Article

A Scenario-Adaptive Driving Behavior Prediction Approach to Urban Autonomous Driving

Xinli Geng [1,2], Huawei Liang [2,*], Biao Yu [2], Pan Zhao [2], Liuwei He [1,2] and Rulin Huang [1,2]

[1] Department of Automation, University of Science and Technology of China, Hefei 230026, China; gxinli@mail.ustc.edu.cn (X.G.); hlw1991@mail.ustc.edu.cn (L.H.); ruling@mail.ustc.edu.cn (R.H.)

[2] Institute of Applied Technology, Hefei Institute of Physical Science, Chinese Academy of Sciences, Hefei 230008, China; byu@hfcas.ac.cn (B.Y.); pzhao@hfcas.ac.cn (P.Z.)

* Correspondence: hwliang@iim.ac.cn; Tel.: +86-551-6539-3191

Abstract: Driving through dynamically changing traffic scenarios is a highly challenging task for autonomous vehicles, especially on urban roadways. Prediction of surrounding vehicles' driving behaviors plays a crucial role in autonomous vehicles. Most traditional driving behavior prediction models work only for a specific traffic scenario and cannot be adapted to different scenarios. In addition, priori driving knowledge was never considered sufficiently. This study proposes a novel scenario-adaptive approach to solve these problems. A novel ontology model was developed to model traffic scenarios. Continuous features of driving behavior were learned by Hidden Markov Models (HMMs). Then, a knowledge base was constructed to specify the model adaptation strategies and store priori probabilities based on the scenario's characteristics. Finally, the target vehicle's future behavior was predicted considering both a posteriori probabilities and a priori probabilities. The proposed approach was sufficiently evaluated with a real autonomous vehicle. The application scope of traditional models can be extended to a variety of scenarios, while the prediction performance can be improved by the consideration of priori knowledge. For lane-changing behaviors, the prediction time horizon can be extended by up to 56% (0.76 s) on average. Meanwhile, long-term prediction precision can be enhanced by over 26%.

Keywords: autonomous vehicle; scenario-adaptive; driving behavior prediction; ontology model

1. Introduction

Autonomous vehicles (AVs), as one of the most promising topics in both academia and industry in recent decades, are expected to change the future of the world in various ways [1]. AVs can benefit the next generation of transportation systems due to their safety, mobility, and efficiency advantages [2]. Moreover, AVs are effective test platforms for various advanced technologies, such as artificial intelligence, automation control, machine learning, and so on [1]. Based on efforts from around the world, related technologies have been studied in depth [1,3]. Considerable related applications have been tested in controlled environments and applied in commercial vehicles [3]. For example, Google's AVs have driven more than 1.2 million miles on public roads [4]. All Tesla vehicles in production are equipped with all the hardware needed for fully autonomous capability and newer applications will gradually be activated [5]. Although preliminary successes have been achieved with AVs, there are still many intractable issues from both a technical and a societal perspective [2,6].

Generally, the architecture of AVs can be divided into three layers, i.e., a perception layer, a planning layer, and a trajectory control layer [1]. The perception layer aggregates real-time data from multiple sensors to estimate the surrounding scenarios. The trajectory control layer generates commands to control the vehicle according to the planned trajectory and velocity. As the core module,

planning layer usually consists of three modules, i.e., route planning, behavior decision-making, and path planning. Route and path planning module generate proper global routes and safe local trajectories. The behavior decision-making module provides safe and reasonable abstract driving actions (e.g., "change lane to left"). The state-of-the-art decision-making system of AVs should have a forward-looking ability, which means that the surrounding vehicles' future driving behavior should be predicted accurately [7]. Without the prediction function, emergency incidents may happen, such as the collision between MIT's "Talos" AV and Cornell's "Skynet" AV during the 2007 urban challenge, which was held by the Defense Advanced Research Projects Agency (DAPRA) [3].

As mentioned above, driving behavior prediction plays a significant role in AVs' decision-making system. However, as latent information, the surrounding vehicles' future behaviors cannot be directly measured by any available sensors or obtained by inter-vehicle communication technologies (e.g., Connected Vehicles (CV)-related technologies have not been widely deployed) [8].

In order to predict surrounding vehicles' driving behavior, AVs should monitor the variation in the target vehicle's movement and understand the current scenario from the view of the target vehicle. Then, the future behavior can be predicted by prediction models. Usually, three types of information are applied, i.e., vehicle kinematics, the relationships between target vehicle and surrounding entities (e.g., other vehicles, lane lines), and a priori knowledge. Vehicle kinematics refers to the target vehicle's movement states, such as the lateral velocity and acceleration. The interactions with road elements and other vehicles are described by relative distance, velocity, and direction from the target vehicle to a given entity. These relationships can be characterized by continuous values (e.g., 10 meters) or logical sentences (e.g., "*isCloseTo*"). At each time step, for a given target vehicle, a traffic scenario can be defined by these relationships. Each relationship can be considered as one scenario-dependent characteristic. A diversity of scenarios can be defined for the variation of these characteristics. The interactions between the target vehicle and road entities or other vehicles are regulated by a priori knowledge (e.g., traffic rules and commonsense knowledge of driving) [9]. For example, it can be believed that a vehicle moving in the leftmost lane has a very low probability of turning left, as otherwise it will crash into the curb.

Without an inborn ability to perceive and interpret traffic scenarios accurately predicting driving behavior for typical urban roadways is quite challenging for AVs [10]. Most existing models for driving behavior prediction were based on one or multiple types of information mentioned in the previous paragraph. Table 1 presents related up-to-date studies [11–20] that predict surrounding vehicles' driving behaviors. In the table, existing related studies are categorized into three groups according to the features applied in the study.

Table 1. Comparative review of works on driver behavior prediction for surrounding vehicles [1].

Involved Features	Scenario Adaptable	Used Methods [2]	Implemented in
Vehicle kinematics, Relations with road entities	no	TTx HMM SVM-BF	[12,16] [12,16] [12,20]
Vehicle kinematics, Relations with road entities, Interactions with other vehicles	no	BN Layered-HMM	[11] [15]
Vehicle kinematics, Relations with road entities, Interactions with other vehicles, A priori knowledge	no no Yes Yes	DBN Cost-map + BF HSS-HMM SMT + SLP	[13] [14] [17,18] [19]

[1] Driving behavior prediction studies based on drivers' features (like driver's head movement, driving style, etc.) were neglected as they cannot be detected by AVs. [2] TTx: Time To Something; HMM: Hidden Markov Model; SVM-BF: Support Vector Machines-Bayesian Filter; BN: Bayesian Network; DBN: Dynamic Bayesian Network; HSS-HMM: Hybrid State System-Hidden Markov Model; SMT + SLP: Scenario Model Tree + Single-Layer Perceptron.

Vehicle kinematics and relations with road entities were considered by almost all existing studies. These features can be applied directly by TTx (Time to something) models to predict the behavior of the target vehicle. For example, Lefèvre et al. used TTC (Time to Line Crossing) to predict whether the vehicle will depart from the current lane or not [16]. These features can also be pre-learned by models and then we can find the most similar behavior during the on-line prediction phase by pattern matching. The Hidden Markov Model (HMM) and Support Vector Machines (SVM) with Bayesian Filter (BF) are usually applied to learn the pattern of each behavior [12,20]. Besides vehicle kinematics and relations with road entities, interactions between the target vehicle and its surrounding vehicles also have a significant influence on target vehicles' future behavior. A Dynamic Bayesian Network (DBN) was applied widely to develop interaction-aware prediction models. The interaction-aware models were also believed to be the most comprehensive according to the study of Lefèvre et al. [9]. However, their applications in AVs are limited because of the expensive computation [9]. In order to relieve the curse of dimensionality, Bahram et al. [14] proposed a long-term behavior prediction framework in which model-based and supervised-learning-based subtasks were combined. A complex spatial–temporal cost map with consideration of traffic rules was constructed to determine the interactions. However, in this study, just 60 lane changes of other traffic participants were considered and it is difficult to construct a cost map. All the previous studies tried to build a driving behavior prediction approach with consideration of as many clues as possible, but most previous models fail to consider all the clues. In particular, a priori knowledge is never even considered.

Most mentioned models were designed for one specific scenario, such as lane-changing models for a non-junction segment, turn left/right models for a junction area, etc. However, AVs drive through dynamically changing environments in which a diversity of scenarios show up over time. Multiple scenario-specific models should be used and appropriate models should be activated according to the scenario characteristics, which means that a reasonable adaptation approach should be designed. Aiming to solve the adaptation issue, Gadepally et al. proposed a Hybrid State System and Hidden Markov Model (HSS + HMM) to graft and prune "meta-states" (a group of candidate models) [17,18]. The different scenarios were specified by Finite State Machine (FSM) with scenario-dependent features. However, this model was applied for driving behavior estimation (estimating behavior after it has actually happened) and not for prediction. The FSM will become untraceable for large-scale scenarios due to its inherent disadvantages. Another study [19] proposed a hierarchy scenario model tree (SMT) to specify different scenarios and used single-layer perceptron (SLP) to learn scenario-specified behaviors. For both of these related studies, the traffic rules and drivers' driving experience were not considered and so it is difficult to integrate them into their proposed models, such as "vehicles on a smaller road yield to vehicles on the main road". Furthermore, simply scenario representation methods were applied and only the relationships between two vehicles were considered. However, the target vehicles' future behavior is influenced by multitudinous surrounding entities.

Based on the literature review and the comparison of existing methods, the main objects of this study are summarized as:

- Improving the adaptability of existing models.

In the real world, AVs should drive through a diversity of complex urban scenarios. A reliable activation strategy that can select appropriate candidate prediction models for a given scenario will extend the application scope of the traditional prediction models.

- Combining priori knowledge to improve the quality of prediction.

Priori knowledge information is helpful for driving behavior prediction [9]. For example, it is believed that a moving vehicle has quite a low probability of changing lanes to the right when there is another vehicle abreast in the right lane. Based on the literature review, it is still not possible to describe these relationships clearly without increasing the computation complexity.

Aiming at the aforementioned goals, this study proposed a scenario-adaptive approach to predict driving behaviors on urban roadways. The main contributions of this study are as follows:

- A novel ontology-based model was proposed to represent the traffic scenarios.

This study developed a novel ontology-based model to represent the spatial state of surrounding entities and their inter-relationships. Based on this model, the scenario can be expressed explicitly and semantic inter-relationships can be represented, which makes it more understandable than traditional scenario description approaches.

- A rule-based reasoning system was developed to improve the adaptability of the driving behavior prediction models.

Based on the structured description of traffic scenarios, adaptation rules were edited. In the rules, candidate scenario-specific models, input features, and priori probabilities for candidate models were specified based on the scenario characteristics.

- The future behavior of the target vehicle was determined by both a priori and a posteriori knowledge.

This is the first study to combine a priori probability and a posteriori probability for behavior prediction. HMM-based probabilistic models were trained off-line and tested on-line to get the a posteriori probability for each possible behavior. A priori probability is provided by a rule-based reasoning system.

In the following sections, the problem formulation and an overview of the proposed approach are presented. The ontology model to describe traffic scenarios and the HMMs to learn continuous features are presented. Then, the approach to predict the future behavior of a target vehicle is given. Finally, the proposed approach is evaluated in real traffic scenarios on urban roadways. The results of the experiments are presented and discussed. We finally conclude this paper with contributions, limitations, and future work in Section 4.

2. Materials and Methods

2.1. Problem Formulation and Method Overview

2.1.1. Problem Formulation

The driving behavior prediction algorithm should be running for each target vehicle at every time step. The prediction process can be divided into three steps: First, for target vehicle i at time t, the characteristics of its current scenario s^i need to be extracted and analyzed. Then, pre-trained driving behavior models $M_{s^i}^i$, input features $F_{s^i}^i$, and a priori probabilities $P_{s^i}^i$ for $M_{s^i}^i$ should be specified. Finally, one of the pre-defined maneuvers b^i should be assigned for the next time steps by

$$b^i = \mathcal{M}(F_{s^i}^i, M_{s^i}^i, P_{s^i}^i). \tag{1}$$

This process can also be named "behavior tagging," tagging a sequence of observations with a similar driving behavior b^i. b^i belongs to a finite set of predefined maneuvers:

$$b^i \in \mathfrak{B} := \{LK, LCL, LCR, TR, TL, GS, SS, ...\}, \tag{2}$$

corresponding to lane keeping (LK), lane change to left (LCL), lane change to right (LCR), turn right (TR), turn left (TL), go straight at intersection (GS), stop before the stop line (SS). In this study, we just considered the aforementioned lateral behaviors and more behaviors can be added easily. The candidate driving behavior prediction models $M_{s^i}^i$ are chosen based on s^i from the complete model set:

$$M_{s^i}^i \subset \mathfrak{M} := \{M_{LK}, M_{LCR}, M_{TR}, ...\}, \tag{3}$$

where $\mathfrak{M}$ contains all the possible driving behavior models, in which one behavior model corresponds to one type of behavior. The models' input features $F_{s^i}^i$ are determined based on s^i and $M_{s^i}^i$, and belong to the pre-defined feature set:

$$F_{s^i}^i \subset \mathfrak{F}, \tag{4}$$

where $\mathfrak{F}$ is a pre-defined feature set, including all the possible input features for all the models in $\mathfrak{M}$. In this study, the input features that contribute more prediction power were selected. For the lane-changing behaviors (LK, LCR, LCL), there are two features. The first one is the difference $d_{diff-lane}$ between distance to right lane line $d_{right-lane}$ and distance to left lane line $d_{left-lane}$. The second is the lateral velocity v_{lat} of the target vehicle. These two features have been verified as the most important features for lane-changing behavior prediction [14]. For the junction-related maneuvers (TL, TR, GS, SS), the Time to Intersection TTI, and longitudinal velocity v_{lon}, v_{lat} of the target vehicle were chosen [12].

$P_{s^i}^i$ in Equation (1) is a vector to represent the a priori probabilities for the candidate models. These values are weights to balance the possibility of the candidate behaviors in the given scenario:

$$P_{s^i}^i = p(M_j), M_j \in M_{s^i}^i, \tag{5}$$

where the summary of these weights are equal to one:

$$\sum p(M_j) = 1, M_j \in M_{s^i}^i. \tag{6}$$

$P_{s^i}^i$ is defined by drivers' experiences or learned from training data. $F_{s^i}^i$, $M_{s^i}^i$, $P_{s^i}^i$ are determined by the current scenario s^i of the $i-$th vehicle.

2.1.2. Overview of the Proposed Approach

Figure 1 presents the architecture of the proposed driving behavior prediction approach, which can be divided into two phases: off-line phase and on-line phase. During the off-line phase, the road elements information was extracted from digital maps and described by the proposed ontology model. In order to learn the continuous features of driving behaviors, field data from typical urban traffic scenarios are collected first. Each type of driving behavior was learned by one HMM, i.e., a set of behavior models $\mathfrak{M}$ can be trained. Semantic rules are edited based on a priori knowledge (traffic rules, driving experiences) to define the candidate behavior models, input features, and prior probabilities for candidate models using scenario-dependent characteristics. All these rules were stored in the knowledge base. During the on-line prediction phase, all the learned models and semantic rules in the knowledge base were loaded into the working memory during the initialization process. At each time step, the scenario is represented based on a pre-defined ontology model and sensing data. $F_{s^i}^i$, $M_{s^i}^i$, $P_{s^i}^i$ are specified by the rule-based reasoning module. Then, each candidate learned model is tested to obtain the likelihood of each corresponding driving behavior $p(M_j)$. Finally, the behavior tag is decided by the behavior tagging model $\mathcal{M}(\cdot)$, which considers both model testing results (a posteriori probabilities) and a priori probabilities.

From the standpoint of the reasoning mode, the proposed driving behavior prediction system can also be divided into qualitative reasoning modules (enclosed by dotted boxes in Figure 1) and quantitative modules (enclosed by solid boxes in Figure 1). The qualitative modules act as a logical reasoner to infer the current scenario and generate $\{F_{s^i}^i, M_{s^i}^i, P_{s^i}^i\}$. Quantitative modules mainly model or test the continuous features of different driving behaviors.
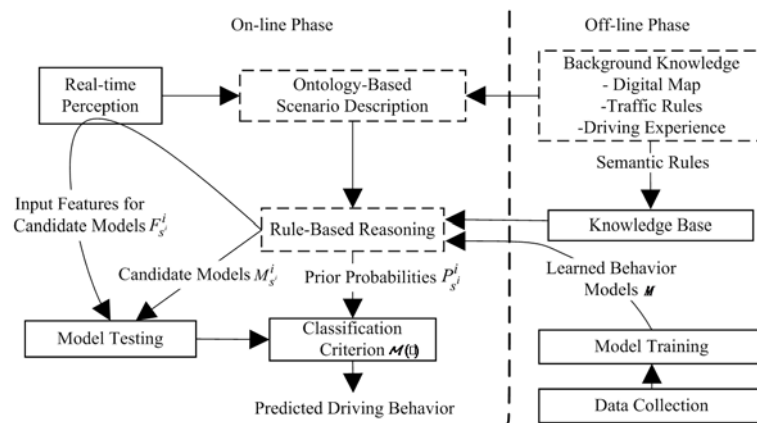
Figure 1. Overview of the proposed driving behavior prediction approach.

2.2. Ontology-Based Scenario Description and Reasoning

2.2.1. A Novel Ontological Model for Scenario Description

The traffic scenario can be represented by various methodologies, e.g., an occupancy grid map [7], Bayesian Network [8,11,13], etc. The representation methods that just consider the entities' spatial locations (like an occupancy grid map) cannot describe the relationships between different entities. Although a Bayesian Network can depict the inter-relations by carefully designed networks, the network usually will become pretty complex when considering a large number of entities. Moreover, it is hard to express the semantic relationships and integrate traffic rules. With the ability to define hierarchical classes and relationships between these classes, ontologies have been used to represent the contextual entities and expert knowledge unambiguously in knowledge-intensive systems [21,22]. As an interoperable and understandable expression form, ontology was defined as a formal and explicit specification of a shared conceptualization [23]. For different applications, ontology models should be defined based on corresponding domain knowledge. For autonomous driving, this technology has already been applied in some simple traffic scenarios [21,24–27]. However, there is still no standard ontology for autonomous driving on urban roadways so far.

In this study, we chose Description Logic (DL) as a formalism to represent the ontology knowledge base and store it with Web Ontology Language 2 (OWL 2) [28]. The DL knowledge base is composed of a terminological box (TBOX) and assertional box (ABOX). TBOX defines concepts (also known as classes) to describe the terminology in the application domain, e.g., *Vehicle*, *Sidewalk*. Each concept is defined by its data property axioms (also named attributes), e.g., *distanceToVehicle* is equal to 100 m. Relationships between these classes are described by object property axioms, e.g., *Vehicle isCloseTo Sidewalk*. Usually, the defined classes can be structured into a taxonomic topology. In order to describe the hierarchical relations, TBOX defined class expression axioms. These axioms were described by a set of operators such as equivalence operator $\equiv$, inclusion operator $\sqsubseteq$, intersection $\sqcap$, union $\sqcup$, complement $\neg$, etc. Complex descriptions are constructed based on atomic (i.e., basic) concepts (A, B) and atomic roles R. For example, the union of the basic classes or the association between concepts and roles can lead to more complex descriptions, e.g.,

$$C \equiv A \sqcup B \quad \text{or} \quad D \equiv \exists R.C. \tag{7}$$

The above classes and axioms are usually defined by domain experts based on their domain knowledge. ABOX defines the instances of TBOX, which are also named individuals $(a, b, ...)$. In addition, concept assertions $B(a)$ and role assertions $R(a, b)$ were defined to characterize these individuals' properties and relations, e.g., *Vehicle* (vehicle001), *hasLane* (segment001, lane012).

In order to represent the complex traffic scenario characteristics and intricate relationships between entities, this study proposed a novel ontology model to explicitly describe the traffic scenarios. The ontology model was edited by Protégé [29] software and parts of the ontology terminology are graphically represented in Figure 2. The classes (including *Entity* branch, *Attributes* branch) are enclosed by rectangles. Their semantic relationships are connected by the hierarchical object properties (dotted lines of different colors). The taxonomy relationships between concepts are plotted by light blue solid lines. The proposed ontology aims to cover all the possible scenarios in the urban environment. The *Entity* consists of four sub-entities:

1. *MapEntity.* The map entity is the most important part to describe the geometric topology of the roadway and road entities. The *MapEntity* class has two sub-classes: *AreaEntity* and *PointEntity*. The former entity represents one area of the road that cannot be simplified by a location point, such as *RoadPart*, *Sidewalk*, *Junction*, *Segment*, and *Lane*. *PointEntity* indicates the remaining road elements that can be considered a location point, e.g., *TrafficSign*, *LaneMarker*, *StopLine*. The detailed definition of these road entities and the decomposition are shown in Figure 3 (in the figure just one part of the entities are shown with the corresponding identifying number).

2. *ObstacleEntity.* This entity indicates the static and moving obstacles around the target vehicle. *StaticObstacle*, *Pedestrian*, *Vehicle*, *Animal*, *DynamicObstacle* are included in this entity:

$$ObstacleEntity \equiv StaticObstacle \sqcup Pedestrian \sqcup Vehicle \sqcup DynamicObstacle. \tag{8}$$

3. *EgoVehicle.* The ego vehicle entity was defined to express related attributes for the ego vehicle, including *VehicleType*, *EquippedSensors* (e.g., radar, cameral, etc.) and so on.

4. *Environment.* This entity considers environmental conditions such as *Weather*, *LightingConditions*, and *RoadSurfaceConditions*. These factors can significantly affect driving behaviors [30]. The variables in this entity usually cannot be easily detected by equipped sensors on the car, but can be acquired from on-line forecast systems (like a weather forecast, etc.).
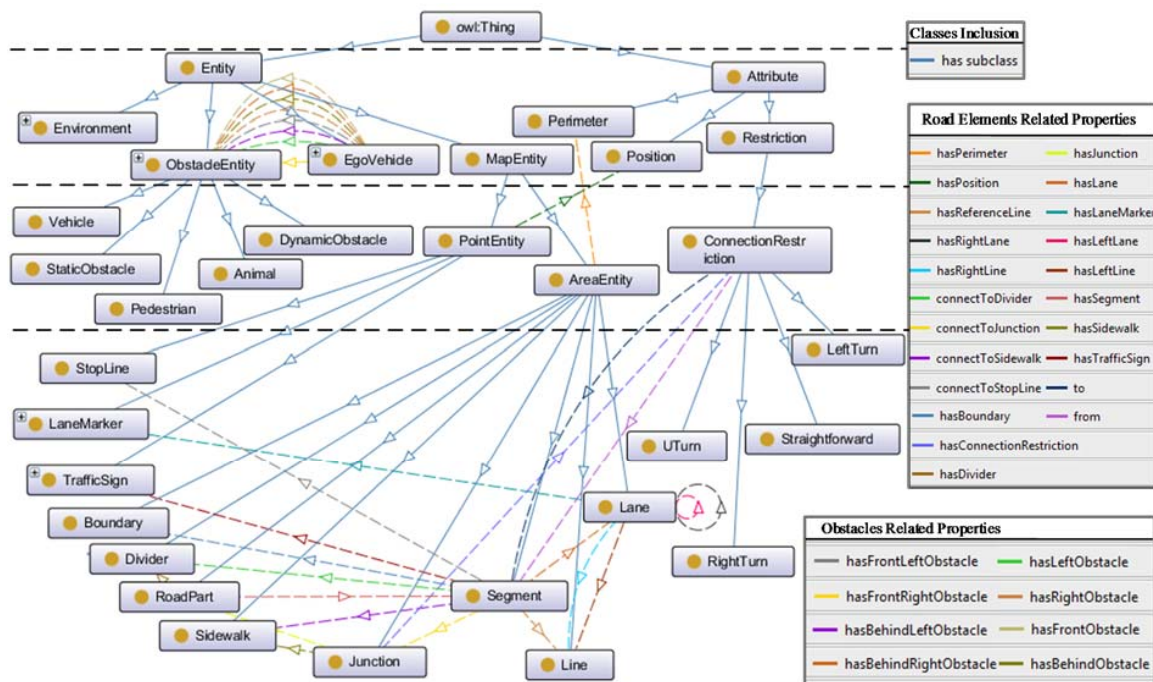

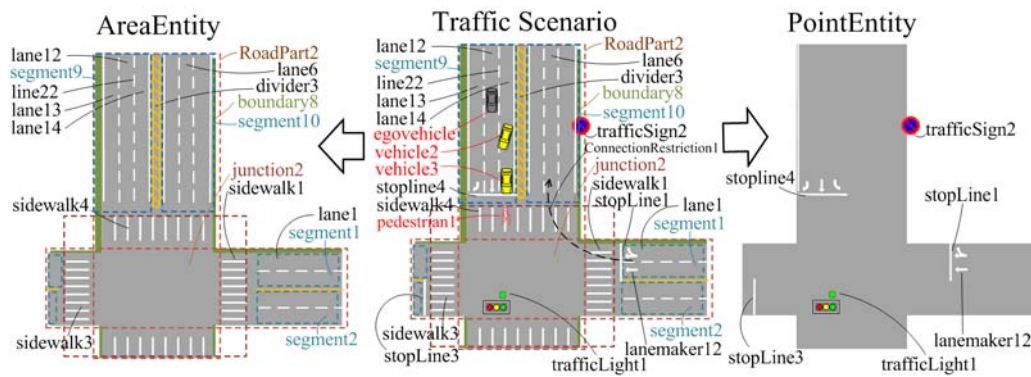**Figure 2.** Layout of the ontology model to describe the traffic scenario.

Figure 3. Schematic figure for road entity decomposition.

Attributes were defined to represent locations, area range, and restriction types. Three types of attributes were defined, i.e., *Position*, *Perimeter*, *ConnectionRestriction*. *Position* and *Perimeter* represent the GPS location and the scope for *PointEntity* and *AreaEntity* correspondingly. The *ConnectionRestriction* represents the connection directions between different segments.

Road elements related object properties were defined to express the semantic relationships between road elements pairs, such as segment to segment, segment to junction, junction to traffic light, etc. All these relationships are represented in Figure 2 and plotted by dash lines. The semantic relationships between different concepts were described by operators, for example,

$$
\begin{cases}
Sidewalk & \sqsubseteq AreaEntity \sqcap \\
& \exists hasSidewalk.Junction \\
TrafficSign & \sqsubseteq PointEntity \sqcap \\
& \exists hasTrafficSign.Segment
\end{cases}
. \tag{9}
$$

The object properties related to the obstacles are also shown in Figure 2. This study discretizes the direction from the view of the target vehicle (as Figure 4 shows). In this way, their relative position can be described qualitatively by *hasFrontLeftObstacle*, *hasFrontObstacle*, and so on [31].

Data properties are defined to characterize the detailed value of the classes' attributes, such as the distance to the StopLine (*distanceToStopLine*), type of lane (*isLeftMost*), and value of position (*valueOfPosition*). The data type of this property can be *double*, *boolean*, or *string*. In this study, the distance and velocity are discretized (as Figure 4 shows) to achieve a qualitative description.
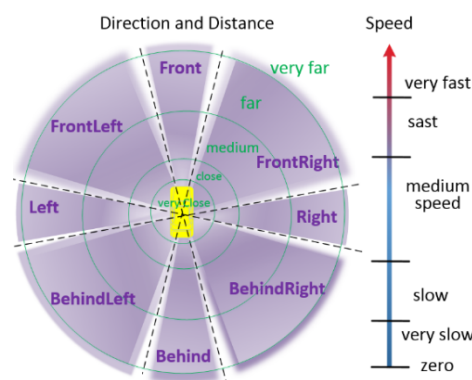

Figure 4. Qualitative mapping.

Based on the above definitions, for a given traffic scenario, each entity can be contextualized as one individual. The map-related instances were extracted from the high-resolution digital map [32]

and assigned an identification number (as Figure 3 shows). Instances of traffic participants were built based on perception data. Figure 5 shows some example instances for the traffic scenario in Figure 3. The data attributes for each individual were assigned, e.g., the velocity of *vehicle2* is 20 km/h, and the location of *junction2* is expressed as a rectangle with four GPS coordinates. Relationships between different road entities were determined by their spatial relationships, e.g., sidewalk4 and junction2 have the relationship *hasSidewalk*. Interactions between vehicles can be defined according to pre-defined qualitative relationships, e.g., *egovehicle hasBehandRightObstacle* with *vehicle2*. Also, relationships between traffic participants and road entities can also be described, e.g., the *currentlane* of *egovehicle* is *lane12*.
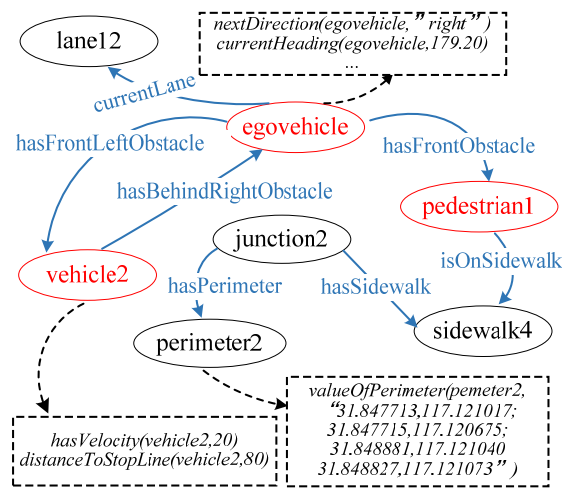

Figure 5. Example individuals for traffic scenario.

2.2.2. Knowledge Expression and Rule-Based Reasoning

Based on the defined ontology model, the Semantic Web Rule Language (SWRL) rules were designed to assign F_{si}^i, M_{si}^i, P_{si}^i based on the scenario characteristics. The SWRL rules consist of antecedent and consequent. The candidate driving behavior models, input features of the models, and a priori probabilities were encapsulated into the consequent. Traffic rules and drivers' experiences were applied to edit the antecedent, which was determined by the traffic scenario characteristics. The interactions between the target vehicle and its surrounding road entities and vehicles were considered with the regulation of traffic rules and driver experiences. The example rules are presented in Table 2.

Rule #1 defined the candidate models when the target vehicle tries to cross the intersection (this rule is applied for *vehicle3* in Figure 3). When the target vehicle (*vehicle3* in Figure 3) approaches the stop line (*stopline4* in Figure 3), the traffic light (*trafficLight1* in Figure 3) turns green and there is a pedestrian (*pedestrian1* in Figure 3) crossing the sidewalk (*sidewalk4* in Figure 3). In this case, in order to yield to the pedestrian, it can be derived that the target vehicle will be more likely to stop or execute turning behaviors. Rules #2 and #3 will be explained in Section 3.2.3. In this study, 58 SWRL rules were edited.

Although the SWRL rules have the advantage of being easy to understand, reasoning on a large amount of SWRL rules is computationally expensive [21]. In this study, considering the inconsistencies of format [33], the pre-defined ontology and SWRL rules were re-formatted to Prolog, which is a highly effective programming language for symbolic, non-numeric computation [34]. The Prolog language has good logical knowledge representation and reasoning. Furthermore, rich predicates were provided by Prolog, which makes the knowledge storage and reasoning much more convenient. In this study, SWI-Prolog (an open source implementation of Prolog) was applied for its versatile implementation and rich interfaces with other programing languages [35].

Table 2. Example rules that have been used in this study.

ID	SWRL Rules
Rule #1	*vehicle*(?V) ∧ *distanceToJunction*(?J, *close*) ∧ *hasTrafficLight*(?J, ?TL) ∧ *hasState*(?TL, *green*) ∧ *onRoadSegment*(?V, ?Seg) ∧ *connectToSidewalk*(?Seg, ?SW) ∧ *hasObstacle*(?SW, ?P) ∧ *hasSpeed*(?P, *low*) ∧ *hasDirection*(?P, ?Vertical)
	→*result*(?V," M{LK, TR, TL, SS}, F{TTI, v_{lon}, v_{lat}}, P{0.10, 0.30, 0.30, 0.30}")
Rule #2	*vehicle*(?V) ∧ *distanceToJunction*(?J, *far*) ∧ *hasSpeed*(?V, *medium*) ∧ *hasTargetFront*(?V, ?V_F)∧ *hasSpeed*(?V_F, *veryslow*) ∧ *hasRightFront*(?V, ?V_{RF}) ∧ *hasSpeed*(?V_{RF}, *fast*) ∧ *hasRightRear*(?V, ?V_{RR}) ∧ *hasSpeed*(?V_{RR}, *slow*) ∧ *distanceToTarget*(?V_{RR}, *close*) ∧*relativeSpeed*(?V_{RR}, *fast*)
	→*result*(?V, " M{LK,LCL,LCR}, F{$d_{diff-lane}$, v_{lat}}, P{0.10, 0.45, 0.45}")
Rule #3	*vehicle*(?V) ∧ *distanceToJunction*(?J, *close*) ∧ *hasSpeed*(?V, *fast*) ∧ *hasTargetFront*(?V, ?VF) ∧ *onRoadSegment*(?V, ?Seg1) ∧ *connectToJunction*(?Seg1, ?J) ∧ *connectToSidewalk*(?Seg1, ?SW) ∧ *hasObstacle*(?SW, *null*) ∧ *hasConnectionRestriction*(?J, ?CR) ∧ *IndividualOf*(?CR, ?X) ∧ *differentFrom*(?X,*Left*) ∧ *differentFrom*(?X,*Right*) ∧ *differentFrom*(?X,*UTurn*) ∧ *From*(?CR, ?Seg1) ∧ *To* (?CR, ?Seg2) ∧ *hasSeed*(?VF, *veryslow*)
	→*result*(?V, " M{LK, LCL, LCR }, F{TTI, v_{lon}, $d_{diff-lane}$, v_{lat}}, P{0.40, 0.10, 0.50}")
...	

2.3. Learning Quantitative Features

The continuous features of each type of behavior were learned by HMMs. In this section, the foundations of HMM are presented and the method of learning the features is provided.

HMMs have been used extensively for sequential data processing in a lot of fields, such as speech recognition, speech tagging [36], and so on. For the isolated word recognizer, each word was modeled by a distinct HMM, based firstly on the characteristics of this word. Once each word in the vocabulary is learned, for any given unknown word that needed to be recognized, the likelihoods for all the possible models can be calculated. The corresponding word for the model that gains the maximum likelihood will be considered as the result for the unknown word. As with word recognition, HMMs are pretty suitable for modeling driving behavior.

As a two-layer stochastic process, the HMM consists of a set of hidden states (*Q*), observations (*O*), transition probabilities (*A*), emission probabilities (*E*), and initial state probabilities (Π). They can be represented by the following equations:

$$Q = \{q_i\}, i = 1, 2, .., N \tag{10}$$

$$O = \{o_k\}, k = 1, 2, .., O_w \tag{11}$$

$$A = \{a_{ij} = P(q_j^{t+1}|q_i^t)\} \tag{12}$$

$$E = \{e_{ik} = P(o_k|q_i)\} \tag{13}$$

$$\Pi = \left\{\pi_i = P(q_i^1)\right\}. \tag{14}$$

For a given time *t*, the system being modeled is in one of the hidden states q_i^t, so is not observable. However, this state produces a set of observations with a certain probability (*E*). The transition probability from the current state to other states at the following time step is represented by *A*. The initial probability for each state is described by Π, where $\pi_i = P(q_1 = s_i)$. Then, an HMM can be determined by the triple $\lambda(A, E, \Pi)$.

Because the selected features of driving behavior are continuous values, the Gaussian Mixture Model (GMM) was applied to model the continuous probability density function (pdf) by

$$e_i(\mathrm{O}) = \sum_{m=1}^{M} c_{jm} \mathcal{N}\big(O\big|(\mu_{jm}, \sigma_{jm})\big), \ 1 \le j \le N, \tag{15}$$

where c_{jm} is the mixture coefficient for the $k - $ th mixture in $j - $ th state, and $\mathcal{N}$ is the pdf of Gaussian distribution with mean μ_{jm} and covariance matrix σ_{jm}. The mixture gains satisfy the following constraint:

$$\sum_{m=1}^{M} c_{jm} = 1, \ 1 \le j \le N, \tag{16}$$

where $c_{jm} \ge 0$, $1 \le m \le C$. C is the total number of mixtures. Then the pdf can be normalized by

$$\int_{-\infty}^{\infty} e_i(x) \, dx = 1, \ 1 \le j \le N. \tag{17}$$

Based on the aforementioned formulation, the HMM can be determined by $\lambda(A, E, \Pi, C, \mu, \sigma)$.

In this study, field data were collected for different scenarios. The observations for corresponding behaviors were extracted and used to train the HMMs. For each behavior, the K-means algorithm was applied to clustering the training data and estimate the initial distribution of the Gaussian functions. Next, the Baum–Welch algorithm [36] was applied to estimate the maximum likelihood HMM parameters, i.e.,

$$\lambda^*(A, E, \Pi, C, \mu, \sigma) = \underset{\lambda}{\mathrm{argmax}} \ P(\mathrm{O}|\lambda(A, E, \Pi, C, \mu, \sigma)). \tag{18}$$

Then, the HMM $\lambda^*(A, E, \Pi, C, \mu, \sigma)$ was considered as the driving behavior model for this type of behavior.

2.4. Driving Behavior Prediction

This section introduces the approach to predict the behaviors for the target vehicle (as Figure 6 shows). Firstly, a set of HMMs (λ_{LCR}, λ_{TR}, λ_{LK}, $\cdots$) corresponding to different behaviors were learned off-line based on the method in the last section. All these models were loaded into the memory of the on-line procedure during the initialization process. When given sequential observations for the $i - $ th vehicle are measured, the rule-based reasoning module will generate the candidate models M_{si}^{i}, corresponding input features F_{si}^{i}, and a priori probabilities according to the scenario characteristics. Each of the selected models λ_i will be tested to estimate how well ($P(\mathrm{O}|\lambda_i)$) the model fits the given observations, using the Forward algorithm [36].
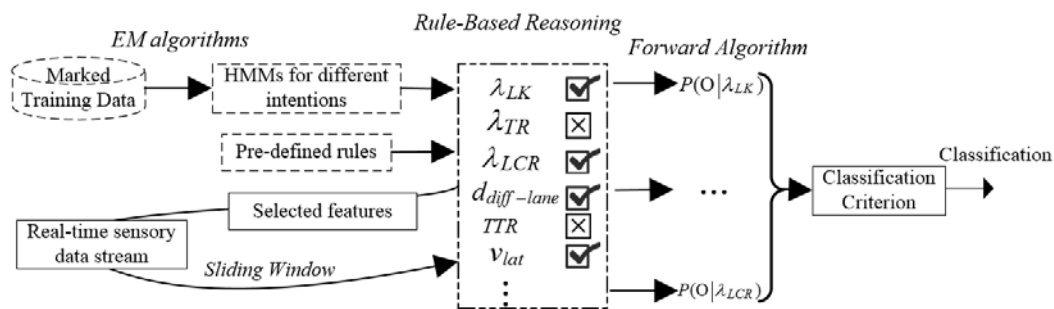

Figure 6. A flow diagram of on-line driving behavior prediction.

Unlike existing studies, which just use a posteriori probabilities to find the corresponding behavior for the model with the maximum likelihood

$$\underset{\lambda}{\mathrm{argmax}} \ P(\mathrm{O}|\lambda), \lambda \subset M_{si}^{i}, \tag{19}$$

in our proposed approach the a priori probabilities were combined with the a posteriori probabilities based on Bayes's rule, i.e.,

$$\overline{P}(O|\lambda) = P(O|\lambda)P_{s^i}^i. \tag{20}$$

The behavior corresponding to the model with the highest probability,

$$\underset{\lambda}{\arg\max}\ \overline{P}(O|\lambda), \tag{21}$$

will be considered as the behavior for the *i*th target vehicle.

3. Experiments and Discussions

3.1. Experimental Platform Construction

Figure 7 shows the experimental vehicle, "Intelligent Pioneer II", which was developed by our team and won the first prize in the "2010 Intelligent Vehicle Future Challenge of China" sponsored by the Natural Science Foundation of China [37]. Various sensors are equipped on the vehicle to enable its autonomous driving functions. In this study, just parts of the sensors were applied, i.e., the IBEO four-layer laser scanners, one Velodyne HDL-64E LIDAR, one high-resolution camera, and one differential GPS/INS (SPAN-CPT) system. The sensor configuration of the vehicle is shown in Figure 8. The differential GPS module provides the ego vehicle's location, speed, and heading. The moving obstacles were detected and tracked by the four-layer laser scanners (at the front of the vehicle), based on our previous work [25]. The location, velocity, size, and category of the moving vehicles (e.g., pedestrian, vehicles, etc.) can be measured based on their spatial–temporal relations with the experimental vehicle. All the surrounding situations were recorded by the camera. The perception range of the selected sensors is shown in Figure 8. We conducted the experiments in Hefei City (roadways are shown in Figure 9), which is a large city in China.

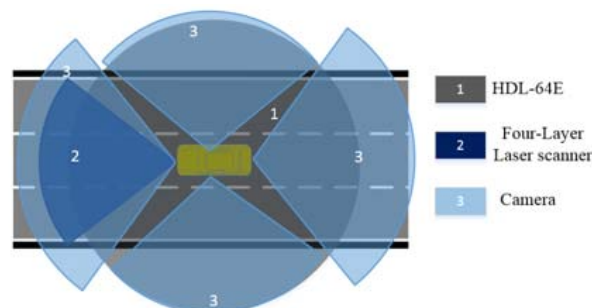
Figure 7. The Pioneer III autonomous vehicle.

Figure 8. Perception range of the autonomous vehicle.

Figure 9. Testing roadways with various layouts.

Before field data collection, the high-resolution maps were constructed. The map data were collected based on our experimental vehicle, and then post-processed and checked manually. Finally, the map, with more than 1600 road entities (e.g., stop signs, lane markers, lane lines, etc.) covered about 10 km of roadways (as Figure 9 shows). In the left part of Table 3, the elements of collected roadways are summarized.

Table 3. Extracted road elements and collected events.

Road Elements		Moving Targets		
Elements	Number	Behaviors	Events	Samples
Route parts	40	LK	412	107,272
Segments	231	LCL	198	60,701
Lane	312	LCR	289	83,289
Markers	336			
Sidewalk	150	Junction related	42	4300
Junctions	40			
Traffic Signs	486	Others	313	12,421
Total	1595	Total	1,254	267,983

3.2. Experiments and Results Analysis

3.2.1. Data Collection and Model Training

After the construction of the high-resolution map and the knowledge base, field data were collected on the test urban roadways over 15 days. The moving obstacles on the road were tracked at a frequency of 12.5 Hz. The data were recorded into the log files. In the end there were about 1000 events (one event is referred to as one-time tracking of a target vehicle), and 345 target vehicles were acquired. These events are summarized in the right-hand part of Table 3. The collected trajectories (in the experimental autonomous vehicle's coordinate system) for target vehicles are presented in Figure 10. The experimental autonomous vehicle has a 750×500 rectangular Cartesian coordinate system, in which the heading of the experimental vehicle is the positive direction of the vertical axis and the left direction is the positive direction of the horizontal axis. The size of each cell is 20 cm $\times$ 20 cm and the coordinates of the experimental vehicle are (250, 500). Because few events were detected at the junctions, all the events corresponding to junction-related maneuvers (i.e., TR, TL, and GS) are grouped together.

For each event, selected features were identified and extracted. In order to evaluate the performance of the behavior prediction models, the Reference Points (RPs) should be defined to represent the moments that the target vehicle executes maneuvers. For the lane-changing behavior, we

select the RPs as the moments when the bumper of the target vehicle touches the lane line. Also, any observations from less than 3.2 s before the RPs were considered in this study. Because it is difficult to define RPs for lane keeping events, all the LK samples were considered. Each sample, together with the previous O_w samples, was considered as one LK event.
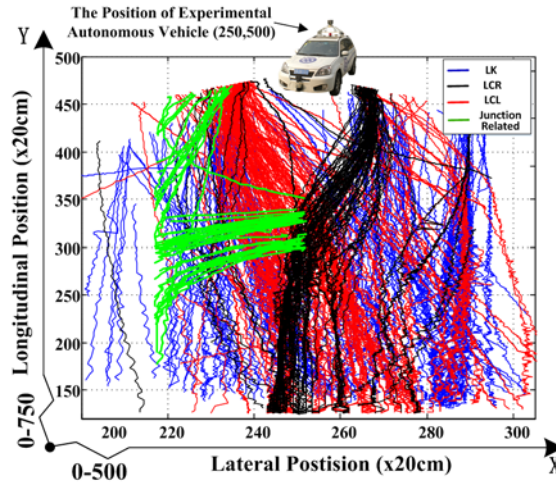

Figure 10. Collected trajectories in the experimental vehicle's coordinate system.

The parameters for the HMMs should be selected carefully, i.e., the observation window size O_w, the number of the hidden states N, and the number of the mixtures M. In this study, the Monte Carlo testing method was used to test a large set of these parameters' combinations. Each combination was generated with a number between 2 and 20 for N, a number between 5 and 40 for O_w, and a number between 2 and 10 for M. For the imbalance of the training data (most events belong to lane keeping), the best combination of HMMs were defined based on the $G - mean$ value, which was computed by

$$G - mean = \sqrt[3]{TPR_{LK} \cdot TPR_{LCR} \cdot TPR_{LCL}}, \tag{22}$$

where TPR and TNR are the true positive rate and the true negative rate, respectively. The parameter combination holding the highest $G - mean$ value was selected for the final HMM.

Based on the aforementioned steps, the parameters for the lane-changing behavior models are $N = 6$, $O_w = 10$, $M = 2$ with a $G - mean$ of 0.985.

3.2.2. Quantitative Evaluation

During the data collection phase, few turning events were detected at junctions. In order to get more representative results for the evaluation, the proposed approach evaluated non-junction related behaviors, i.e., LK, LCR, and LCL. The proposed approach (referred to in the following as Approach A) was compared with the state-of-art model [12,16] (Approach B), which is also based on HMMs but selects the most similar behavior whose model holds the maximum likelihood without consideration of background knowledge by Equation (19). Bearing in mind that most events belong to LK, evaluation measures for the imbalance data [38] were selected to evaluate the performance of Approaches A and B, i.e.,

$$prc = \frac{TPR}{TPR + FPR} \tag{23}$$

$$F_1 = \frac{2prc \cdot TPR}{prc + TPR} \tag{24}$$

$$G - mean = \sqrt{TPR \cdot TNR} \tag{25}$$

where TPR, TNR, and the false positive rate (FPR) are calculated by their definitions [38]. Because there are three behaviors for the non-junction road segments, this system belongs to a multi-class classification problem. The measures for each maneuver were calculated against the remaining classes.

The final results are shown in Figure 11. The results of Approach A are plotted by blue lines and the results of Approach B are plotted by black lines. Different maneuvers are distinguished by different markers. The results for 0.5 s, 1.0 s, and 1.5 s prediction time horizons are also extracted and presented in Table 4.
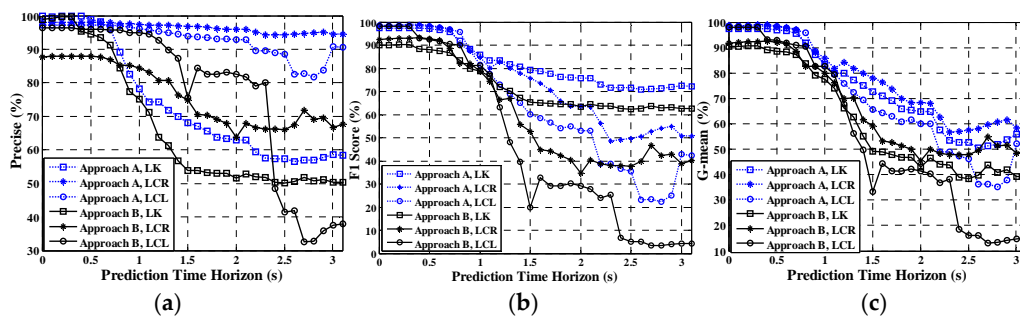

Figure 11. Performance evaluation results. (**a**) *prc*; (**b**) F_1; (**c**) $G - mean$.

In general, compared with Approach B, the precision for all the behaviors can be improved obviously by the proposed approach, especially for long-term prediction. Even when the prediction time horizon is 1.5 s, our approach still can maintain precision of 68% for the LK maneuver and around 95% for LCL or LCR maneuvers, which are improved by more than 26%. The significant improvement of F_1 score and $G - mean$ also verified the long-term prediction ability of the proposed approach, i.e., the prediction time horizon can be extended significantly. The developed system can predict the target vehicles' future driving behaviors before 2.1 s for LCR and 1.85 s for LCL, which is 56% longer than Approach B. This is mainly because of the combination of a priori probabilities, which implied multiple interactions with surrounding entities based on the interpreting of traffic scenarios. Based on the results, with consideration of a priori knowledge, the target vehicles' driving behaviors can be predicted more quickly and precisely.

As the prediction time horizon increases, the performance of Approach B will decrease sharply (Figure 11a). However, the proposed approach can keep a slower decreasing rate. After 1.5 s, all three measurements are below 90%. This is mainly caused by the stochastic behavior of the target vehicle. It is even harder for a human driver to predict surrounding vehicles' behaviors over such a long term. Most previous studies just try to improve the prediction performance of the lane-changing behavior (LCL, LCR), but the precision of LK was neglected [14,16]. In this study, we found that LK is pretty hard to predict for the urban roadways (the lowest precision in Figure 11). This may be caused by the highly dynamic nature of urban traffic. Compared with freeway traffic, the speed of the target vehicles is lower but more unexpected behaviors occur, like a sharp cut-in.

Table 4. Performance of the two approaches for time horizons of 0.5 s, 1.0 s, and 1.5 s.

Approach	Behaviors [1]	Measures									Average Prediction Time (s)
		0.5 s			1.0 s			1.5 s			
		prc	F1	G-Mean	prc	F1	G-Mean	prc	F1	G-Mean	
A	LK	0.99	0.97	0.97	0.78	0.86	0.84	0.68	0.79	0.73	–
	LCR	0.98	0.98	0.98	0.97	0.85	0.86	0.97	0.76	0.78	2.10
	LCL	0.97	0.98	0.96	0.96	0.82	0.83	0.94	0.60	0.66	1.85
B	LK	0.94	0.88	0.87	0.75	0.79	0.77	0.54	0.65	0.49	–
	LCR	0.88	0.93	0.92	0.84	0.79	0.80	0.75	0.53	0.59	1.34
	LCL	0.96	0.93	0.93	0.95	0.81	0.83	0.75	0.20	0.33	1.16

[1] LK: Lane Keeping; LCR: Lane Change to Right; LCL: Lane Change to Left.

3.2.3. Field Test

The ontology-based traffic scenario representation and rule-based reasoning module were compiled by SWI-Prolog and integrated with the experimental vehicle's main procedure. The learned HMMs for different driving behaviors were loaded into the memory during the initialization process.
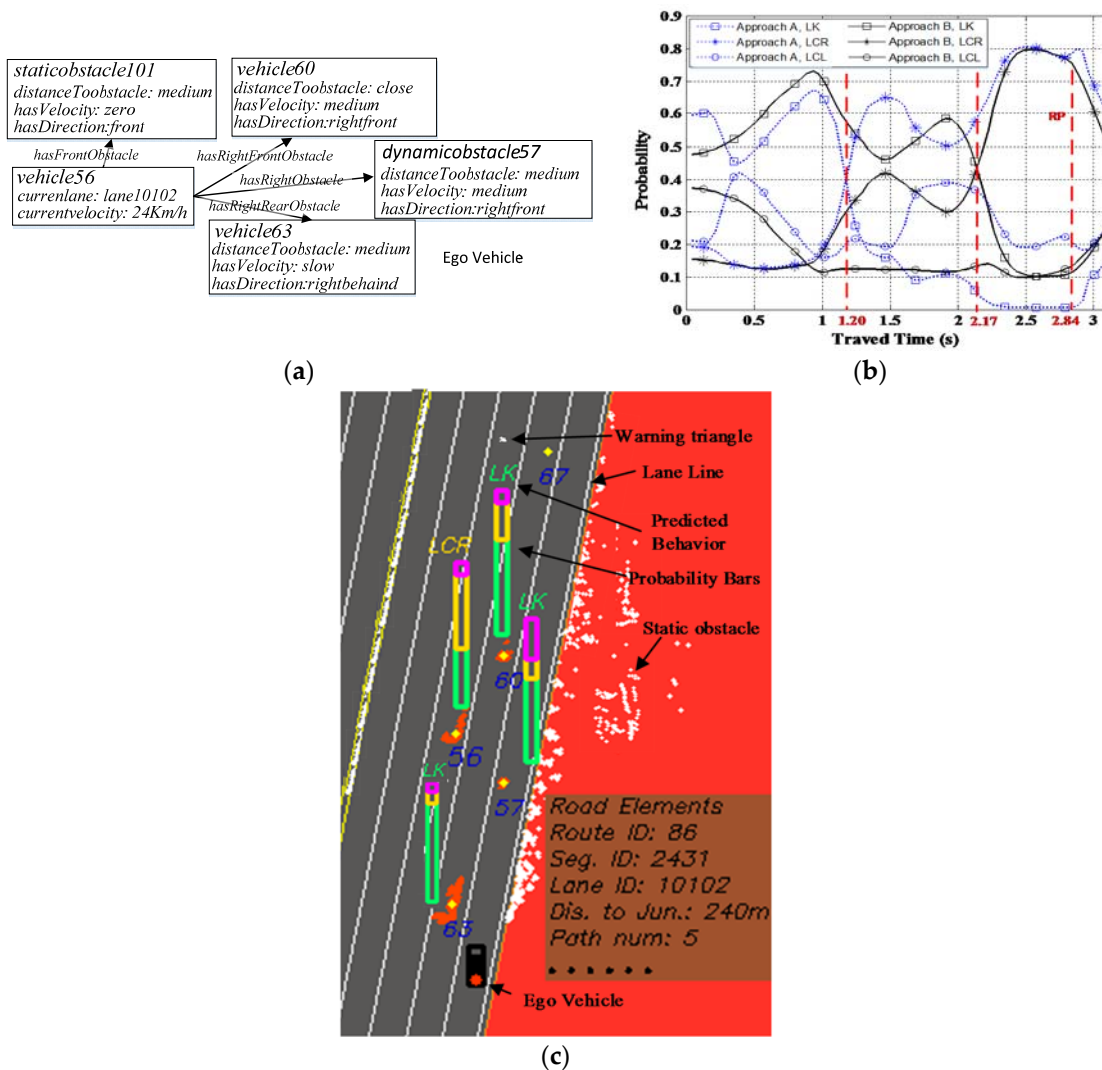
In this section, two representative experimental scenarios (as Figure 12 shows) are illustrated. The first scenario is at an arterial route, where there is a broken-down car parked in the third lane (counted from the right side of the route). A warning triangle is placed at the rear of the broken car. Some vehicles and motorcycles are rushing from the back of the ego vehicle. The white vehicle numbered 56 (*vehicle56*; the ID was assigned according to the target tracking algorithm) is approaching the warning triangle (*staticobstacle101*), while vehicle 60 (*vehicle60*) and vehicle 63 (*vehicle63*) are moving in the second lane. The driving behavior prediction algorithm can predict all the behaviors for these three vehicles and the motorcycle (*dynamicobstacle57*). In this paper, we just present the results for *vehicle56*, which is surrounded by two right-side moving vehicles and a static obstacle at the front. The second scenario is near a supermarket; there are three moving vehicles (*vehicle346*, *vehicle379*, and *vehicle409*), one junction (*junction214*) with a sidewalk (*sidewalk75*), and one pavement sign at the front of the ego vehicle. Some vehicles are parked in the rightmost lane. *vehicle379* is maintaining a high speed and the proceeding vehicle has a very low velocity. *vehicle409* is also keeping a high speed to travel through the sidewalk. This scenario can be considered an overlapped scenario, consisting of a junction scenario and a non-junction scenario. The results of driving behavior prediction for *vehicle379* are presented in the following paragraphs.

For Scenario 1, in fact, *vehicle56* changed lanes to the right before arriving at the warning triangle. Approach A and Approach B were applied to predict this lane-changing behavior. The results for Scenario 1 are shown in Figure 13. For Approach A, the relative positions of the moving obstacles and their semantic interrelationships (shown in Figure 13a) are analyzed first. Then, the rule-based reasoning module generated the candidate models and input features of the models according to the relationships between *vehicle56* and the surrounding objects (for *vehicle56*, Rule #2 in Table 2 is activated). The a priori probabilities were combined with the a posteriori probabilities by Equation (20) and the behavior tag is defined by Equation (21). Figure 13b shows the results of Approaches A and B. Approach A can predict the LCR intention about 1.64 s before the RP, while Approach B can predict the LCR intention 0.67s before the RP. This means Approach A can extend the prediction time horizon by 0.95 s. The scenario inference interface of the proposed approach is shown in Figure 13c. The non-drivable area is in red and static obstacles are represented by white dots. The moving obstacles are plotted in red color and their centers (yellow dots) are also marked. The behavior generated by the proposed approach for each target is represented by corresponding abbreviations and probability bars.

(**a**)　　　　　　　　　　　　　　　　　　　　　　(**b**)

Figure 12. Two experimental scenarios. (**a**) Scenario 1: a broken-down car stops at an arterial route; (**b**) Scenario 2: a car cuts in from the adjacent lane.
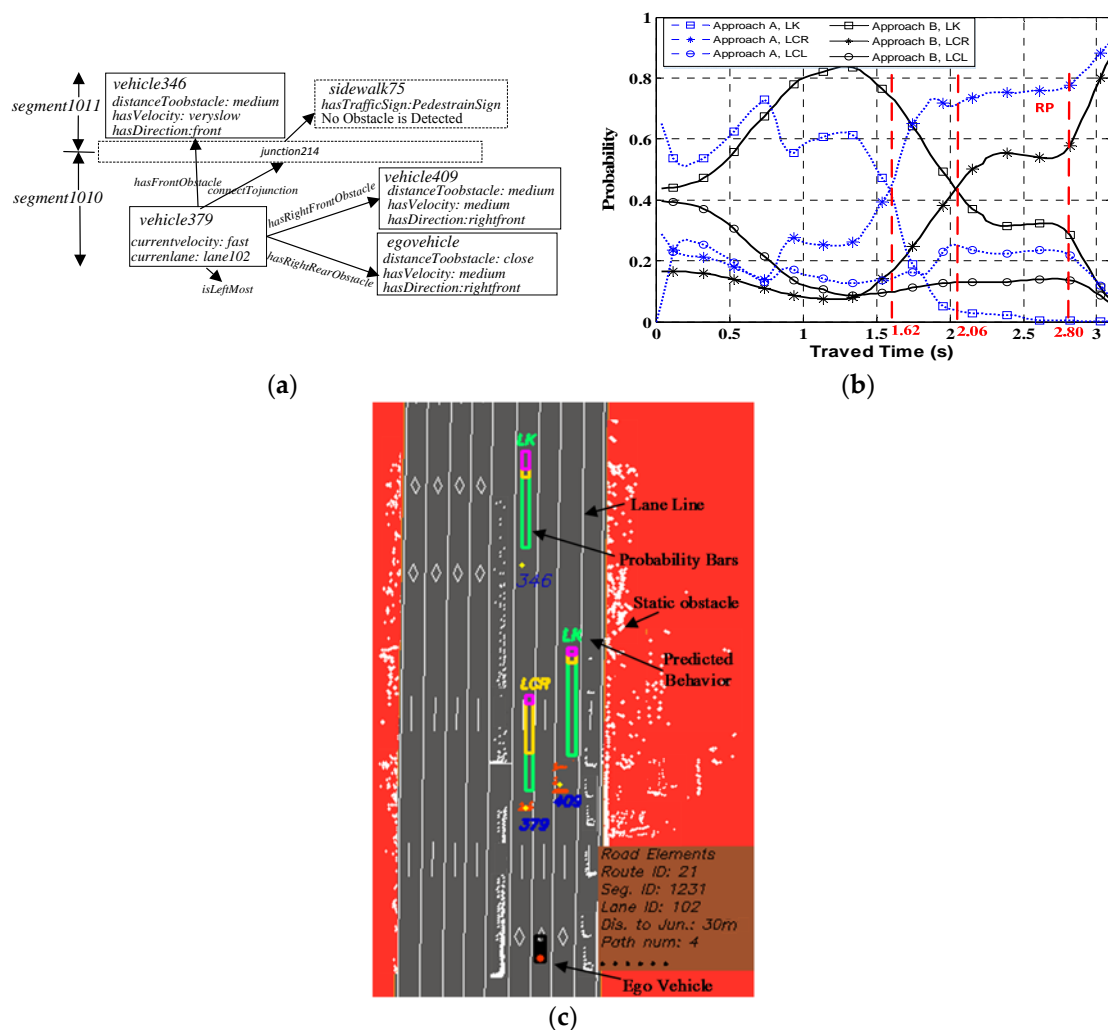
(**a**)

(**b**)

(**c**)

Figure 13. The results for Scenario 1. (**a**) Scenario reasoning; (**b**) Likelihood of different behaviors; (**c**) Driving behavior prediction interface.

The processed results for Scenario 2 are shown in Figure 14. Rule #3 in Table 2 is activated for this scenario. According to the scenario description and reasoning module, there are just two segments (*segment1010*, *segment1011*) connected to *junction214*. The restriction between *segment1010* and *segment1011* forbids the *vehicle379* to TR or TL. So the candidate models for this scenario are limited to LK, SS, LCL, and LCR. Because this is an uncontrolled intersection and there is no pedestrian crossing *sidewalk75*, it can be implied that *vehicle379* is not intent to stop before the stop line. The existence of the slow-moving vehicle *vehicle346* in front leads to *vehicle379* having a high probability to LCR. Finally, Approach A extends the prediction time horizon up to 0.44 s (Figure 14b). The predicted results are shown in Figure 14c. The prediction of future behavior of *vehicle379* can increase the autonomous vehicle's reaction time in order to avoid the crash caused by *vehicle379*'s cut-in.

For Scenarios 1 and 2, the candidate models can be selected according to the rule-based reasoning module. Without this reasoning module, it is hard to define the candidate models and input features of the models, especially for overlapped scenarios. The composite scenarios here refer to a scenario consisting of multiple basic scenarios. Like Scenario 2, it can be considered a combination of a junction-related scenario and a non-junction scenario, where both lane lines and a sidewalk can be found. Moreover, based on the background knowledge, the randomness of the target vehicle can be decreased by the restriction of road elements and the relationships with other vehicles. In addition,

due to the combination with drivers' experience, the lane-changing behavior can be detected before the variation of vehicle kinemics.

(a)

(b)

(c)

Figure 14. The results for Scenario 2. (**a**) Scenario reasoning; (**b**) Likelihood of different behaviors; (**c**) Driving behavior prediction interface.

Based on scenario reasoning and driving behavior prediction, the evolution of the current scenario can be predicted up to some seconds into the future. It should be emphasized that the proposed approach was designed to allow adaptiveness for all possible urban scenarios and is not limited to the two aforementioned scenarios.

4. Conclusions

In this study, a scenario-adaptive driving behavior prediction approach was proposed for autonomous driving on urban roadways. The scenario-dependent characteristics (road elements, traffic participants, and their inter-relationships) were represented by the proposed ontology model. The continuous features of each driving behavior were learned by HMMs with field data. Based on a priori knowledge (traffic rules, driving experience), a rule-based reasoning method is designed to specify candidate models, input features, and a priori probabilities for different scenarios. The future behavior of each target vehicle was predicted with consideration of both the a priori probabilities and the testing results of trained HMMs (a posteriori probabilities).

Experiments were conducted to evaluate the proposed approach. Based on the experimental results, the driving scenarios on the urban roadways can be described and interpreted by the proposed ontology model. For various urban traffic scenarios, the candidate models, input features, and a priori probabilities can be generated. The target vehicle's lane-changing behaviors can be predicted on average 1.85 s (for LCL) or 2.10 s (for LCR) in advance. Compared with the state-of-the-art driving behavior prediction model, the proposed approach can extend the prediction time horizon by up to 56% (0.76 s) on average, and the precision can be improved by 26% for long-term predictions. The proposed ontology model can improve an autonomous vehicle's scenario understanding ability greatly with the considering of semantic information. The evolution of the scenario can be predicted by the proposed approach, which can serve as the basis for safer decision-making algorithms for AVs.

Even though the proposed approach has advantages for driving behavior prediction, there are still some limitations, such as the time-consuming and tedious work to edit the SWRL rules. In future work, we will apply more advanced methods to generate these rules, such as learning rules from the training data. Furthermore, more field data will be collected and the proposed framework will be verified for more scenarios (e.g., highway entrance, T-junctions, uncontrolled intersections, etc.) with more driving behaviors.

Author Contributions: Xinli Geng and Huawei Liang conceived and designed the experiments; Xinli Geng and Biao Yu performed the experiments; Xinli Geng, Liuwei He and Rulin Huang analyzed the data; Xinli Geng and Pan Zhao contributed reagents/materials/analysis tools; and Xinli Geng wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Katrakazas, C.; Quddus, M.; Chen, W.H.; Deka, L. Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions. *Transp. Res. C Emerg. Technol.* **2015**, *60*, 416–442. [CrossRef]

2. Luettel, T.; Himmelsbach, M.; Wuensche, H.J. Autonomous ground vehicles: Concepts and a path to the future. *Proc. IEEE* **2012**, *100*, 1831–1839. [CrossRef]

3. Buehler, M.; Iagnemma, K.; Singh, S. *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*, 1st ed.; Springer: Canbridge, MA, USA, 2009; pp. 1–59.

4. Google's Self-Driving Car Company is Finally Here. Available online: https://www.wired.com/2016/12/google-self-driving-car-waymo/ (accessed on 29 December 2016).

5. All Tesla Cars Being Produced Now Have Full Self-Driving Hardware. Available online: https://www.tesla.com/blog/all-tesla-cars-being-produced-now-have-full-self-driving-hardware (accessed on 29 December 2016).

6. Paden, B.; Michal, C.; Yong, S.Z.; Yershov, D.; Frazzoli, E. A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles. *arXiv*, 2016; arXiv:1604.07446.

7. Liu, W. Planning under Uncertainties for Autonomous Driving on Urban Road. Ph.D. Thesis, National University of Singapore, Singapore, July 2016.

8. Gindele, T.; Brechtel, S.; Dillmann, R. Learning driver behavior models from traffic observations for decision making and planning. *IEEE Intell. Transp. Syst. Mag.* **2015**, *7*, 69–79. [CrossRef]

9. Lefèvre, S.; Dizan, V.; Christian, L. A survey on motion prediction and risk assessment for intelligent vehicles. *Robomech J.* **2014**, *1*, 1–14. [CrossRef]

10. Gnatzig, S.; Schuller, F.; Lienkamp, M. Human-machine interaction as key technology for driverless driving-A trajectory-based shared autonomy control approach. In Proceedings of the 21st IEEE International Symposium on Robot and Human Interactive (RO-MAN), Paris, France, 9–13 September 2012; pp. 913–918.

11. Gindele, T.; Brechtel, S.; Dillmann, R. A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments. In Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems (ITSC), Madeira island, Portugal, 19–22 September 2010; pp. 1625–1631.

12. Aoude, G.S.; Desaraju, V.R.; Stephens, L.H.; How, J.P. Driver behavior classification at intersections and validation on large naturalistic data set. *IEEE Trans. Intell. Transp. Syst.* **2012**, *13*, 724–736. [CrossRef]

13. Agamennoni, G.; Nieto, J.I.; Nebot, E.M. Estimation of multivehicle dynamics by considering contextual information. *IEEE Trans. Robot.* **2012**, *28*, 855–870. [CrossRef]

14. Bahram, M.; Hubmann, C.; Lawitzky, A.; Aeberhard, M.; Wollherr, D. A combined model-and learning-based framework for interaction-aware maneuver prediction. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1538–1550. [CrossRef]

15. Laugier, C.; Paromtchik, I.E.; Perrollaz, M.; Yong, M.; Yoder, J.D.; Tay, C.; Nègre, A. Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety. *IEEE Intell. Transp. Syst. Mag.* **2011**, *3*, 4–19. [CrossRef]

16. Lefevre, S.; Gao, Y.; Vasquez, D.; Tseng, H.E.; Bajcsy, R.; Borrelli, F. Lane keeping assistance with learning-based driver model and model predictive control. In Proceedings of the 12th International Symposium on Advanced Vehicle Control, Tykyo, Japan, 22–26 September 2014.

17. Gadepally, V.; Krishnamurthy, A.; Ozguner, U. A framework for estimating driver decisions near intersections. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 637–646. [CrossRef]

18. Gadepally, V.; Krishnamurthy, A.; Ozgüner, U. A Framework for Estimating Long Term Driver Behavior. *arXiv*, 2016; arXiv:1607.03189.

19. Bonnin, S.; Weisswange, T.H.; Kummert, F.; Schmueddrich, J. General behavior prediction by a combination of scenario-specific models. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 1478–1488. [CrossRef]

20. Kumar, P.; Perrollaz, M.; Lefevre, S.; Laugier, C. Learning-based approach for online lane change intention prediction. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV 2013), Gold Coast City, Australia, 23–26 June 2013; pp. 797–802.

21. Armand, A.; Filliat, D.; Ibañez-Guzman, J. Ontology-based context awareness for driving assistance systems. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Dearborn, MI, USA, 8–11 June 2014; pp. 227–233.

22. Tenorth, M.; Beetz, M. KnowRob: A knowledge processing infrastructure for cognition-enabled robots. *Int. J. Robot. Res.* **2013**, *32*, 566–590. [CrossRef]

23. Gruber, T.R. A translation approach to portable ontology specifications. *Knowl. Acquis.* **1993**, *5*, 199–220. [CrossRef]

24. Zhao, L.; Ichise, R.; Sasaki, Y.; Liu, Z.; Yoshikawa, T. Fast decision making using ontology-based knowledge base. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; pp. 173–178.

25. Huang, R.; Liang, H.; Chen, J.; Zhao, P.; Du, M. An intent inference based dynamic obstacle avoidance method for intelligent vehicle in structured environment. In Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO), Zhuhai, China, 6–9 December 2015; pp. 1465–1470.

26. Lattner, A.D.; Gehrke, J.D.; Timm, I.J.; Herzog, O. A knowledge-based approach to behavior decision in intelligent vehicles. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 6–8 June 2005; pp. 466–471.

27. Vacek, S.; Gindele, T.; Zollner, J.M.; Dillmann, R. Using case-based reasoning for autonomous vehicle guidance. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, 29 October–2 November 2007; pp. 4271–4276.

28. Bock, C.; Haase, P.; Hoekstra, R.; Horrocks, I.; Ruttenberg, A.; Sattler, U.; Smith, M. *OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax*, 2nd ed.; W3C Recommendation: Cambridge, MA, USA, 2012.

29. A Free, Open-Source Ontology Editor and Framework for Building Intelligent Systems. Available online: http://protege.stanford.edu/ (accessed on 29 December 2016).

30. Geng, X.; Liang, H.; Xu, H.; Yu, B. Influences of Leading-Vehicle Types and Environmental Conditions on Car-Following Behavior. *IFAC-PapersOnLine* **2016**, *49*, 151–156. [CrossRef]

31. Gehrke, J.; Lattner, A.; Herzog, O. Qualitative mapping of sensory data for intelligent vehicles. In Proceedings of the Workshop on Agents in Real-Time and Dynamic Environments at the 19th International Joint Conference on Artificial Intelligence (IJCAI), Edinburgh, UK, 30 July–5 August 2005; pp. 51–60.

32. Aeberhard, M.; Rauch, S.; Bahram, M.; Tanzmeister, G.; Thomas, J.; Pilat, Y.; Kaempchen, N. Experience, Results and Lessons Learned from Automated Driving on Germany's Highways. *IEEE Intell. Transp. Syst. Mag.* **2015**, *7*, 42–57. [CrossRef]

33. Samuel, K.; Obrst, L.; Stoutenberg, S.; Fox, K.; Franklin, P.; Johnson, A.; Peterson, J. Translating owl and semantic web rules into prolog: Moving toward description logic programs. *Theory Pract. Log. Program.* **2008**, *8*, 301–322. [CrossRef]

34. Ivan, B. *Prolog Programming for Artificial Intelligence*, 4th ed.; Pearson Education: North York, ON, Canada, 2011.

35. SWI Prolog: Robust, Mature, Free Prolog for the Real World. Available online: http://www.swi-prolog.org/ (accessed on 29 December 2016).

36. Rabiner, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **1989**, *77*, 257–286. [CrossRef]

37. Mei, T.; Liang, H.; Kong, B.; Yang, J.; Zhu, H.; Li, B.; Zhang, W. Development of 'Intelligent Pioneer' unmanned vehicle. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Madrid, Spain, 3–7 June 2012; pp. 938–943.

38. Sun, Y.; Kamel, M.S.; Wong, A.K.; Wang, Y. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recogn.* **2007**, *40*, 3358–3378. [CrossRef]