

II. DRIVER INTENTION ESTIMATION FRAMEWORK

As explained in Section I, the driver intention estimation is the way of estimating the driver's decisions from the observable vehicle dynamics. An example of intersection scenario is shown in Fig. 1. The dashed red line shows the intention of the autonomous red car to turn left as it approaches the intersection. It must know the intention of another vehicle shown in green that has the right of way. The red car must know the path the green car will follow before turning left.

If the green car turns right, then the red car can turn left. Otherwise, the red car must stop until the green one passes the intersection. Hence, the red car should estimate the intention of the green car's driver from the vehicle's continuous observations collected using the use of on-board sensors such as Global Positioning System (GPS) and Controller Area Network (CAN) and transmitted through vehicle to vehicle communication, lidar, or radar [5].

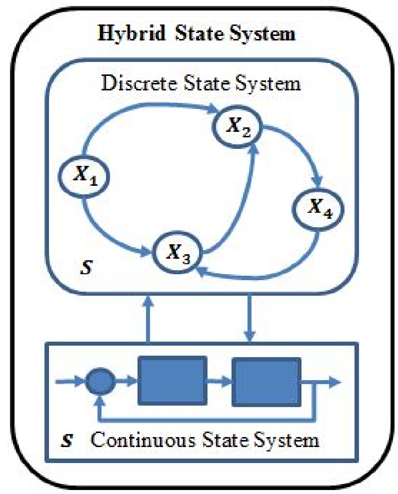


Fig. 2: Hybrid State System Setting.

A. Hybrid State System (HSS) Framework

The hierarchical relationship of a continuous-state plant and discrete-state system has been modeled using an HSS setting in diverse applications including HSS estimation [6] and autonomous vehicles [16]. As shown in Fig. 2, the HSS setting incorporates two parts including a low level continuous state system (CSS) and a higher level discrete-state system (DSS). It represents the interplay of a vehicle and its driver. It can be used to track, estimate, and predict the behavior of the vehicle and its driver. A driver reacts according to discrete events and does corresponding driving decisions on the higher level that affects the lower level continuous vehicle trajectory. The formulation and equation of the HSS setting is presented in [17].

B. Hidden Markov Models (HMM)

In this work, HMM is used as a mathematical technique that models the relationship between the DSS and CSS parts of the HSS framework [18], [19]. It represents a stochastic relationship between the driver decisions and vehicle dynamics. The changes in driver's decisions affects the continuous vehicle observations. The HMM trained with these observations determines the driver states which are not predefined. In this section, we present the background on HMM. We will discuss HMMs, and HMM inference tasks. Here, capital letters are used to denoted the variables, and lower-case letters for their values.

An HMM is a stochastic statistical model of a discrete Markov chain of a finite number of hidden variables X that can be observed by sequence of a set of output variables Y from a sensor or other sources. The transition probability from one state to another in this Markov chain is time-invariant, which makes the model stationary. The observed variables Y are stochastic with observation (emission) probabilities that are also time invariant. The overall HMM consists of n distinct hidden states and m corresponding observable symbols. In general, the observations can be discrete or continuous, however in this work, we focus on the discrete observations. The variable X has $n \geq 2$ hidden states, denoted by $x_i, 1 \leq i \leq n$, and the variable Y has $m \geq 2$ observable symbols, denoted by $y_j, 1 \leq j \leq m$. Formally, an HMM λ is defined by the parameters set (A, O, Γ) that are defined as follows:

1) The prior probability distribution (initial vector) Γ has entries $\gamma_i = p(x_i), 1 \leq i \leq n$, which are the probabilities of state x_i being the first state in the Markov chain.
2) The transition matrix A has entries $a_{i,j} = p(x_j|x_i), 1 \leq i, j \leq n$, which are the probabilities that transit from state x_i to state x_j in the Markov chain.
3) The observation matrix O has entries $o_{i,j} = p(y_j|x_i), 1 \leq i \leq n, 1 \leq j \leq m$, which are the probabilities to observe y_j given current state is x_i.

An example of an HMM where X has two states, and Y three symbols is shown in Fig. 3a. The two states are x_1 and x_2 and based on them three symbols y_1, y_2 or y_3 are observed. The parameters of the HMM (including initial vector, transition matrix and observation matrix) are also shown in Fig. 3a.

The representation of the HMM in Fig. 3a using dynamic Bayesian network is shown in Fig. 3b, unrolled for T time slices [25]. The initial vector, transition matrix and observation matrix of the HMM are represented by the labels Γ, A and O in the Bayesian network graph nodes respectively. The superscript for the variables X and Y indicate the time slice under consideration.

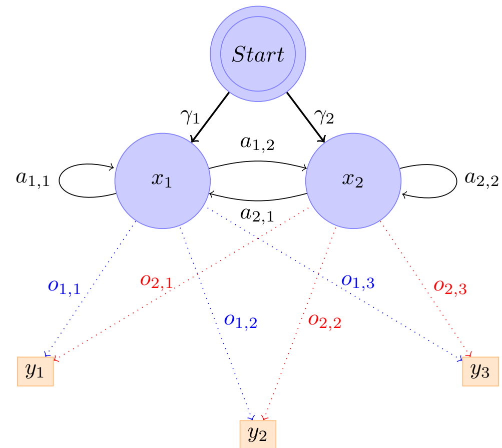
In this paper, $\mathbf{y_e}^t$ represents the actual evidence variable Y in time slice t and $\mathbf{y_e}^{t_i:t_j}$ represents a sequence of observations $\mathbf{y_e}^{t_i}, ..., \mathbf{y_e}^{t_j}$.

Inference in HMMs: Inference in temporal models means computing the conditional probability distribution of X at time t, for given evidence till time T, that is $p(X^t|\mathbf{y_e}^{1:T})$. The main inference tasks include *filtering* for $T = t$, future state *prediction* for $t > T$ and *smoothing* that is inferring the past for $t < T$. The *Forward-Backward algorithms* computes the following probabilities for all hidden states i for time $t \leq T$:

- forward probability $F(i,t) = p(x_i^t, \mathbf{y_e}^{1:t})$, and
- backward probability $B(i,t) = p(\mathbf{y_e}^{t+1:T}|x_i^t)$

resulting in

$$p(x_i^t|\mathbf{y_e}^{1:T}) = \frac{p(x_i^t, \mathbf{y_e}^{1:T})}{p(\mathbf{y_e}^{1:T})} = \frac{p(x_i^t, \mathbf{y_e}^{1:T})}{\sum_{j=1}^{n} p(x_j^t, \mathbf{y_e}^{1:T})} = \\ = \frac{F(i,t) \cdot B(i,t)}{\sum_{j=1}^{n} F(j,t) \cdot B(j,t)} \tag{1}$$

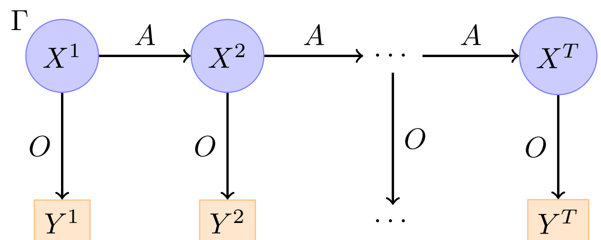


Initial Vector, Γ	
x_1	x_2
γ_1	γ_2

Transition Matrix, A		
	x_1	x_2
x_1	$a_{1,1}$	$a_{1,2}$
x_2	$a_{2,1}$	$a_{2,2}$

	Observation Matrix, O		
	y_1	y_2	y_3
x_1	$o_{1,1}$	$o_{1,2}$	$o_{1,3}$
x_2	$o_{2,1}$	$o_{2,2}$	$o_{2,3}$

(a)



$$X^1, \cdots, X^T \in \{x_1, x_2\}$$
$$Y^1, \cdots, Y^T \in \{y_1, y_2, y_3\}$$

(b)

Fig. 3: (**a**) Representation of an example hidden Markov model. (**b**) Its representation in dynamic Bayesian network for T time slices.

An iterative algorithm known as *Baum-Welch* method (Expectation Maximization) is used to train the HMM from the vehicle observations to estimate the driver state [20], [21]. It uses the *forward* and *backward algorithms* to compute the model parameters $\lambda = (A, O, \Gamma)$ that give maximum-likelihood estimates [21].

The *forward algorithm* is used to estimate the driver's intention using the trained models, $\lambda_1, \lambda_2, ..., \lambda_n$. It calculates the probabilities $P(\mathbf{Y}|\lambda_i), i = 1, 2, ..., n$ for each model for the observations sequence, $\mathbf{Y} = \{Y^1, Y^2, ..., Y^t\}$. The estimated driver's intention i at time step t is represented by the model λ_i that gives the highest likelihood probability. Thus, at time t the driver's state is calculated as

$$S(t) = \arg\max_i P(Y^1, Y^2, ..., Y^t|\lambda_i) \quad i = 1, \ldots, n \tag{2}$$

In Fig. 4, this concept is used. The final compare step in the figure represents Equation (2). As, this work focus on estimating the driver's intention near a road intersection, the possible driver action at the intersection are represented by the models λ_i that are straight, left, right and stop as shown in the figure.

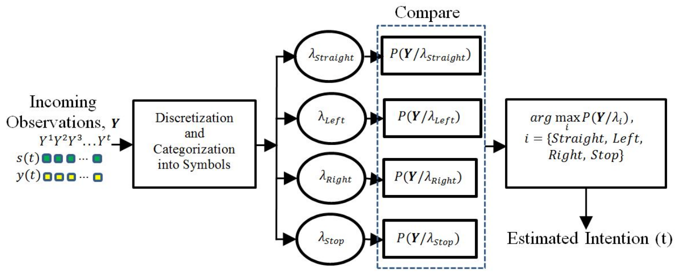


Fig. 4: Estimation of driver's intention, that best represents the new observations sequences, using four discrete HMMs for the different driver intention including Straight, Left, Right and Stop.

III. DATA COLLECTION AND ANALYSIS

The data set used to train the proposed model explained in Section II, was collected by researchers at the Ohio State University[5]. The objective of this work is estimating the driver's intention from the observable vehicle dynamics. Here, it is assumed that using the right combination of sensors, we can collect observations including yaw-rate, speed, and acceleration.

The data was collected using a 2012 Honda Accord car fitted with sensors [5]. The sensors include NovAtel GPS unit, Honda Accord controller area network (CAN) bus and Three HD cameras. The *GPS unit* collects GPS longitude, time-stamp, GPS latitude and others. The *CAN bus* collects odometer, lateral acceleration, steering wheel angle, time-stamp, speed, yaw-rate, and others. The *cameras* show right side, left side and front views of the vehicle.

Selected participant drivers drove the vehicle through paths representing common road scenarios met in daily driving. The scenarios include driving in a highway (merging and entering/exiting ramps) and road intersections (Right turn, Left turn, Straight, and Stop).

The ground truth data is extracted from the collected data where the time-stamps representing the different maneuvers are marked manually by watching the videos when the car approaches the intersection. For example, when a vehicle approaches an intersection and turn left, the time-stamp 6 seconds before and 5 seconds after the center of the intersection is marked as "Left Turn" and the corresponding speed, acceleration and yaw-rate of the vehicle is extracted. Fig. 5 shows different example maneuvers for the ll seconds time series observations of speed, acceleration and yaw-rate. These observations can describe the driver's intention at the intersection as shown in [5], [11], [12], [24]. Despite acceleration being an important parameter to capture the change in velocity, however, in this work, we only consider the speed and yaw-rate to develop the models. As shown in Fig. 5, the acceleration for the different maneuvers is almost the same.

The speed and yaw-rate time-series observations are categorized into different groups. The speed is grouped into high speed (Class 1) for values greater than or equal to 10 m/s and low speed (Class 0) for less than 10 m/s. The low speed maneuvers like left turn, right turn and stop are below 10 m/s. The yaw-rate is grouped into three classes including Class 0 for yaw-rate less than -3 rad/s, Class 1 for yaw-rate greater than or equal to -3 rad/s and less than or equal to 3 rad/s, and Class 2 for yaw-rate greater than 3 rad/s. The -3 rad/s and 3 rad/s are chosen to show the separation between a left turn (negative turn), right turn (positive turn) and random small turns. The classes are categorized into discrete symbols to train the HMM models. Speed in Class 0 and yaw-rate in Class 0 categorized into symbol 1. Likewise, the combination of the classes results in 6 symbols. Consequently, each maneuver 11 seconds time-series speed and yaw-rate data is converted to a sequence of symbols.

IV. RESULTS

The continuous time series observations of the vehicle's speed and yaw-rate at time $t = \tau$ are discretized and used to train the HMM as sequence of discrete symbols. The four different HMM models for each driver's intention are combined to classify the vehicle trajectory at each time step. To evaluate the classification performance of the model, a Confusion Matrix, where the total number of the predicted classes are in the column and the actual classes are in the row direction is used.

The discrete HMMs have N=4 possible hidden states and M=6 observable symbols. The model is trained with 70% of the dataset and tested with 30%. Here, the dataset contains 9 straight, 6 left turn, 5 right turn, and 7 stop maneuvers. The continuous speed and yaw-rate data that is discretized and combined into sequence of symbols, is ll seconds with 0.1 seconds time step. This results in around 990 straight, 660 turn left, 550 right turn, and 770 stop data points for each time $t = \tau$. The total number of symbols in the data set becomes around 2970 with their corresponding labels that is divided into 70% training and 30% testing set. In other words, the HMMs are trained with 6 straight, 4 left turn, 4 right turn, and 5 stop, and tested with 3 straight, 2 left turn, 1 right turn, and 2 stop maneuvers. The performance of the proposed model with the 30% testing set is shown on Table I in a confusion matrix. The accuracy of the model is 89.45 % that is calculated as the accurate data point predictions divided by the total test data points, ((327+192+69+192)/872)*100%.

The estimated intention of the driver at every time $t = \tau$ in the 11 seconds time-span from the discrete HMM is shown in Fig. 6. In the figure, the discretized sequence of observations from different vehicle maneuvers including straight, right turn, left turn and stop are estimated. For example, Fig. 6b shows the estimation of the driver's intention to "Turn Left" from a ground truth data of speed and yaw-rate corresponding to a left turn maneuver that is discretized and combined to form sequence of symbols. At each time step $t = \tau$ equation (2) is computed for the incoming sequence of symbols to determine

TABLE I: CONFUSION MATRIX FOR DISCRETE HMM USING 30% TESTING SET AT t=τ

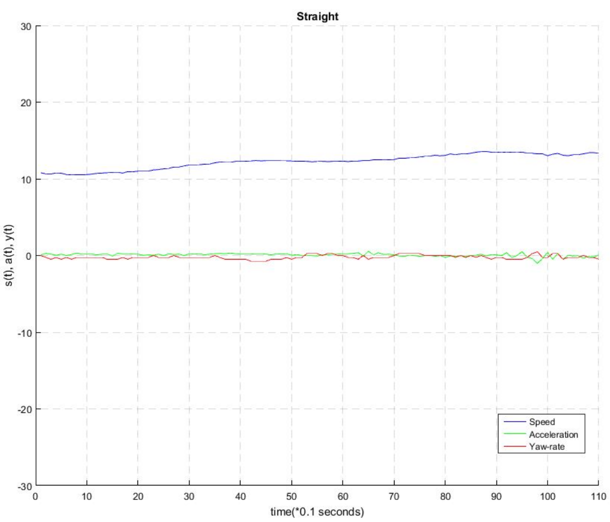
Actual Maneuvers at t=τ	**Predicted Maneuvers at t=τ**			
	Straight	*Left Turn*	*Right Turn*	*Stop*
Straight	327	0	0	0
Left Turn	0	192	0	26
Right Turn	0	18	69	22
Stop	8	18	0	192

Accuracy = 89.45

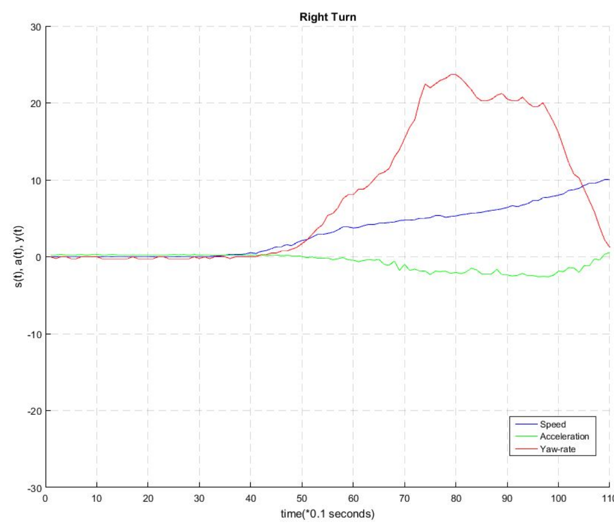
from the four models which one best estimates the driver's intention as shown in Fig. 4. This is repeated for the other example maneuvers for a vehicle approaching an intersection and going straight, turning right or stopping. Generally, the figure shows the proposed approach has solved the problem of estimating driver intention near a road intersection very well.

V. CONCLUSION AND FUTURE WORK

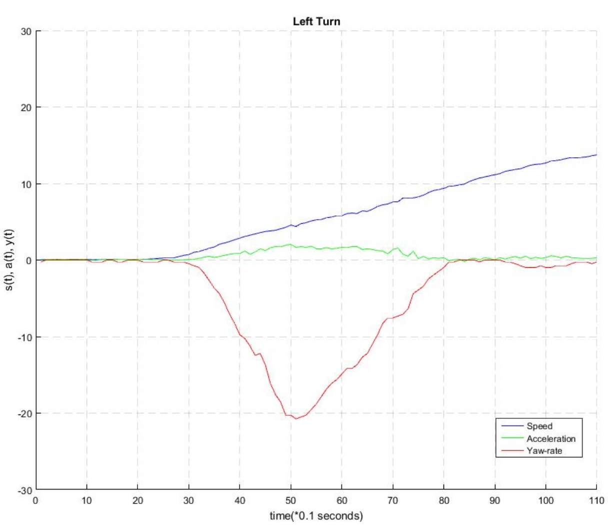
In this work, the problem of driver's intention estimation near a road intersection using discrete HMM is studied. The model is trained with discretized sequence of speed and yaw-rate of the vehicle approaching the intersection. The acceleration of the vehicle is not significant in differentiating the different intention of the driver at the intersection. The HMM uses Markov chains intuition to represent the stochastic process that results in the observations. The HSS framework that relates the driver's discrete decisions and the continuous vehicle dynamics is used as a base to develop this model.



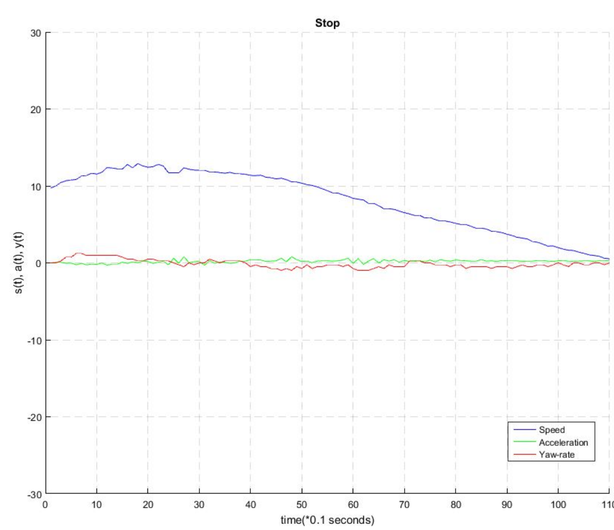
(a)



(b)



(c)



(d)

Fig. 5: Time series observations of Speed, Acceleration and Yaw-rate for 11 seconds (a) Straight (b) Right Turn (c) Left Turn (d) Stop.

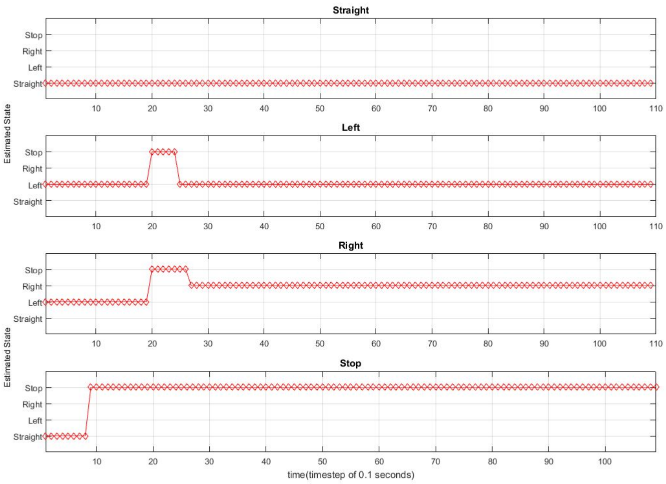


Fig. 6: Estimation of driver's intention at each timestep using four discrete HMMs for the different individual vehicle maneuvers including (a) Straight (b) Left (c) Right and (d) Stop.

Thus, the HMM is used as the mathematical technique to relate the two parts of the HSS framework. The result is very promising to be applied in ADAS. In future work, the performance of the proposed approach in a bigger dataset and other machine learning techniques for better recognition will be investigated. Moreover, utilizing large scale dataset from SHRP2, the developed method will be used in other driving situations such as lane change, high-way driving, and other near crash events.

ACKNOWLEDGMENT

This research is supported by the US Department of Transportation (USDOT) under University Transportation Center (UTC) Program (DTRT13-G-UTC47).

REFERENCES

[1] W. D. Jones, "Keeping cars from crashing," IEEE Spectrum, vol. 38, no. 9, pp. 40-45, 2001.

[2] A. Kurt, and U. Ozguner, "A probabilistic model of a set of driving decisions," In 14th International IEEE Conference Intelligent Transportation Systems (ITSC), pp. 570-575, 2011.

[3] B. Song, and D. Delorme, "Human driver model for smart AHS based on cognitive and control approaches," In ITS America conference, 2000.

[4] U. Ozguner, T. Acarman, and K. Redmill, Autonomous ground vehicles, Artech House Publishers, 2011.

[5] V. Gadepally, A. Krishnamurthy, and U. Ozguner, "A Framework for Estimating Driver Decisions near Intersections," Trans. Intell. Transp. Syst., vol. 15, no. 2, pp.637-646, 2014.

[6] A. Kurt, J. Yester, Y. Mochizuki, and U. Ozguner, "Hybrid-state driver/vehicle modelling, estimation and prediction," In Proceedings 13th IEEE ITSC, pp. 806-811, 2010.

[7] I. Patel and Y. S. Rao, "Speech Recognition Using Hidden Markov Model with MFCC-Subband Technique", 2010 International Conference on Recent Trends in Information, Telecommunication and Computing, Kochi, Kerala, 2010, pp. 168-172.

[8] E. Birney, "Hidden Markov models in biological sequence analysis", in IBM Journal of Research and Development, vol. 45, no. 3.4, pp. 449-454, May 2001.

[9] R. S. Mamon and R. J. Elliott (Eds.), "Hidden Markov Models in Finance", International Series in Operations Research & Management Science, Springer US, 2014.

[10] H. Bunke and T. Caelli, "Hidden Markov Models: Applications in Computer Vision", World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2001.

[11] S. Amsalu, A. Homaifar, F. Afghah, S. Ramyar, and A. Kurt, "Driver Behavior Modeling near Intersections Using Support Vector Machines based on Statistical Feature Extraction," in IEEE Intelligent Vehicles Symposium, June 28 - July 1, 2015.

[12] S. B. Amsalu and A. Homaifar, "Driver behavior modeling near intersections using Hidden Markov Model based on genetic algorithm", 2016 IEEE International Conference on Intelligent Transportation Engineering (ICITE), Singapore, 2016, pp. 193-200.

[13] N. Oliver and A. Pentland, "Graphical models for driver behavior recognition in a smart car," in Proc. IEEE IV Symp., 2000, pp. 7-12.

[14] N. Kuge, T. Yamamura, O. Shimoyama, and A. Liu, "A driver behavior recognition method based on a driver model framework," SAE Trans., vol. 109, no. 6, pp. 469-476, 2000.

[15] D. Mitrovic, "Reliable method for driving events recognition," IEEE Trans. Intell. Transp. Syst., vol. 6, no. 2, pp. 198-205, Jun. 2005.

[16] A. Kurt and U. Ozguner, "Hybrid state system development for autonomous vehicle control in urban scenarios," in Proceedings of the IFAC 2008 World Congress, pp. 9540-9545, 2008.

[17] M. Dogruel and U. Ozguner, "Discrete and hybrid state system modeling and analysis," Turkish Journal of Electrical Engineering and Computer, vol. 5, no. 2, pp. 263-286, 1997.

[18] L. Rabiner and B. Juang, "An introduction to hidden Markov models," IEEE ASSP Mag., vol. 3, no. 1, pp. 4-16, Jan. 1986.

[19] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," Proc. IEEE, vol. 77, no. 2, pp. 257-286, Feb. 1989.

[20] L. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," Ann. Math. Stat., vol. 37, no. 6, pp. 1554-1563, Dec. 1966.

[21] J. Bilmes,"A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," Int. Comput. Sci. Inst., BerkeleSewumehony, CA, Tech. Rep. ICSI-TR-97-021, 1997.

[22] S. Picek, D. Jakobovic and M. Golub, "On the Recombination Operator in the Real-Coded Genetic Algorithms," Proceedings of the IEEE Congress on Evolutionary Computation, pp.3103 -3110, Jun. 2013.

[23] M. Lozano, F. Herrera, and J. R. Cano, "Replacement strategies to preserve useful diversity in steady-state genetic algorithms," Information Sciences, vol. 178, Issue 23, pp 4421-4433, Dec. 2008.

[24] G. Aoude, V. Desaraju, L. Stephens and J. How, "Behavior Classification Algorithms at Intersections and Validation using Naturalistic Data," IEEE Intell. Veh. Sympo, 2011.

[25] K. Murphy, Hidden Markov Model (HMM) toolbox for Matlab. [Online]. Available: http://www.ai.mit.edu/ murphyk/Software/ HMM/hmm.html.