



Learning V2V interactive driving patterns at signalized intersections[☆]

Weiyang Zhang^a, Wenshuo Wang^{b,*}

^a Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109, United States

^b Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, United States



ARTICLE INFO

Keywords:

Interaction patterns
Vehicle-to-vehicle (V2V)
Bayesian nonparametric learning
Driving primitives

ABSTRACT

Semantic understanding of multi-vehicle interaction patterns at intersections play a pivotal role in proper decision-making of autonomous vehicles. This paper presents a flexible framework to automatically extract these interaction patterns from observed temporal sequences based on driving primitives. A Bayesian nonparametric approach is developed to segment vehicle-to-vehicle (V2V) sequential observations into small analytically interpretable components (called driving primitives) without using prior knowledge. Then, an unsupervised clustering method is developed to gather these driving primitives into groups (called driving patterns) by integrating with dynamic time warping. In addition, the extracted driving primitives are used to quantitatively analyze the similarities between behaviors at different intersections using a relative entropy metric, i.e., KullbackLeibler (KL) divergence. Finally, 706 naturalistic V2V events from eight typical urban signalized intersections are used to validate the effectiveness of the proposed primitive-based framework. Experimental results demonstrate that there exist 15 types of interactive driving patterns for V2V behaviors at intersections in our database. Moreover, the distribution of interactive driving patterns could characterize the types of intersections.

1. Introduction

Intersection driving scenario in this paper is referred to as the scenario where multiple vehicles are spatially close to and interact with each other over a while at intersections (Wang et al., 2018a). It is one of the most complex and challenging driving scenarios for self-driving cars (Grembek et al., 2019). The latest statistical data of the National Highway Traffic Safety Administration (NHTSA) indicates that around 40% of all crashes and 50% of severe collisions in the U.S. occur at intersections (Administration, 2018). Human driving behavior is a dynamic and stochastic process in nature (Nechyba and Xu, 1998). The road users do not solely determine their behaviors based on their current states but also predict the behaviors of others at intersections (Grembek et al., 2019). The limited and obscured sight distances and wrong intuition on gap acceptance are the main factors in these accidents (Creaser et al., 2007a). For autonomous vehicles, the typical process of negotiating at intersections usually consists of a closed loop of perception, decision making, and control. Due to uncertainties on the continuous state of nearby vehicles and their potential discrete states such as braking and turning, decision making is becoming the most crucial and challenging component (Schwarting et al., 2018). Thus, in order to make autonomous vehicles able to interact with nearby human drivers smoothly and safely, the intersection driving scenarios should be thoroughly investigated.

[☆] This article belongs to the Virtual Special Issue on “Trajectory Optimizaiton.”

* Corresponding author.

E-mail addresses: zhangwy@umich.edu (W. Zhang), wwsbit@gmail.com, wenshuow@andrew.cmu.edu (W. Wang).

Many solutions have been developed to analyze and model the interactive driving behavior at intersections, thus providing essential decision-making rules for autonomous vehicles; for example, subjectively decomposing the whole scenarios into finite segments. Researchers segment the intersection driving scenarios into several discrete states using the Gaussian mixture models (GMM) (Gadeppally et al., 2014; Havlak and Campbell, 2014; Wang et al., 2018c; Butakov and Ioannou, 2015) and then described them via stochastic processes such as the Markov process (Tang et al., 2016; Aoude et al., 2012, 2011). Dynamic Bayesian networks (DBN) was also implemented, wherein the discrete states were empirically determined by subjective observation (Kasper et al., 2012; Gindele et al., 2015; Fu et al., 2018; Liang and Lee, 2014). However, the approaches mentioned above require the prior knowledge of, for example, road conditions (e.g., intersection type) and signal information (e.g., signalized and unsignalized traffic), which makes it restricted to some specific conditions. For instance, Sarkar et al. (2017) mainly focused on the left turn scenarios of traffic agents at the T-shape intersection. Aoude et al. (2012) implemented a hidden Markov model (HMM) to classify the driving behaviors occurred only at signalized intersections, while Creaser et al. (2007b) mainly evaluated the drivers gap acceptance decisions at the rural intersections with two-way stop signs. The approaches aforementioned are suitable to use under specific situations; however, it is practically intractable for complicated scenarios because of insufficient prior knowledge on the interactive driving patterns among the traffic agents.

Recently, some deep-learning methods have been implemented to facilitate the decision-making process of autonomous vehicles. For example, Bhatt et al. (2017) introduced a monocular-camera-based convolutional network (CNN) method to detect if vehicles reach the intersections. Isele et al. (2018) proposed a navigation method at unsignalized intersections based on deep Q-Networks (DQNs). Although these methods have been successfully implemented in specific cases, they are still limited to be used to understand the context behind of the behaviors and may need huge computational resource. Thus, segmenting multi-vehicles interaction behavior into recognizable patterns can help us understand the scenarios, and thereby facilitate computational-cost algorithms to practice, for example, partially observable Markov decision processes (POMDP). The POMDP method is a common and flexible way to model sequential decision problems under uncertainties of the intersection driving patterns by providing a mathematically rigorous framework (Brechtel et al., 2014). However, solving POMDP is computationally intractable, especially for the intersection driving scenarios containing unknown states. Fortunately, discretizing driving scenario into countable patterns or states can increase the computation speed. For instance, Galceran et al. (2017) simplified the full POMDP solution by reducing the decision to a limited set of Bayesian change points. Wei et al. (2011) developed a point-based MDP by only taking uncertainty into account at the first planning step in single-lane merging scenarios. Essa and Sayed (2018) analyzed the rear-end conflicts behaviors by splitting the trajectories into 4 different traffic flow states. Lutzeler and Dickmanns (2000) implemented a vehicle decision-making system by dynamically specifying fixation points via cameras. Thus, segmenting complex driving scenarios into discrete patterns can facilitate the decision-making learning process and reduce the computational cost, especially for the issues in high dimensional space.

According to the discussion above, it is necessary to develop an approach that can semantically decompose the complex interactive driving behavior at intersections into discrete states with less prior knowledge. Also, uncertainties of the potential discrete states should be considered. However, the complexity and uncertainty of driving environment makes it hard to find a mathematically rigorous unified approach to analyze the behaviors at all kinds of intersections. Besides, the big traffic data in a high-dimensional and large-scale space will overwhelm the human mind and heuristic analysis (Wang et al., 2017a).

This paper will introduce a primitive-based framework, which can automatically decompose the driving behavior at intersections into several interpretable interaction patterns with less prior knowledge by integrating Bayesian nonparametric learning with clustering algorithms. The main contributions of this paper are threefold.

- Presenting a primitive-based framework to learn interactive driving patterns at intersections based on Bayesian nonparametric statistics.
- Developing an efficient clustering method to gather driving patterns at intersections over spatiotemporal features of driving primitives.
- Verifying the effectiveness of our proposed framework for analyzing interactive driving patterns through naturalistic driving data.

The remainder of this paper is organized as follows. Section 2 introduces the developed primitive-based framework. Section 3 introduces the clustering algorithm and evaluation approaches. Section 4 displays the experiments and data collection. Section 5 discusses and analyzes the experimental results. Section 6 concludes this work and discusses future work.

2. Methods to extract driving patterns

This section will describe the interactive behaviors at intersections mathematically and then introduce the methodologies to extract the driving primitives. Our proposed framework of learning interactive driving patterns mainly consists of four steps (Fig. 1): primitive extraction, feature representation, clustering, and driving pattern evaluation and analysis.

2.1. Intersection scenario and its primitives

In order to facilitate the analysis, this paper mainly focuses on the V2V interactive driving behavior at intersections with two vehicles engaged. The sequential observations include position (latitude and longitude) and speed of two vehicles, formulated as

$$\mathbf{S} = \{s_1, \dots, s_t, \dots, s_T\} \quad (1)$$

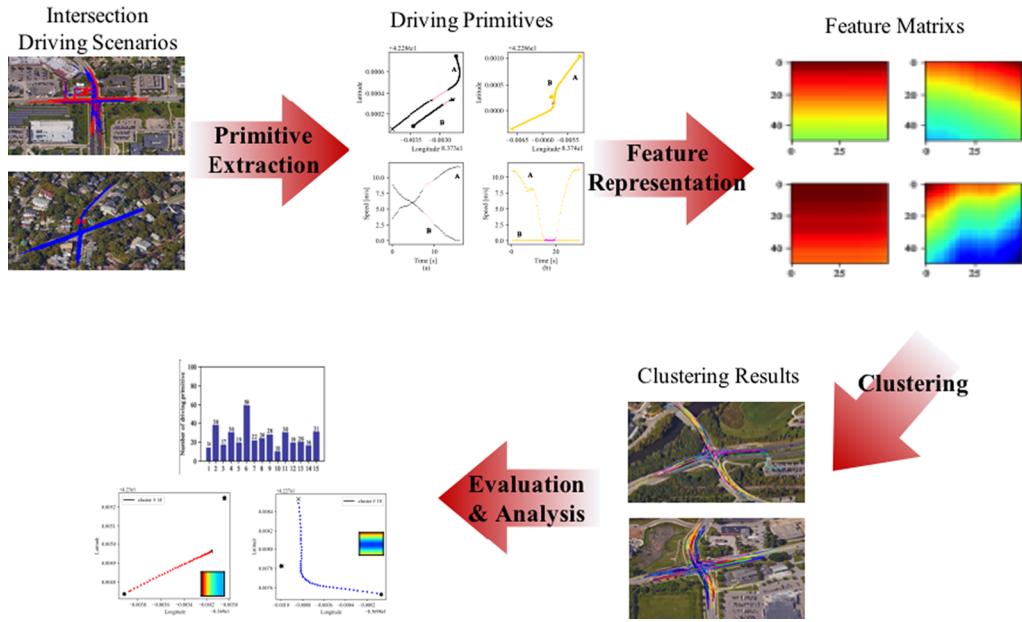


Fig. 1. Diagram of our proposed driving primitive-based framework.

where $s_t = [p_t^{(1)}, p_t^{(2)}, v_t^{(1)}, v_t^{(2)}] \in \mathbb{R}^6$ with $p_t^{(1)} \in \mathbb{R}^2$ and $p_t^{(2)} \in \mathbb{R}^2$ the positions of the two vehicles at time t , $v_t^{(1)}$ and $v_t^{(2)}$ represent the speed of the two vehicles at time t , and T is the time length of sequential observations S . The segments (denoted as \bar{s}_k) of the sequential observations S can be described by

$$\bar{s}_k = \{s_m, \dots, s_n\} \quad (2)$$

where $\bar{s}_k \subseteq S$ with $1 \leq m \leq n \leq T$, and k represents the k -th primitives. Thus, the time length of \bar{s}_k is $(m - n + 1) \in [1, T]$. The extracted primitives will not overlap each other in a single data sequence. The primitive with very short duration provides little information for driving pattern analysis and will be removed.

2.2. Bayesian nonparametric learning

Taking a straight analysis of the multiplex driving scenarios with high dimensional data is manually intractable because of our limited prior knowledge. Some approaches have been developed to analyze and capture the spatial characteristics of data sequences, for example, support vector machines (Ben-Hur et al., 2001; Wang et al., 2017b) and distance-based measurements (Euclidean distance and Manhattan distance) (Majone, 1970). However, their analysis is computationally intractable and also did not consider the temporal dynamic of time series. This paper will introduce a Bayesian nonparametric learning approach by integrating a hierarchical Dirichlet process (HDP) with a hidden Markov model (HMM) (Fox et al., 2011) (denoted as the sticky HDP-HMM). This approach can automatically decompose the spatiotemporal sequences into primitives with consideration of the uncertainties of observations but with less prior knowledge. In what follows, some preliminaries of the sticky HDP-HMM will be discussed.

2.2.1. HMM

Dynamic driving behavior can be interpreted as a Markov process in which the future predictions are independent of all but the most recent observations. A flexible and efficient model is the discrete Markov model with a layer of discrete hidden states, called HMM. This paper utilizes HMM as the basis to describe the dynamic process of two-vehicles interactions. Thus, each data sample in time series $S = \{s_t\}_{t=1}^T$ can be considered as an observation. Given a set of hidden state \mathcal{X} , each hidden state at time t will belong to an entry of \mathcal{X} , i.e., $x_t = x_i \in \mathcal{X}$, where x_i is the i -entry in \mathcal{X} and can be considered as the i -type primitive. Then, the adjacent data set $\{s_m, \dots, s_n\}$ with the identical entry in \mathcal{X} form a single primitive \bar{s}_k . The transition probability from states x_i to x_j is denoted as $\pi_{i,j}$ with $\pi_i = \{\pi_{i,1}, \pi_{i,2}, \dots\}$. The observation s_t at time t given hidden state x_t is generated by $s_t = F(s_t | x_t, \theta_{x_t})$ governed by parameters θ_{x_t} . Thus, our case can be formulated using the HMM model

$$x_t | x_{t-1} \sim \pi_{x_{t-1}} \quad s_t | x_t \sim F(\theta_{x_t}) \quad (3)$$

2.2.2. HDP

The uncertainties on the time length T of the data sequence can make the transition probability $\pi_{i,j}$ various from case to case. Usually, the intersection driving scenario with a long duration can contain many relevant primitives. Thus, the Dirichlet process (DP) (Teh et al., 2005), denoted by $DP(\gamma, H)$, is introduced to ensure $\sum \pi_{i,j} = 1$ for any cases when $i \rightarrow \infty$, H is the base distribution

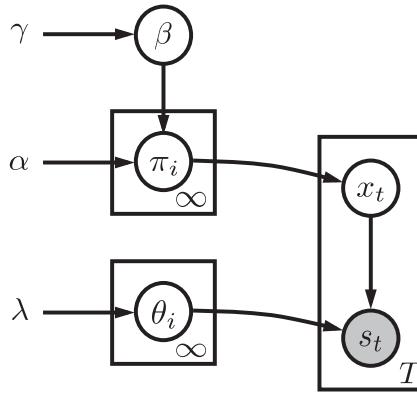


Fig. 2. Graphical illustration of the sticky HDP-HMM.

measure. Two separate suitable random sequences can realize the Dirichlet process: one is the weights of given probability measure (β_i), and the other one is the Dirac measure (δ_{θ_i}). It can be mathematically interpreted as

$$G_0 = \sum_{i=1}^{\infty} \beta_i \delta_{\theta_i}, \quad \theta \sim H \quad (4a)$$

$$\beta_i = v_i \prod_{\ell=1}^{i-1} \left(1 - v_\ell\right), \quad v_i \sim \text{Beta}\left(1, \gamma\right) \quad (4b)$$

where γ is the hyperparameter and β_i are the weights sampled by a stick-breaking construction, denoted as $\beta \sim \text{GEM}(\gamma)$, which ensures the sum of weights β_i is always equal to one for arbitrary number of data samples when $i \rightarrow \infty$. In order to put less prior knowledge on the hyperparameter and make the information between primitives shareable, the prior of the HMM transition probability measures G_j is defined by a hierarchical DP (HDP)

$$G_j = \sum_{i=1}^I \pi_{j,i} \delta_{\theta_i} \quad (5)$$

where δ_θ is a mass concentrated at θ . In our case, we take $G_j \sim \text{DP}(\alpha, G_0)$, where G_0 is drawn by DP.

2.2.3. Sticky HDP-HMM

Based on the above discussion, the sticky HDP-HMM (Fox et al., 2011; Wang et al., 2018b; Wang and Zhao, 2018) is introduced. Compared with HDP-HMM, the additional sticky parameter $\kappa \in [0, 1]$ can govern the expected self-transition probability by adjusting the concentrate values to $\alpha + \kappa$ and base probability measure to $\frac{\alpha\beta + \kappa\delta_i}{\alpha + k}$. Fig. 2 illustrates the entire process. Therefore, the generative model can be summarized as

$$\begin{aligned}
 \beta | \gamma &\sim \text{GEM}(\gamma) \\
 \pi_i | \alpha, \beta, \kappa &\sim \text{DP}\left(\alpha + \kappa, \frac{\alpha\beta + \kappa\delta_i}{\alpha + k}\right) \\
 x_t | x_{t-1} &\sim \pi_{x_{t-1}} \\
 s_t | x_t &\sim F(\theta_{s_t}) \\
 \theta_i | H &\sim H
 \end{aligned} \quad (6)$$

where $t \in [0, T]$. All the hidden parameters can be estimated via the expectation-maximization algorithm. In order to make the posterior probability estimation of hidden states analyzable and computationally tractable given the observation, all the hyperparameters are set as a Gamma distribution in the light of the conjugate prior. For the parameters θ , in our case, the base measure is subject to a Gaussian distribution, which makes the posterior estimation tractable with Gibbs sampling algorithms (Wang et al., 2018b).

3. Clustering of driving patterns

Based on the extracted driving primitives, the patterns of these primitives are then clustered. In what follows, an unsupervised clustering approach is developed to gather the homogeneous primitives into groups.

3.1. Feature representation

It is practically intractable to group sequential data in the spatiotemporal space using traditional clustering algorithms such as k -means because directly computing the center of a trajectory is meaningless. An alternative is to select the representative features of these primitives to do clustering. These selected features should be informative enough to capture the spatiotemporal interaction characteristics of vehicles in the intersection primitives. Some approaches have already been developed such as the Euclidean distance with mean values, Pearson's correlation coefficient with cross-correlation-based distances, short time series distance (Möller-Levet et al., 2003), dynamic time warping (Müller, 2007), and probability-based distance function (Liao, 2005). In our case, the temporal and spatial features of two vehicles are both essential to represent their dynamic interaction. Besides, the selected features should reserve as sufficient information as possible because less prior knowledge of the interactive driving patterns is available. Therefore, a dynamic time warping (DTW) approach is implemented to measure the interactive relation of two vehicles through a feature matrix, thereby revealing their spatiotemporal interactions. Here, the position trajectory makes an equal contribution as speed to represent the dynamic interaction behavior, and their corresponding feature matrices can be separately calculated via DTW.

Given a driving primitive \bar{S}_k with a length of $(m - n + 1)$, the elements in \bar{S}_k are denoted as

$$p^{(1)} = [p_m^{(1)}, p_{m+1}^{(1)}, \dots, p_n^{(1)}] \quad (7a)$$

$$p^{(2)} = [p_m^{(2)}, p_{m+1}^{(2)}, \dots, p_n^{(2)}] \quad (7b)$$

$$v^{(1)} = [v_m^{(1)}, v_{m+1}^{(1)}, \dots, v_n^{(1)}] \quad (7c)$$

$$v^{(2)} = [v_m^{(2)}, v_{m+1}^{(2)}, \dots, v_n^{(2)}] \quad (7d)$$

The feature matrices of position trajectory and speed are denoted as $M^{(p)}$ and $M^{(v)}$ in $\mathbb{R}^{(n-m+1) \times (n-m+1)}$, respectively. Considering $p_{i,j}$ and $v_{i,j}$ as the elements of feature matrices $M^{(p)}$ and $M^{(v)}$ with $p_{i,j}, v_{i,j} \in \mathbb{R}_{\geq 0}$ and $m \leq i, j \leq n$, respectively, then $p_{i,j}$ represents the local similarity between $p_i^{(1)}$ and $p_j^{(2)}$, and $v_{i,j}$ represents the local similarity between $v_i^{(1)}$ and $v_j^{(2)}$. The local similarities $p_{i,j}$ and $v_{i,j}$ are quantitatively computed by

$$p_{i,j} =: D_p(p_i^{(1)}, p_j^{(2)}) \quad (8a)$$

$$v_{i,j} =: D_v(v_i^{(1)}, v_j^{(2)}) \quad (8b)$$

where $D(\cdot, \cdot)$ is a distance measurement function. A small value of $p_{i,j}$ indicates the positions of the two vehicles are spatially close to each other, and a small $v_{i,j}$ value represents a small speed difference of two vehicles. Given a two-dimensional data point $p \in \mathbb{R}^2$, the Euclidean distance is used to calculate D_p

$$D_p(p_i^{(1)}, p_j^{(2)}) = \|p_i^{(1)}, p_j^{(2)}\|_2 \quad (9a)$$

While $v \in \mathbb{R}$ is a one-dimensional data sequence and the Euclidean distance measurement is unsuitable for implementing. In order to represent the discrepancy of two vehicles' speed, the Manhattan distance is applied to calculate D_v

$$D_v(v_i^{(1)}, v_j^{(2)}) = \|v_i^{(1)}, v_j^{(2)}\|_1 \quad (9b)$$

Thus, the DTW can characterize the spatiotemporal relationship between two vehicles with quantitative feature matrices.

3.2. Data scaling and normalization

The length of primitives is considerably different from each other. Directly implementing Eqs. (9a) and (9b) to primitives will generate the feature matrices in different sizes, thus making clustering algorithms practically intractable. Therefore, scaling-up and scaling-down the extracted primitives into the same length is requisite before employing Eqs. (9a) and (9b). A linear interpolation was used to scale all the extracted primitives into a preset length l (Meijering, 2002). Given any two primitive sequential data $s_0 = [p_0, v_0]$ at time t_0 and $s_1 = [p_1, v_1]$ at time t_1 , the unknown primitive vector s_t at time $t \in [t_0, t_1]$ can be approximated by

$$s_t = \left[p_0 + \left(t - t_0 \right) \frac{p_1 - p_0}{t_1 - t_0}, v_0 + \left(t - t_0 \right) \frac{v_1 - v_0}{t_1 - t_0} \right] \quad (10)$$

The value of l can be set as the mean of the length of all the extracted primitives.

The magnitude of the value of feature matrices regarding position trajectory is significantly different from speed, which could cause futile results. Normalization is one of the solutions to this problem. Many off-the-shelf approaches have already been developed to do normalization such as standard score, the coefficient of variation, feature scaling, and division by the maximum. Considering both $p_{i,j}$ and $v_{i,j}$ in the feature matrices are non-negative, normalization is carried out via the division by its maximum value:

$$M_{\text{norm}}^{(p)} = \frac{M^{(p)}}{M_{\text{max}}^{(p)}} \quad (11\text{a})$$

$$M_{\text{norm}}^{(v)} = \frac{M^{(v)}}{M_{\text{max}}^{(v)}} \quad (11\text{b})$$

where $M_{\text{max}}^{(p)}$ and $M_{\text{max}}^{(v)}$ are the maximum values of feature matrices in terms of position trajectories and speeds, respectively.

3.3. Clustering

In order to obtain the interactive driving patterns at intersections, the unsupervised learning – clustering algorithms – is implemented. Many clustering methods have already been developed and used such as k -means (Jain, 2010), BIRCH (Zhang et al., 1996), DBSCAN (Guha et al., 1998; Xu and Tian, 2015). As discussed above, the trajectory matrices are independent on the speed matrices, thus we reshape the two feature matrices into one feature vector to make it compatible with the clustering algorithms.

Given one primitive \bar{S} with the normalized feature matrix $M_{\text{norm}}^{(p)}$ and $M_{\text{norm}}^{(v)}$, we reshape them into a vector ϕ by

$$\begin{aligned} f: (M_{\text{norm}}^{(p)}, M_{\text{norm}}^{(v)}) &\rightarrow \phi \\ \mathbb{R}^{l \times l} \times \mathbb{R}^{l \times l} &\rightarrow \mathbb{R}^{1 \times 2l^2} \end{aligned} \quad (12\text{a})$$

$$\phi = [p_{1,1}, \dots, p_{l,l}, v_{1,1}, \dots, v_{l,l}] \quad (12\text{b})$$

Then, the k -means clustering is used to group the feature vectors $\phi = \{\phi_1, \phi_2, \dots, \phi_N\}$, where N is total amount of the primitives and ϕ can be partitioned as k ($\leq N$) clusters set $C = \{C_k\}$. Assume μ_i as the center of cluster C_i , the objective of k -means is to minimize the distance within clusters by optimizing

$$\min \sum_{i=1}^k \sum_{\phi_i \in C_i} \left\| \phi_i - \mu_i \right\|^2 \quad (13)$$

Thus, minimizing the sum of within-cluster squared errors allows us to get an optimal solution. Finally, the primitives with similar temporal-spatial features are gathered into the same groups.

3.4. Clustering result evaluation

Evaluating the clustering performance is challenging with unknown ground truth. In our case, it is intractable to compute the accuracy over the ground truth since the correct partition is not available in the multi-vehicle interaction scenarios (Arbelaitz et al., 2013). So it can be challenging to determine the number of clusters and quantify clustering result in terms of accuracy. Here, the within-cluster distance λ_w and between-cluster distance λ_b are introduced to evaluate clustering performance (Arbelaitz et al., 2013), computed by

$$\lambda_w = \frac{\sum_{i=1}^{n_c} \sum_{\phi_i \in C_i} \left\| \phi_i - \mu_i \right\|^2}{N - n_c} \quad (14\text{a})$$

$$\lambda_b = \frac{\sum_{i=1}^{n_c} n_i \|\mu_i - \bar{\mu}\|^2}{n_c - 1} \quad (14\text{b})$$

where n_i is the number of driving primitives in the i -th cluster, n_c is the amount of the clusters, and $\bar{\mu}$ is the mean of ϕ . Considering the degrees of similarity will increase for both with-in distance and between distance when increasing the number of clusters n_c . Thus, the values of λ_w and λ_b will both decrease correspondingly. The cluster number is determined by weighing the computational cost and the decreasing rate of λ_w and λ_b .

4. Data preprocessing and parameters setting

Model parameters should be appropriately set to facilitate sequential analysis and evaluation. This section will introduce data collection and model settings, including three parts: data collection and extraction, data pre-processing, hyper-parameter settings.

4.1. Data collection and filtering

The raw experimental data was collected from the database organized by the University of Michigan Safety Pilot Model Development (SPMD) program (Bezzina and Sayer, 2015; Wang and Zhao, 2018). The data we used in this paper is from the US Department of Transportation (USDOT) (<https://catalog.data.gov/dataset/safety-pilot-model-deployment-data>). For this program, the dedicated short-range communications (DSRC) technology was used for the communication between two vehicles. The DSRC would record the driving data (i.e., position and speed) of each equipped vehicle at intersections when the distance of two vehicles was less than 100 m. The onboard GPS recorded the latitude and longitude information of each vehicle, and the by-wire speed sensor

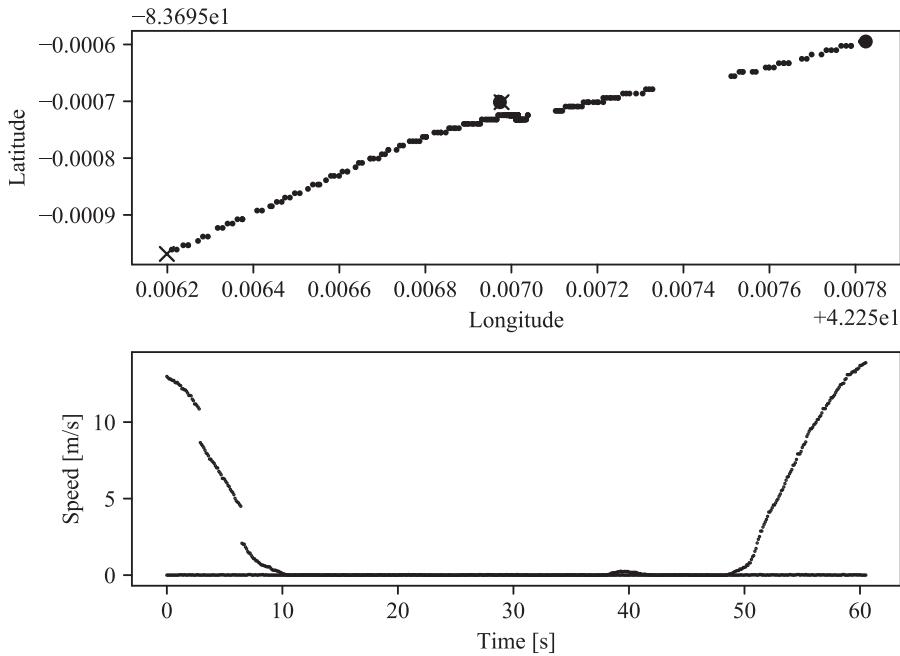


Fig. 3. An example of abnormal encounter scenarios at intersections with two obvious discontinuous intervals, i.e. partial GPS data are missing.

recorded speed. Data refresh frequency was 10 Hz.

Before training models on these data samples, some abnormal data samples were removed; for example, some encountering driving records with deficient data were removed because of missing GPS signals. It is common when the vehicle is driving in some specific place, for example, going through bridges and tunnels. Fig. 3 shows a driving encounter with deficient data: one car always keeps stationary, and the other car slows down, keeps stationary, and then accelerates. An approach was implemented to remove these abnormal data by evaluating the derivative of each point with

$$s'_t = \frac{s_{t+1} - s_t}{\Delta t} \quad (15)$$

When $|s'_{t+1} - s'_t|$ is greater than our defined threshold, the data will be considered as abnormal and then removed.

4.2. Data preprocessing

Selecting the intersections that contain many driving encounter records can enrich data samples. First, the distribution of collected data with 8936 driving events was shown on Google map (Fig. 4), and then the intersections with a high density of records were selected. Fig. 4 indicates that most V2V encounters occurred at intersections, and most intersections recorded too few events to do an analysis. In addition, the diversity of intersections can help to gain insights in the distribution of interactive driving patterns over intersections. Based on the data sufficiency and diversity, eight representative intersections consisting of 706 driving events were finally selected, as marked on Fig. 4. Green region represents few V2V encounters at intersections while deep red represents many V2V encounters at intersections. In our case, a large intersection usually contains more scenarios than a small one. For instance, the intersection #3 equipped with traffic lights is one of the transportation hubs of Ann Arbor (Michigan), which contains 203 V2V encounter events. Inversely, the intersection #8 (a small intersection with a 2-way stop sign) only contains 53 events, which is the only intersection with stop signs and also the number of event records is higher than 50 in the database. Although other intersections with stop signs also logged some samples, the amounts of the V2V records is too low to provide efficient analysis. Besides, the recorded events with a short period can not provide sufficient information for analysis. The driving primitives with a short period provide limited information and may cause a computation problem while implementing interpolation (10). Hence, our focus is on the driving primitives with the number of data points greater than 10 (i.e., 1 s with a sampling frequency of 10 Hz). All the selected intersections are equipped with traffic lights, as shown in Table 1.

Finally, implementing the sticky HDP-HMM to the 706 driving events generates 2456 primitives. The log-likelihood value of observations can evaluate the learning performance of the sticky HDP-HMM (Wang et al., 2018b), and the predictive log-likelihood for new observations can evaluate the fidelity of the learned models (Taniguchi et al., 2014). Fig. 5 illustrates one example of the learning performance of the sticky HDP-HMM for one driving encounter and indicates that the log-likelihood performance is convergent to around -52 over iteration steps. The predictive likelihood performance of given new observations is around -61 ± 8.73 (mean and standard variance), which indicates that our model performance is satisfied. As discussed before, scaling the intersection primitives to the same length is requisite before employing DTW to generate the representative feature matrices. Determining the

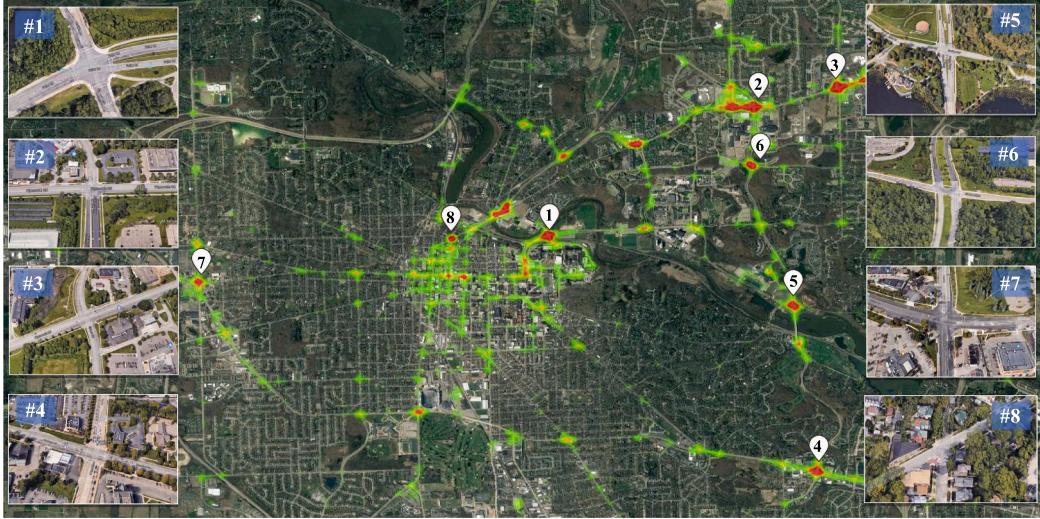


Fig. 4. Visualized distribution of the collected data at intersections in Ann Arbor, Michigan, on the Google map. Red represents a high density of data and emerald represents a low density of data. Markers are the selected intersections. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Details of the selected intersections.

ID	Location (Lat./ Long.)	Road Segments	Traffic Controls	Pedestrian	Turn Left	# of cases
#1	42°17'11"N 83°43'57"W	4-way	4-way traffic lights	Yes	Yes	99
#2	42°18'18"N 83°41'34"W	4-way	4-way traffic lights	Yes	Yes	203
#3	42°16'39"N 83°41'56"W	4-way (2 roads in N-S)	4-way traffic lights	Yes	Yes	39
#4	42°16'50"N 83°46'49"W	4-way	4-way traffic lights	Yes	Yes	28
#5	42°18'09"N 83°42'15"W	4-way	4-way traffic lights	Yes	Yes	173
#6	42°15'24"N 83°41'45"W	4-way	4-way traffic lights	Yes	Yes	55
#7	42°17'43"N 83°42'17"W	4-way (2 roads in N-S)	4-way traffic lights	Yes	Yes	56
#8	42°17'10"N 83°44'45"W	4-way	Traffic lights (E-W) & Stop signs (N-S)	Yes	Yes	53

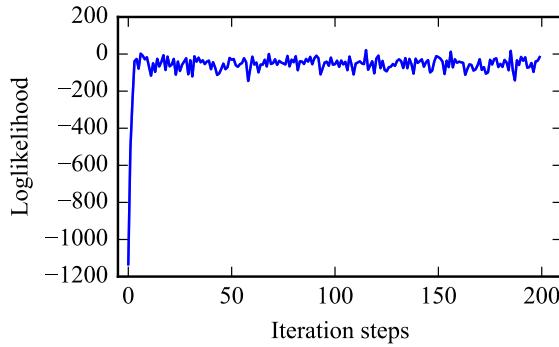


Fig. 5. The learning performance of the sticky HDP-HMM to segment one of the driving encounters.

scaling length is challenging since there exists a dilemma of computational cost and model accuracy: a short length would not adequately reflect the raw interaction patterns, while an overlarge length will enlarge the dimension of the feature vector to significantly increase computational cost. Here, the expected length l was determined according to the distribution of the duration of all extracted primitives, as shown in Fig. 6. It can be seen that most of the driving primitives have a length of around 5.00 s. In order to make a trade-off between computational cost and retrieved information, the expected primitive duration was set as 5.00 s in 10 Hz, i.e., with 50 data samples ($l = 50$) for each primitive. Thus, $M^{(p)}$ and $M^{(v)}$ will be both set as 50×50 dimensions (i.e., $l = 50$). The dimension of primitive feature vector ϕ is $\mathbb{R}^{1 \times 5000}$ after the reshaping function f by (12). The iteration number for estimating the conditional posterior probability of latent states in the sticky HDP-HMM was set as 200. Other detailed model settings can see in Wang et al. (2018b).

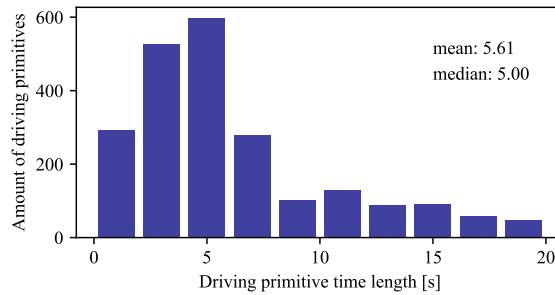


Fig. 6. Distribution of driving primitive time length. The mean time length of primitive is 5.61 s. The median time length of primitive is 5.00 s.

5. Results and analysis

5.1. Intersection primitive extraction result and analysis

Fig. 7 displays some typical examples of driving encounter scenarios with the extracted primitives. Each encounter consists of trajectory (top) and speed (bottom) profiles. Different colors in a single driving encounter represent different primitives. Colors in different intersection scenarios are independent; for example, the pink segments in Fig. 7(a), (c) and (d) are independent of each other. The results indicate that the stick HDP-HMM can automatically learn the margins of these driving primitives from complex interactive driving scenarios. According to the experimental results, three main conclusions can be drawn as follows.

- (1) The amount and type of extracted primitives vary over scenarios. The encounter behavior in Fig. 7(a), (b) and (c) consists of 3 primitives, while Figs. 7(d) only consists of one primitive. The primitives in a single event could be in the same color, that is, they have the same attributes. As shown in Fig. 7(a) and (b), both of them are segmented in three primitives with two kinds of colors (black & pink in (a) and golden & rose in (b)).
- (2) These extracted driving primitives at intersections are analytically interpretable. Fig. 7(c) displays a typical driving behavior at intersections: one vehicle (vehicle A) keeps stationary at the intersection with the engine on then turns right, while the other vehicle (vehicle B) slows down and stops at the intersection. This encounter behavior is automatically decomposed as three driving primitives, which can be semantically interpreted as follows.
 - Vehicle A keeps stationary while vehicle B decelerates first and then stops.
 - Both vehicle A and B keep stationary with the engines on.
 - Vehicle A turns right while vehicle B still does not move.
 Based on the behaviors of primitives described above, the extraction results are in line with human understanding, thus facilitating subsequent analysis and applications, for example, decision-making design for autonomous vehicles in complex scenarios (Galceran et al., 2017).
- (3) Speed value provides valuable information in V2V encounter scenarios analysis. Figs. 7(b) and (d) describe similar trajectory scenarios: vehicle A goes through the intersection and vehicle B keeps stationary. However, encounter behavior in Fig. 7(b) is decomposed into three different primitives while Fig. 7(d) itself is considered as a single primitive because the speed dominates

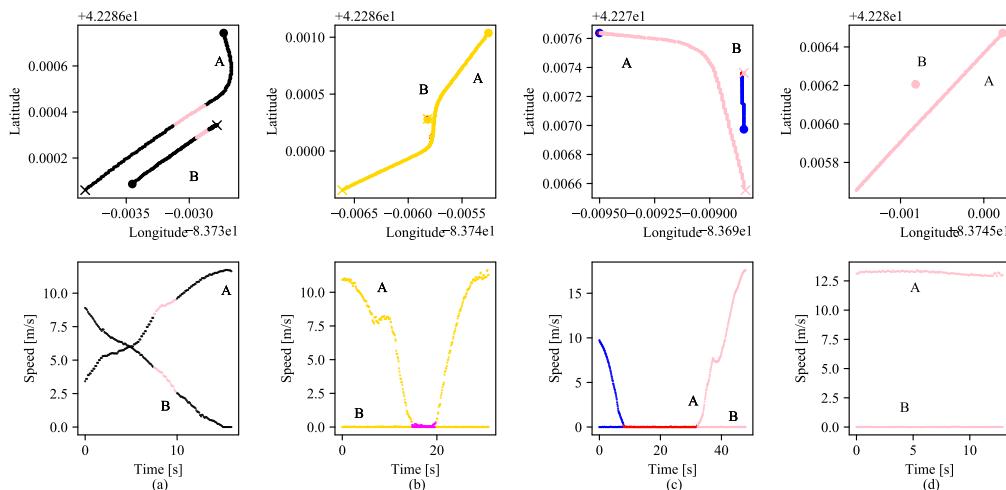


Fig. 7. Examples of four typical intersection scenarios decomposition result, consisting of vehicle trajectories (top) and vehicle speed (bottom). Capital A and B represent two vehicles, respectively. The dot and cross represent the starting points and endpoints of the trajectories of two vehicles.

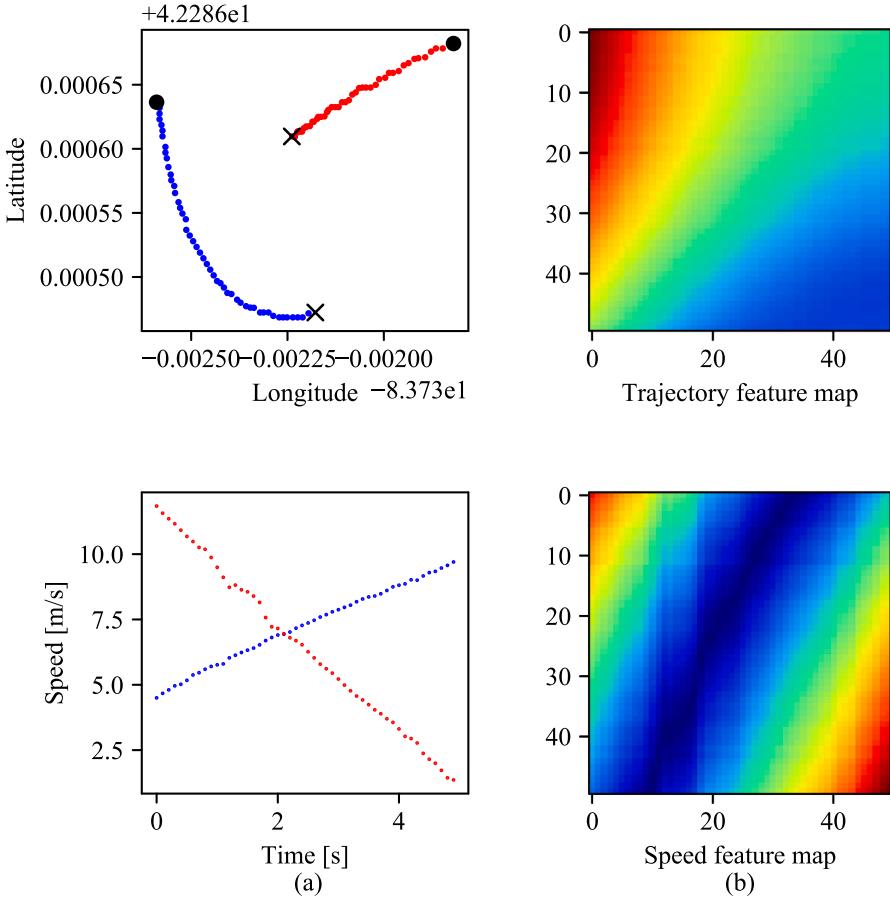


Fig. 8. Example of an intersection primitive with its trajectories and speed feature matrices. (a) The trajectory and speed of extracted primitives. (b) The feature matrices calculated by DTW.

the difference: The speed of vehicle A in Fig. 7(b) has a higher variance than the vehicle A in Fig. 7(d). Hence, speed information is the critical component to distinguish the intersection scenarios with similar position trajectories, which could enrich decomposition results.

According to the analysis above, the complex intersection scenarios can be decomposed into some semantically interpretable primitives via the sticky HDP-HMM.

5.2. Driving primitive feature analysis

Fig. 8(a) represents the trajectories of extracted primitives regarding position trajectories and speeds of two vehicles at an intersection after scaling. Blue and red lines represent the two engaged vehicles. Fig. 8(b) shows the feature matrices of its normalized trajectories feature matrix (top) and speed feature matrix (bottom) by DTW. Dark blue represents a high value, and dark red represents a low value. Taking the speed feature matrix (Fig. 8(b)) as an example, the color changes from dark red to dark blue and then to dark red again over the main diagonal, which indicates the speed difference between the two vehicles is changing from large to small and then to large again. For instance, the crossover of the speed of two vehicles occurred at around 2 s is represented as a distinct dark blue area in the middle of speed feature map, indicating a minor difference.

As the above discussed, the spatiotemporal dynamic interactions of two vehicles can be reserved in the feature matrices through DTW. Also, the feature map allows us to visualize the relationship between two vehicles and thus facilitate to compare and analyze the differences among individual primitives.

5.3. Clustering result evaluation and analysis

A variety of primitives were extracted from driving encounter scenarios using the sticky HDP-HMM. In order to figure out how many types of primitives exist in the data to further analyze the interactive driving patterns at intersections, the k -means is used to cluster these primitives into groups. The temporal-spatial features for training the k -means algorithm are computed via DTW.

Determining the number of clusters is an essential step in k -means clustering. The clustering aims to gather objects into

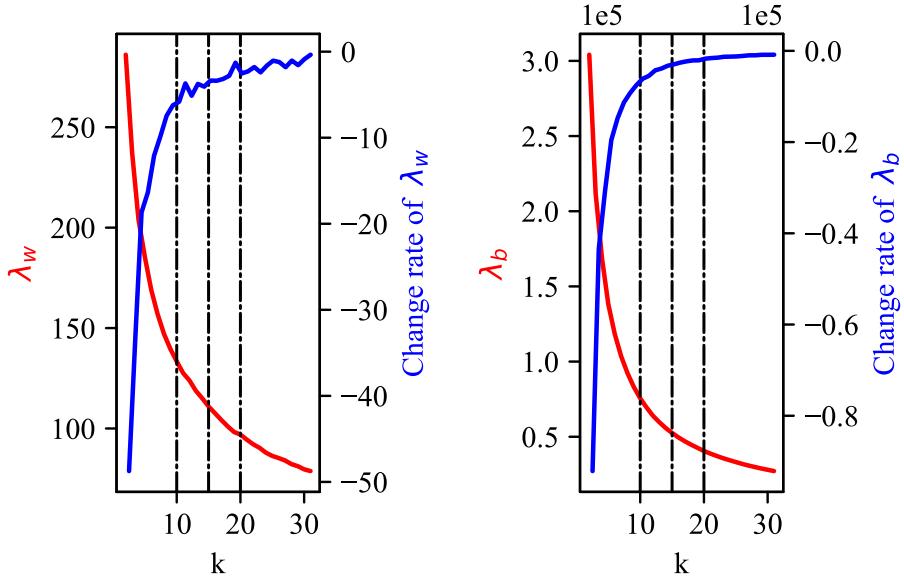


Fig. 9. Results of within distance λ_w and between distance λ_b over the number of clusters k .

homogeneous groups that are far from one-another. Hence, the method to evaluate clustering performance is based on the changes and change rates of within-clusters distance λ_w and between-clusters distance λ_b over k . Fig. 9 shows that both λ_w and λ_b decrease when increasing k , and their change rates get closed to zero gradually, indicating the clustering performance is convergent. In order to get satisfied clustering results and prevent suffering an excessive computational cost, the elbow point ($k = 15$) of the changing rate of both λ_w and λ_b is selected.

After selecting the number of clusters, the distribution of these primitives is also analyzed, as shown in Fig. 10. Two most typical primitives are in the clusters #2 and #6, accounting for 12.05% (296/2456) and 14.74% (362/2456) of total intersection primitives, respectively. Two least primitives are in the clusters #10 and #14, the number of both is less than 100, accounting for about 7.0% of total driving primitives. The amounts of primitives in the rest 13 clusters are in the range of (100, 200).

In order to do further semantic analysis about the interactive driving behavior at intersections, six primitives are randomly selected from six different clusters, covering two most common clusters (#2 and #6), two least common clusters (#10 and #14), and two clusters (#4 and #12) that are randomly selected from the rest 11 clusters. Fig. 11 displays their trajectory and speed profiles and associated feature maps. Red and blue dot lines represent two vehicles' trajectories. In what follows, a detailed analysis of the results will be presented.

5.3.1. Two most common primitives

Fig. 11(a) (cluster #2) represents the interactive pattern wherein both two vehicles keep stationary during the whole period, which is a typical behavior: two vehicles stop at intersections, for example, in response to the red traffic light. The zero drift of speed

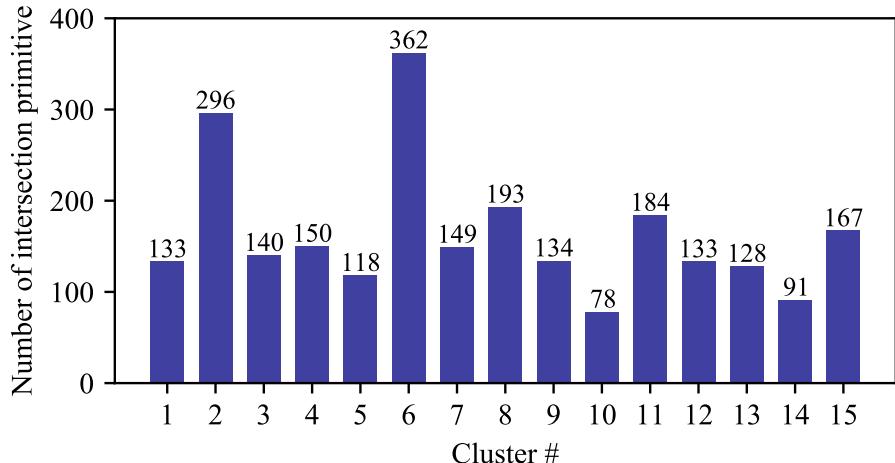


Fig. 10. Distribution of the number of primitives in each cluster calculated via k -means.

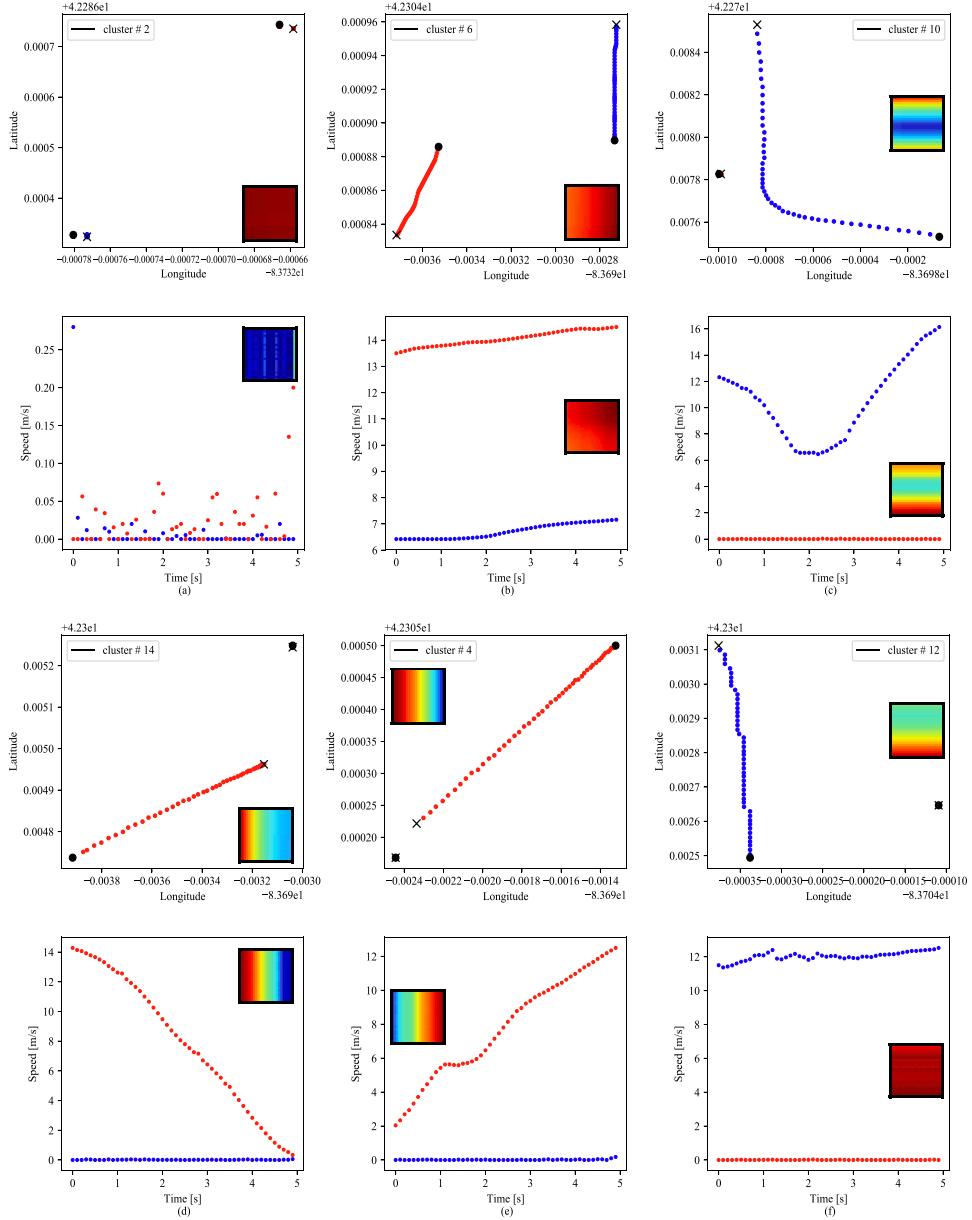


Fig. 11. Trajectories and speed plots of 6 intersection primitives with their own feature map which are randomly selected from 6 clusters. Dot and cross represent the starting point and the ending point, respectively.

sensors causes the fluctuation of the speed in Fig. 11(a). Fig. 11(b) (cluster #6) describes the interactive pattern: the vehicle marked as red is departing from the intersection at a relatively high speed, while the other vehicle marked as blue is approaching the intersection with deceleration. Both two vehicles are heading toward the opposite direction, which is a common encountering behavior occurred at intersections.

5.3.2. Two least common primitives

Fig. 11(c) (cluster #10) represents a relatively rare type of driving patterns: one vehicle marked as blue is decelerating to approach the intersection first and then accelerates to turn right to move away, while the other vehicle marked in red is always keeping stationary with the engine on during the whole period. Fig. 11(d) (cluster #14) displays the other uncommon primitives, where the vehicle in red is approaching the intersection with speed gradually decreasing to zero while the vehicle marked as blue does not move during the whole period.

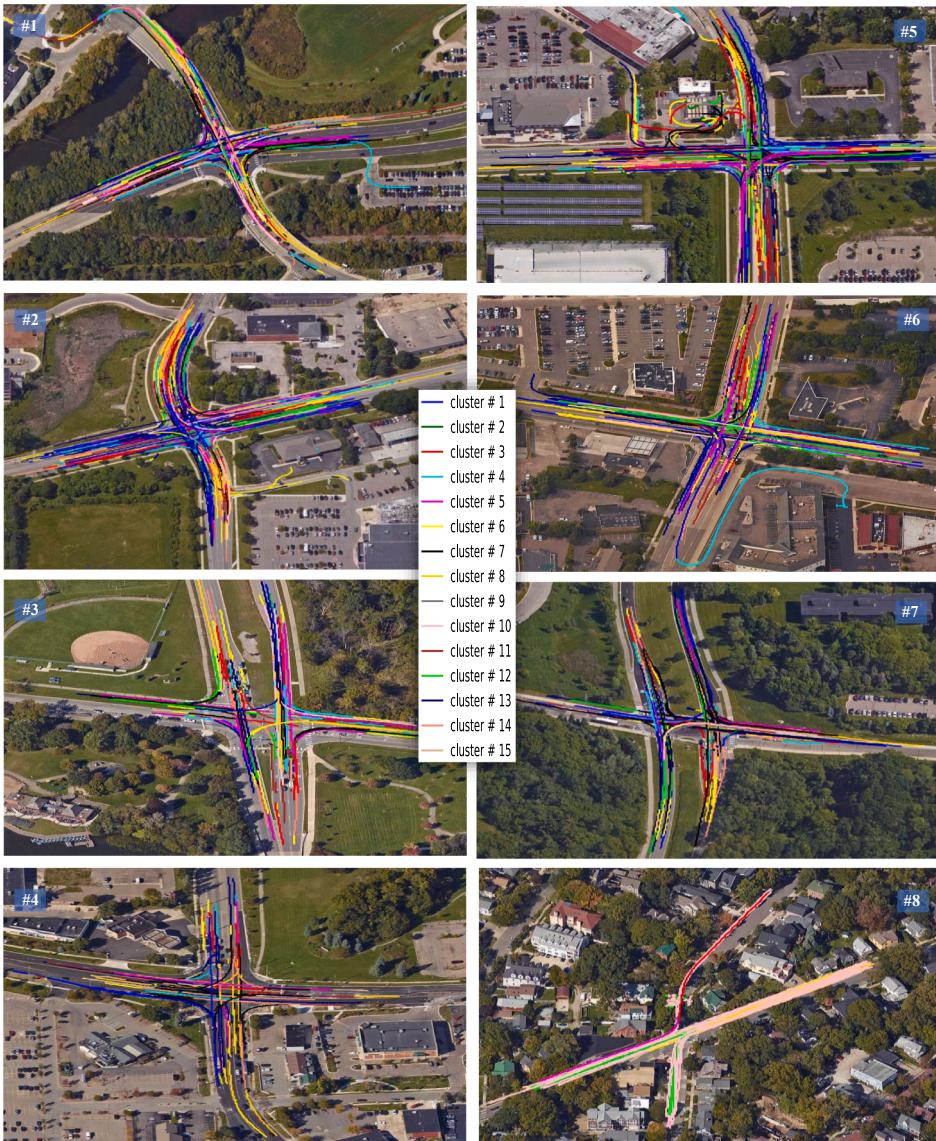


Fig. 12. GPS data of all driving primitives at 8 intersections visualized on Google map with clustering results marked with 15 different colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.3.3. Two randomly selected primitives

Except for the primitives displayed and discussed above, the rest 11 clusters have similar amounts of primitives. Two primitives are randomly selected for in-depth analysis, as shown in Fig. 11(e) (cluster #4) and (f) (cluster #12). Cluster #4 represents the behavior that one vehicle (marked as red) accelerates slightly to approach the intersection while the other vehicle (marked as blue) keeps stationary. Cluster #12 shows the behavior that the vehicle marked as blue crosses the intersection at a constant speed, while the vehicle marked as red does not move.

In order to show all the results, all primitives occurred at eight intersections are visualized separately with the specified colors corresponding to the clusters on the Google map (Fig. 12). The intersection primitives belong to the same cluster have the same color.

5.4. Difference analysis between intersections

After analyzing the typical interactive driving patterns, the distribution of primitives over intersections is then investigated. Fig. 13 displays the distribution of primitives specific at each intersection. It is easy to observe that the distribution of primitives at intersection #8 is significantly different from others. More specifically, the most common primitive occurred at intersection #8 is cluster #10, accounting for 48.72% out of all primitives, which is the least normal driving primitive patterns at other intersections. It describes the interactive driving behavior that one vehicle turns right at the intersection while the other vehicle keeps stationary. The

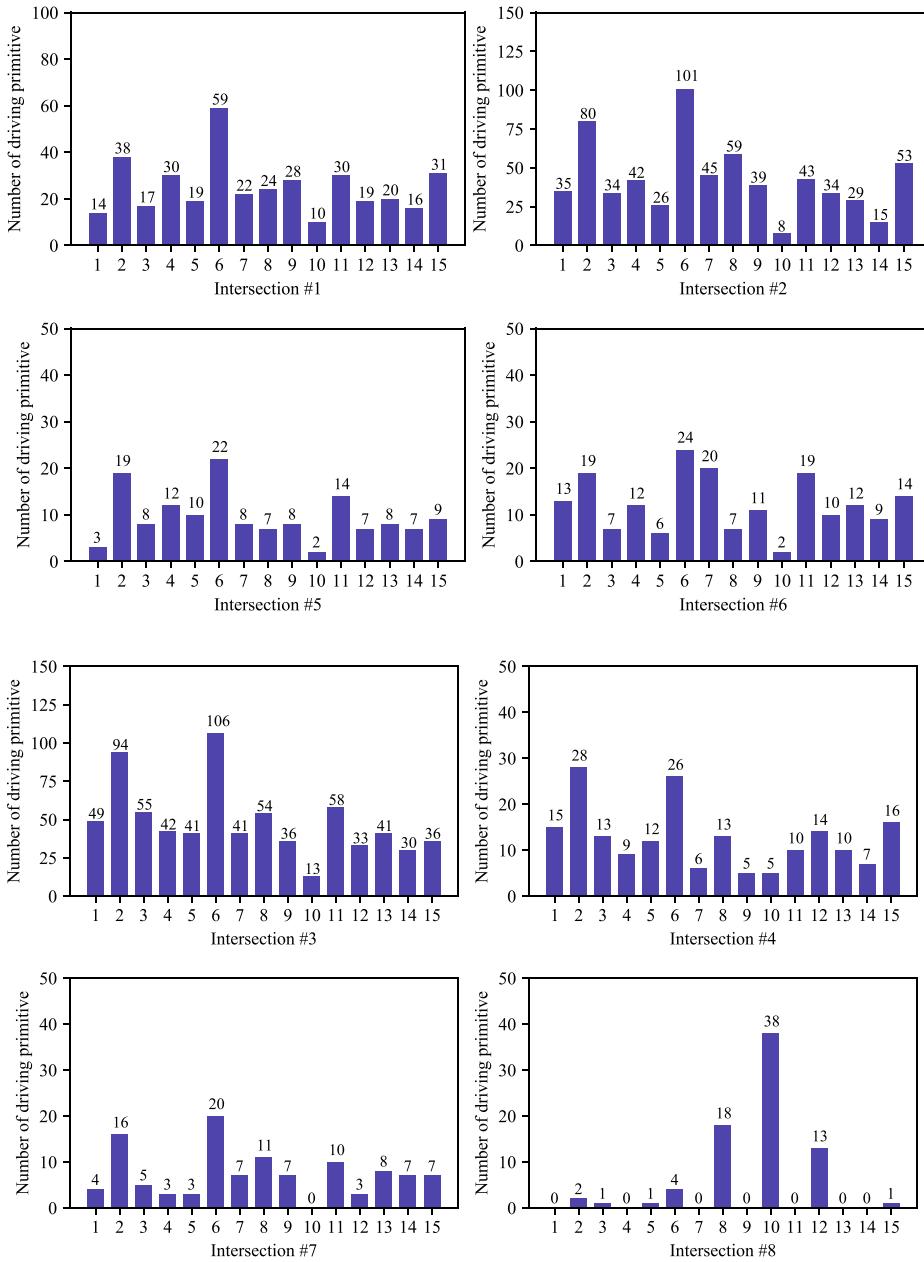


Fig. 13. Distribution of the number of primitives in each cluster at each intersection.

quantitative evaluation of the difference among clusters is based on the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951), which can gauge the similarity between cluster #8 and clusters #1 ~ #7, calculated by

$$D_{KL} = \sum_{i=1}^{n_c} Q_d(i) \ln \frac{Q_d(i)}{P_d(i)} \quad (16)$$

where $P_d(i)$ and $Q_d(i)$ represent the distribution of the i -th cluster at intersections P and Q , respectively, and n_c is the total amount of clusters with $\sum_{i=1}^{n_c} P_d(i) = \sum_{i=1}^{n_c} Q_d(i) = 1$. A higher value of D_{KL} indicates a larger divergence at P intersection and Q intersection regarding the distribution of primitives.

Here, the intersection #1 is treated as the baseline and calculate the divergence between each pair of intersection against the baseline, and the results are shown in Table 2. It can be seen that the KL value for intersection #8 is ten times higher than the KL values for intersections #2 to #7, compared to intersection #1. It indicates that there is a big difference in the distribution of interactive driving patterns between intersection #8 and intersections #1 ~ #7 because the intersections #1 to #7 have heavy traffic

Table 2

KL divergence of each intersection compared with intersection #1.

ID	#1	#2	#3	#4	#5	#6	#7	#8
KL	0	0.0286	0.0412	0.0817	0.0263	0.0556	0.0929	1.765

flow and all of them are equipped with traffic lights, while intersection #8 has a low traffic flow with two-way stop signs. Therefore, the distribution of interactive driving patterns can help distinguish the type of intersections.

6. Conclusion

This paper provided a primitive-based approach with unsupervised learning to learn the interactive driving patterns at intersections. The Bayesian nonparametric learning was employed to segment the intersection driving scenarios into driving primitives. Experiment results from naturalistic driving data indicate that the interaction patterns at intersections could be decomposed into finite kinds of semantically interpretable groups. The distributions of driving primitives at different intersections were investigated, which demonstrated that the distribution of interactive driving patterns could be used as an indicator to identify the types of intersections. The primitive-based framework presented in this paper is suitable to do the semantic analysis for V2V interaction at intersections with traffic lights. Although all the intersections in our paper are with traffic light because of the limitation of data, our proposed framework could be easily extended to the unsignalized intersections, which will be one of our future work.

The Bayesian nonparametric method developed in this paper is based on a mathematically rigorous framework, which can be used to analyze other sequential traffic data such as traffic accident data. The collected data in this paper only consists of the vehicles' trajectory and speed without other potential information. Hence, our future work will be articulated around two axes. The first one is to extend the developed framework to other traffic scenarios with road users' interaction, for example, vehicle-to-pedestrian, vehicle-to-cyclist, and pedestrian-to-cyclist, and their combination. In this way, more features of vehicles presenting at intersections can be extracted, thus allowing us to take a further analysis of complex interactions among road users. The second objective is to consider other factors that could impact interactions between human drivers such as weather, traffic light, and intersection types. Semantically understanding the relations between these factors and interactive driving patterns at intersections could provide a set of recognizable discrete states about complex dynamic systems, thereby benefiting for decision-making design to guarantee self-driving safety and traffic efficiency.

References

- Administration, N.H.T.S., 2018. Fatality analysis reporting system encyclopedia. (accessed on 2018).
- Aoude, G.S., Desaraju, V.R., Stephens, L.H., How, J.P., 2011. Behavior classification algorithms at intersections and validation using naturalistic data. In: Intelligent Vehicles Symposium (IV), 2011 IEEE. IEEE, pp. 601–606.
- Aoude, G.S., Desaraju, V.R., Stephens, L.H., How, J.P., 2012. Driver behavior classification at intersections and validation on large naturalistic data set. *IEEE Trans. Intell. Transp. Syst.* 13 (2), 724–736.
- Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Pérez, J.M., Perona, I., 2013. An extensive comparative study of cluster validity indices. *Pattern Recogn.* 46 (1), 243–256.
- Ben-Hur, A., Horn, D., Siegelmann, H.T., Vapnik, V., 2001. Support vector clustering. *J. Mach. Learn. Res.* 2 (Dec), 125–137.
- Bezzina, D., Sayer, J., 2015. Safety pilot model deployment: Test conductor team report. NHTSA Report DOT HS 812 171, National Highway Traffic Safety Administration (NHTSA). U.S. Department of Transportation.
- Bhatt, D., Sodhi, D., Pal, A., Balasubramanian, V., Krishna, M., 2017. Have i reached the intersection: a deep learning-based approach for intersection detection from monocular cameras. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 4495–4500.
- Brechtl, S., Gindel, T., Dillmann, R., 2014. Probabilistic decision-making under uncertainty for autonomous driving using continuous pomdps. In: Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on. IEEE, pp. 392–399.
- Butakov, V.A., Ioannou, P., 2015. Personalized driver/vehicle lane change models foradas. *IEEE Trans. Veh. Technol.* 64 (10), 4422–4431.
- Creaser, J.I., Rakauskas, M.E., Ward, N.J., Laberge, J.C., Donath, M., 2007a. Concept evaluation of intersection decision support (ids) system interfaces to support drivers' gap acceptance decisions at rural stop-controlled intersections. *Transp. Res. Part F: Traff. Psychol. Behav.* 10 (3), 208–228.
- Creaser, J.I., Rakauskas, M.E., Ward, N.J., Laberge, J.C., Donath, M., 2007b. Concept evaluation of intersection decision support (ids) system interfaces to support drivers' gap acceptance decisions at rural stop-controlled intersections. *Transp. Res. Part F: Traff. Psychol. Behav.* 10 (3), 208–228.
- Essa, M., Sayed, T., 2018. Traffic conflict models to evaluate the safety of signalized intersections at the cycle level. *Transp. Res. Part C: Emerg. Technol.* 89, 289–302.
- Fox, E.B., Sudderth, E.B., Jordan, M.I., Willsky, A.S., 2011. A sticky hdp-hmm with application to speaker diarization. *Ann. Appl. Stat.* 1020–1056.
- Fu, Y., Li, C., Luan, T.H., Zhang, Y., Mao, G., 2018. Infrastructure-cooperative algorithm for effective intersection collision avoidance. *Transp. Res. Part C: Emerg. Technol.* 89, 188–204.
- Gadeppally, V., Krishnamurthy, A., Ozguner, U., 2014. A framework for estimating driver decisions near intersections. *IEEE Trans. Intell. Transp. Syst.* 15 (2), 637–646.
- Galceran, E., Cunningham, A.G., Eustice, R.M., Olson, E., 2017. Multipolicy decision-making for autonomous driving via changepoint-based behavior prediction: theory and experiment. *Auton. Robot.* 41 (6), 1367–1382.
- Gindel, T., Brechtl, S., Dillmann, R., 2015. Learning driver behavior models from traffic observations for decision making and planning. *IEEE Intell. Transp. Syst. Mag.* 7 (1), 69–79.
- Grembek, O., Kurzhanskiy, A., Medury, A., Varaiya, P., Yu, M., 2019. Making intersections safer with i2v communication. *Transp. Res. Part C: Emerg. Technol.* 102, 396–410.
- Guha, S., Rastogi, R., Shim, K., 1998. Cure: an efficient clustering algorithm for large databases. In: ACM Sigmod Record., vol. 27. ACM, pp. 73–84.
- Havlak, F., Campbell, M., 2014. Discrete and continuous, probabilistic anticipation for autonomous robots in urban environments. *IEEE Trans. Rob.* 30 (2), 461–474.
- Isele, D., Rahimi, R., Cosgun, A., Subramanian, K., Fujimura, K., 2018. Navigating occluded intersections with autonomous vehicles using deep reinforcement learning. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 2034–2039.
- Jain, A.K., 2010. Data clustering: 50 years beyond k-means. *Pattern Recogn. Lett.* 31 (8), 651–666.
- Kasper, D., Weidl, G., Dang, T., Breuel, G., Tamke, A., Wedel, A., Rosenstiel, W., 2012. Object-oriented bayesian networks for detection of lane change maneuvers. *IEEE Intell. Transp. Syst. Mag.* 4 (3), 19–31.

- Kullback, S., Leibler, R.A., 1951. On information and sufficiency. *Ann. Math. Stat.* 22 (1), 79–86. <https://doi.org/10.1214/aoms/1177729694>.
- Liang, Y., Lee, J.D., 2014. A hybrid bayesian network approach to detect driver cognitive distraction. *Transp. Res. Part C: Emerg. Technol.* 38, 146–155.
- Liao, T.W., 2005. Clustering of time series data—a survey. *Pattern Recogn.* 38 (11), 1857–1874.
- Lutzeler, M., Dickmanns, E.D., 2000. Ems-vision: recognition of intersections on unmarked road networks. In: Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat. No. 00TH8511). IEEE, pp. 302–307.
- Majone, G., 1970. Distance-based cluster analysis and measurement scales. *Qual. Quant.* 4 (1), 153–164.
- Meijering, E., 2002. A chronology of interpolation: from ancient astronomy to modern signal and image processing. *Proc. IEEE* 90 (3), 319–342.
- Möller-Levet, C.S., Klawonn, F., Cho, K.-H., Wolkenhauer, O., 2003. Fuzzy clustering of short time-series and unevenly distributed sampling points. In: International Symposium on Intelligent Data Analysis. Springer, pp. 330–340.
- Müller, M., 2007. Dynamic time warping. Information retrieval for music and motion 69–84.
- Nechyba, M.C., Xu, Y., 1998. Stochastic similarity for validating human control strategy models. *IEEE Trans. Robot. Autom.* 14 (3), 437–451.
- Sarkar, A., Czarnecki, K., Angus, M., Li, C., Waslander, S., 2017. Trajectory prediction of traffic agents at urban intersections through learned interactions. In: Intelligent Transportation Systems (ITSC), 2017 IEEE 20th International Conference on. IEEE, pp. 1–8.
- Schwarting, W., Alonso-Mora, J., Rus, D., 2018. Planning and decision-making for autonomous vehicles. *Ann. Rev. Control Robot. Auton. Syst.* 1, 187–210.
- Tang, K., Zhu, S., Xu, Y., Wang, F., 2016. Modeling drivers' dynamic decision-making behavior during the phase transition period: an analytical approach based on hidden markov model theory. *IEEE Trans. Intell. Transp. Syst.* 17 (1), 206–214.
- Taniguchi, T., Nagasaka, S., Hitomi, K., Takenaka, K., Bando, T., 2014. Unsupervised hierarchical modeling of driving behavior and prediction of contextual changing points. *IEEE Trans. Intell. Transp. Syst.* 16 (4), 1746–1760.
- Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M., 2005. Sharing clusters among related groups: Hierarchical dirichlet processes. In: Advances in Neural Information Processing Systems, pp. 1385–1392.
- Wang, W., Liu, C., Zhao, D., 2017a. How much data are enough? a statistical approach with case study on longitudinal driving behavior. *IEEE Trans. Intell. Veh.* 2 (2), 85–98.
- Wang, W., Ramesh, A., Zhao, D., 2018a. Clustering of driving scenarios using connected vehicle datasets. arXiv: 1807.08415.
- Wang, W., Xi, J., Chong, A., Li, L., 2017b. Driving style classification using a semisupervised support vector machine. *IEEE Trans. Hum.-Mach. Syst.* 47 (5), 650–660.
- Wang, W., Xi, J., Zhao, D., 2018b. Driving style analysis using primitive driving patterns with bayesian nonparametric approaches. *IEEE Trans. Intell. Transp. Syst.* 20 (8), 2986–2998.
- Wang, W., Zhao, D., 2018. Extracting traffic primitives directly from naturalistically logged data for self-driving applications. *IEEE Robot. Automat. Lett.* 3 (2), 1223–1229.
- Wang, W., Zhao, D., Xi, J., 2018c. Learning and inferring a driver's braking action in car-following scenarios. *IEEE Trans. Veh. Technol.* 67 (5), 3887–3899.
- Wei, J., Dolan, J.M., Snider, J.M., Litkouhi, B., 2011. A point-based mdp for robust single-lane autonomous driving behavior under uncertainties. In: Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, pp. 2586–2592.
- Xu, D., Tian, Y., 2015. A comprehensive survey of clustering algorithms. *Ann. Data Sci.* 2 (2), 165–193.
- Zhang, T., Ramakrishnan, R., Livny, M., 1996. Birch: an efficient data clustering method for very large databases. In: ACM Sigmod Record, vol. 25. ACM, pp. 103–114.