sensors

MDPI

Article

A Hybrid Approach for Turning Intention Prediction Based on Time Series Forecasting and Deep Learning

Hailun Zhang [1] and Rui Fu [1,2,*]

[1] School of Automobile, Chang'an University, Xi'an 710064, China; Iszhanghailun@outlook.com
[2] Key Lab of Vehicle Transportation Safety Technology, Ministry of Transport, Chang'an University, Xi'an 710064, China
* Correspondence: furui@chd.edu.cn; Tel.: +86-029-8233-4722

Received: 24 June 2020; Accepted: 26 August 2020; Published: 28 August 2020

Abstract: At an intersection with complex traffic flow, the early detection of the intention of drivers in surrounding vehicles can enable advanced driver assistance systems (ADAS) to warn the driver in advance or prompt its subsystems to assess the risk and intervene early. Although different drivers show various driving characteristics, the kinematic parameters of human-driven vehicles can be used as a predictor for predicting the driver's intention within a short time. In this paper, we propose a new hybrid approach for vehicle behavior recognition at intersections based on time series prediction and deep learning networks. First, the lateral position, longitudinal position, speed, and acceleration of the vehicle are predicted using the online autoregressive integrated moving average (ARIMA) algorithm. Next, a variant of the long short-term memory network, called the bidirectional long short-term memory (Bi-LSTM) network, is used to detect the vehicle's turning behavior using the predicted parameters, as well as the derived parameters, i.e., the lateral velocity, lateral acceleration, and heading angle. The validity of the proposed method is verified at real intersections using the public driving data of the next generation simulation (NGSIM) project. The results of the turning behavior detection show that the proposed hybrid approach exhibits significant improvement over a conventional algorithm; the average recognition rates are 94.2% and 93.5% at 2 s and 1 s, respectively, before initiating the turning maneuver.

Keywords: advanced driver assistance system; autonomous vehicle; driving intention prediction; online time series prediction; bidirectional long short-term memory network

1. Introduction

With the widespread implementation of advanced driver assistance systems (ADAS) and the rapid development of artificial intelligence, autonomous driving has become a reality [1–4]. This development means that in the future, a mixed environment will be inevitable. Human-driven vehicles, autonomous vehicles, as well as connected vehicles, will travel together on roads. Many scholars have begun to study the characteristics of safe driving of mixed traffic and its impact on drivers [5–11]. It is well-known that intersections represent bottlenecks in urban traffic, reducing traffic efficiency. Due to the complex characteristics of intersections, the accident rate at or near this location is relatively high [12,13]. Numerous collisions and fatal accidents occurred at intersections in the United States, where an estimated 45% of injury crashes and 22% of roadway fatalities are intersection-related [14]. According to the EU community road accident database, in the past decade (2001–2010), intersection-related fatalities accounted for more than 20% in the EU [15]. The inability of drivers to assess correctly and/or observe dangerous situations is believed to be a major factor in these accidents [16]. For example, in Figure 1, the traffic light is green, allowing vehicles on East Street and West Street to pass. The white vehicle on West Street will turn left, followed by a black autonomous vehicle behind it, ready to drive straight through the intersection. The black vehicle

needs to infer the turning behavior of the white vehicle to plan a safe driving path and minimize the acceleration or deceleration rate to improve passenger comfort and reduce emissions [17]. Therefore, the accurate and early detection of the driving intentions of surrounding vehicles is crucial to prevent traffic accidents and improve ride comfort for new-generation ADAS systems. The earlier the detection occurs, the earlier the system can intervene, and every millisecond is critical for safe driving.

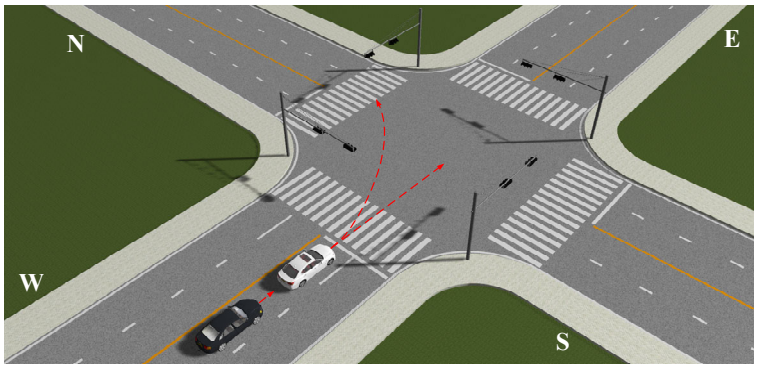


Figure 1. The scene of a vehicle entering the intersection.

In this paper, we propose a method for predicting and recognizing vehicle turning behavior at intersections using a combination of time series prediction and deep learning networks, which can predict the intention of the vehicle before the turning maneuver is initiated. In recent years, scholars conducted extensive research on driver behavior recognition or intention prediction [18–23]. Driving behavior is considered a continuous time-varying dynamic process [24,25], which is consistent with our general cognition. The problem of driving behavior recognition/inference/estimation is transformed into a problem of classification, recognition, or prediction of a time series. Many studies have used machine learning algorithms to analyze driving behavior, such as a continuous hidden Markov model (CHMM), Gaussian mixture model (GMM) [24–29], support vector machine (SVM) [30,31], back-propagation (BP) neural network [18,32], random forest, and Adaboost algorithm [33]. Specifically, in [24] and [27], an algorithm combining an HMM and Bayesian filtering (BF) was proposed to model the vehicle behavior while entering the intersection and performing a lane change. In [26], the author used the steering wheel angle and the steering wheel angle velocity as inputs for model training to develop the CHMM model. In [28], a CHMM and GMM were combined to model lane change and lane-keeping behaviors, respectively. The authors used public data and highway driving as a scenario to recognize and predict the lane change behavior of the target vehicle from the perspective of the host vehicle. In [25] and [29], a hybrid-state system (HSS) and HMM framework were integrated to model vehicle turning behavior at an intersection. In the HSS, the driver's decision was modeled as a discrete state system, and the vehicle dynamics were modeled as a continuous state system. The SVM is a popular algorithm for classification problems, but it cannot model a time series. In [30] and [31], the objective was to classify driving behavior. The author concatenated different meta features of different times in the window to create a feature vector of fixed length or used the means and variances of the data collected in the sliding time window to replace individual measurements. In reference [32], the author developed a BP neural network prediction model of driver lane change behavior using vehicle movement data, relative motion data of the vehicle of interest and surrounding vehicles, and head movement data of the drivers. Another study [33] used ensemble learning methods to model lane changing driving behavior for the first time. The results showed that both methods provided high classification accuracies and low false alarm rates.

In existing research on driving behavior modeling, the HMM+GMM algorithm is most commonly used because the HMM can model time series of any length and infer unobserved (hidden) states. The GMM can model continuous observations using multiple probability density functions. As a result of the recent success of deep learning in image classification, speech recognition, and other fields, many researchers have begun to use this method for driving behavior recognition [34–36]. A long short-term memory (LSTM) model was proposed in [37] to increase the long-term

dependency property and overcome the problem of gradient descent. LSTM is a variant of RNN that can capture and model long-term dependence in time series data. In [38–40], LSTM models were developed to infer the driving intention of vehicles approaching intersections. Since an LSTM can be trained in a sequence-to-sequence prediction manner, it can predict the trend of a period of time in the time series, enabling to predict the future activities of drivers [41].

The following conclusions can be drawn from the reviewed literature. First, regarding lane-changing behavior, what happens from the intent to the execution of the behavior is the result of the interaction between the driver and the surrounding environment. However, changes in the environment, such as pedestrians, front vehicle cut-ins, and motorcycle interference, are often less likely to modify the driver's turning intention. Generally, the behavior of surrounding vehicles will only affect the steering maneuver, and it is difficult to reverse the driver's turning intentions [42]. Intention prediction refers to recognizing the driving maneuver before the initiation of the actual maneuver, and behavior recognition refers to recognizing the driving maneuver in the early stage after the initiation of the maneuver. Since the behavior of the vehicle at the intersection is related to the driver's task level, it is difficult to infer indirectly the driver's intention through changes in the environment, such as the lane change behavior recognition that was proposed in [28]. When the driver's destination is unknown, it is difficult to predict the turn behavior, similar to navigation software. However, different drivers have different driving preferences and characteristics, and the effects of their actions on the kinematics of the vehicle often reflect the driver's intentions [15,43]. We can estimate the driver's future activity from motion information and trajectory information of the vehicle approaching the intersection for a certain period of time, or recognize the activity in time when the vehicle exhibits early turning behavior characteristics. Second, empirical analysis has shown that if ADAS or autonomous vehicles cannot recognize the driving behavior of surrounding vehicles, the system cannot plan a safe and comfortable route in real-time. Most of the above studies were conducted from the perspective of the host vehicle, and the results may not be applicable to the behavior prediction and recognition of surrounding vehicles because the eye movement and facial tracking data of the drivers of the surrounding vehicles cannot be obtained [44,45]. In addition, most research on driving behavior recognition at intersections recognizes turning behaviors in the early stages after the maneuver has started [24,25,29,31]. Whereas references [15,38,39], and [40] claim to achieve intent prediction, ADAS or autonomous vehicles cannot record extensive historical data of the target vehicle to achieve intent prediction. During actual driving, the prediction of the future situation of the surrounding vehicles is often more in line with the driver's preference, and the intention of the surrounding vehicles is inferred based on this [3]. However, in the literature, the prediction of car trajectories has mostly used offline algorithms, including the recurrent neural network (RNN), which is a deep learning network, LSTM [36,46], the interacting multiple model (IMM) [3], and the potential field method [47]. None of these methods meet the real-time requirements of the system, and the intuitive observation of the speed often has a significant effect on predicting the intention of the vehicle [42,48]. It is preferred to obtain the speed of the surrounding vehicles through on-board sensors. Furthermore, most studies that recognized or predicted driving behavior were essentially predicting time series or vector data based on variables that characterize the driving behavior, such as trajectory, speed, and acceleration. It should be noted that the vector data in previous studies is obtained using average or variance processing on the multi-dimensional variables in the time window.

As stated, the main problem to be solved in this article is the early prediction of the driver's intention before the actual turning maneuver begins. We regard the trajectory and kinematic characteristics of the vehicles as time series and consider the temporal context of these parameters. Based on the literature review and the concept of classification and prediction of time series [49,50], we propose a method for turning behavior prediction and recognition using online time series forecasting and deep learning classification. The proposed driver behavior prediction system comprises two layers, including offline behavior learning via a high-level behavior model and online behavior prediction via low-level vehicle state prediction. The schematic is shown in Figure 2. First, the driving behaviors at the intersection are learnt using a deep learning network. A bidirectional

long short-term memory (Bi-LSTM) model is developed to recognize the turning behavior using time series data of the motion parameters and derived parameters characterizing vehicle behavior and intention. Second, an online autoregressive integrated moving average (ARIMA) time series prediction algorithm is used to predict the variables that characterizes the turning behavior. Then, the predicted data are used by the Bi-LSTM to predict drivers' turning behavior.

To the best of the authors' knowledge, this work is the first attempt to integrate online time series forecasting and behavior recognition for driving behavior prediction. The proposed framework is validated using the open-source next-generation simulation (NGSIM) dataset, which is based on real-world driving experiments that were conducted by the Federal Highway Administration's (FHWA) and provided promising performance.

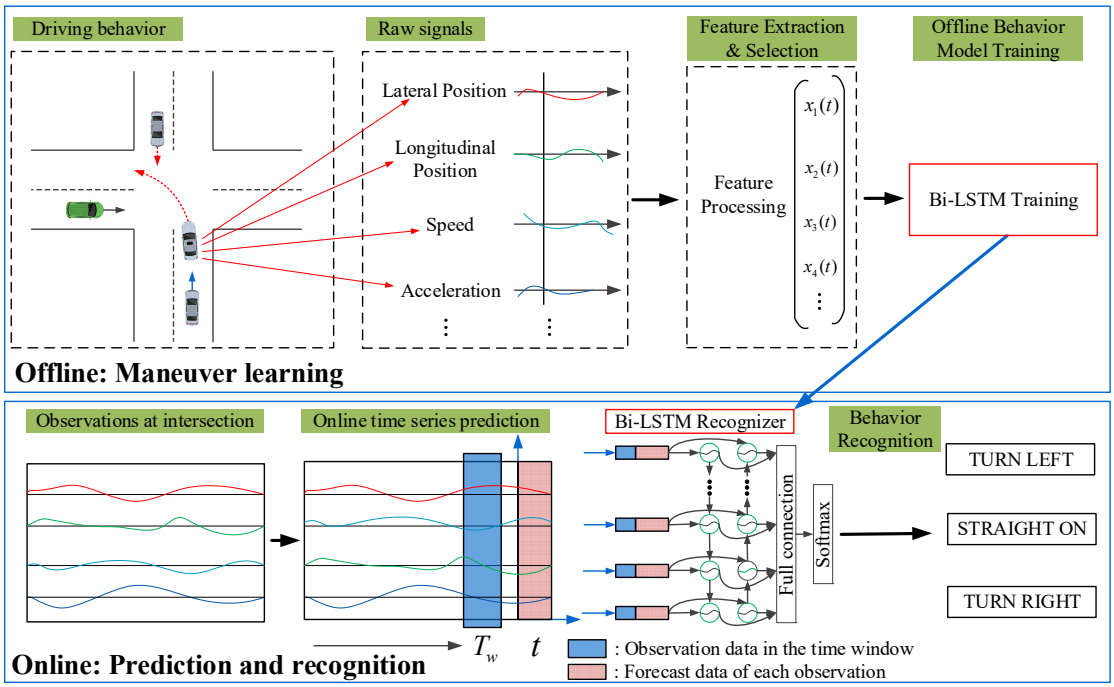


Figure 2. Schematic of intersection behavior prediction and recognition.

The main contributions of this study are as follows.

- The purpose of this research is to realize the intention prediction of the vehicle before the start of the steering maneuver, and propose a framework that combines time series prediction and deep learning methods to apply to the new generation of ADAS or future autonomous vehicles. The framework uses online prediction algorithms to reflect the driving intention of the vehicle in the prediction window, and achieves high recognition accuracy and modeling of vehicle kinematics that indicate steering behavior through Bi-LSTM.
- Since the variables representing the driving behavior are time series data, a novel vehicle behavior prediction method is proposed that combines the ARIMA with an online gradient descent (OGD) optimizer. This method allows for predicting the driving intention without reducing the recognition rate.

The remainder of this paper is organized as follows. Section 2 details the proposed online ARIMA and Bi-LSTM hybrid approach. The experimental data and the data processing and analysis methods are described in Section 3. The experimental results are described in Section 4. Finally, the conclusions are provided in Section 5, and potential future studies are outlined.

2. Framework for Turning Behavior Recognition

Survey results have shown that the dominant factor causing traffic accidents is human error. More than 90% of traffic accidents are partially caused by human error, and 57% of accidents are entirely caused by human error [48]. Perception of danger has always been a focus of ADAS development. The prediction of the driving intentions based on the movement of surrounding vehicles in real-time has been a challenge for ADAS systems.

In this section, we describe the methods used in the proposed framework. The system is divided into two parts. In the offline learning phase, a Bi-LSTM model is used to learn the extracted turning behavior. The online phase takes place in two steps. Multiple time series variables that characterize the turning behavior are predicted using the online ARIMA algorithm, and the predicted data and the derived parameters are used in conjunction with the offline training data of the turning behavior to recognize the turning behavior.

2.1. Bi-LSTM

We know from the literature that when vehicles approach an intersection, the historical information hidden in the vehicle's motion parameters often reflects the driver's turning intention, and previous information often has a considerable impact on the recognition result [15,40,43]. Therefore, it is preferred to use a model that considers the behavioral characteristics of the vehicle for an extended period before approaching the intersection to detect the turning behavior.

RNNs use historical information to assist in current decision-making. However, earlier signals have less important information than more recent signals and RNNs cannot solve the long-term dependence problems. An LSTM network with its unique gate structure not only solves the problem of long-term dependence but also addresses gradient vanishing or explosion that occur when RNNs are used to process time series and can be used to model arbitrary time series [41]. Each gate has its own weight matrix and time lag. A single LSTM unit is shown in Figure 3.

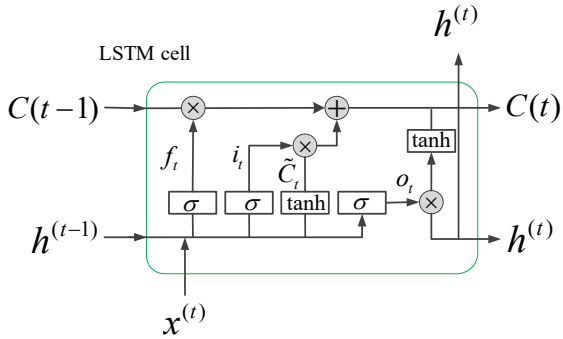


Figure 3. Internal structure of a long short-term memory (LSTM) cell.

A graphic illustration of a standard LSTM network is depicted in Figure 4a. The $\boldsymbol{i}$ is an input gate that controls the size of the new content added to the memory. The $\boldsymbol{f}$ is a forget gate that determines the amount of memory that should to be forgotten. The $\boldsymbol{o}$ is an output gate that adjusts the amount of output memory content. The $\boldsymbol{c}$ is the cell activation vector, which consist of the partially forgotten previous memory $\boldsymbol{c}_{t-1}$ and modulated the new memory $\tilde{\boldsymbol{c}}_t$. t represents the t-th moment [51].

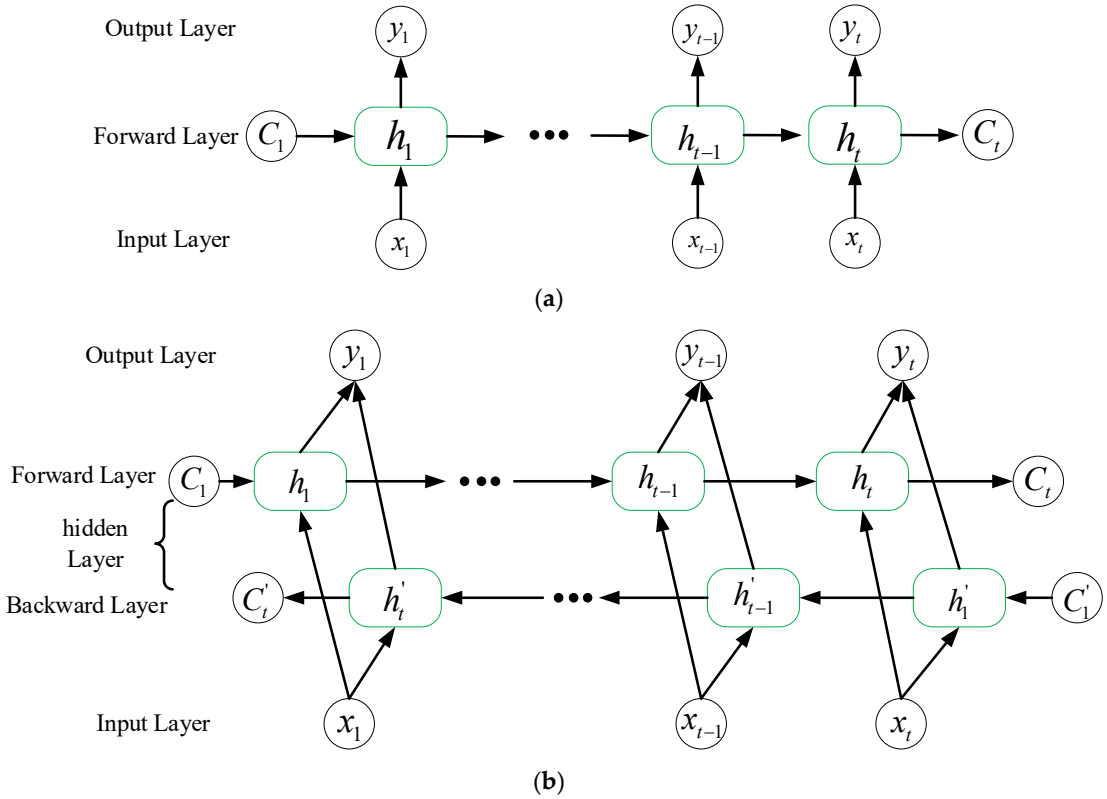


Figure 4. Illustration of a LSTM model (**a**) and a bidirectional (Bi-LSTM) model (**b**).

$$
\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \\
\tilde{c}_t &= tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
c_t &= f_x \otimes c_{t-1} + i_t \otimes \tilde{c}_t \\
h_t &= o_t \otimes \tanh(c_t)
\end{aligned}
\tag{1}
$$

where $\boldsymbol{i}_t$, $\boldsymbol{f}_t$, $\boldsymbol{o}_t$, and $\boldsymbol{c}_t$ represent the values of $\boldsymbol{i}$, $\boldsymbol{f}$, $\boldsymbol{o}$ and $\boldsymbol{c}$ at time t, respectively. $\boldsymbol{y}_t$ is the output of the output layer at time t. $\boldsymbol{x}_t$ is the current input vector. $\boldsymbol{h}_t$ is the hidden layer vector at time t, which contains all the outputs of the LSTM. The $\boldsymbol{W}$ represents the self-updating weight of the hidden layer, and $\boldsymbol{b}$ represents the bias vector. The σ is the sigmoid activation function, and tanh is the hyperbolic tangent function. The operator $\otimes$ is the dot multiplication operation. All gate values and hidden layer outputs are in the range (0, 1).

In [27], lane change intention recognition is considered a speech recognition problem. We know that when filling vacancies in a sentence, it is necessary to grasp the words before the vacancy; however, the words after the vacancy must also be considered to predict an outcome that is in line with the situation and accurate. The Bi-LSTM solves these types of problems. In the example of the sequence, the network connects the same output from two LSTM networks that is learned and modeled from both ends of the sequence. The forward LSTM obtains past data information of the input sequence, and the backward LSTM obtains future data information of the input sequence [52]. Therefore, the network processes information from both ends of the time series and not only considers the influence of historical data but also that of future states, thereby significantly increasing the generalization ability of the model. The model structure of the LSTM is shown in Figure 4b. The formula is expressed as follows:

$$\overrightarrow{h_t} = \overrightarrow{LSTM}(h_{t-1}, x_t, c_{t-1}), t \in [1, T]$$
$$\overleftarrow{h_t} = \overleftarrow{LSTM}(h_{t+1}, x_t, c_{t+1}), t \in [T, 1] \quad (2)$$
$$H_t = [\overrightarrow{h_t}, \overleftarrow{h_t}]$$

where $\boldsymbol{H}_t$ is the hidden state of the Bi-LSTM at time t, including two the LSTMs that operate in the opposite directions $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$.

2.2. Online ARIMA

As mentioned before, the vehicle's driving trajectory and motion feature parameters can be considered a dynamic process in a time series. On this basis, the ARIMA algorithm is adopted. Because ARIMA has good statistical properties and excellent flexibility, it is one of the most commonly used linear models for time series prediction [53].

The proposed online prediction model processes the collected vehicle trajectory and motion feature parameters in the order in which they arrive and simultaneously updates the model parameters. Subsequently, the trajectory and motion feature parameters are predicted for a period based on the constantly updated model. This type of processing is consistent with the characteristics of quantitative observation sequences over time.

Since they are influenced by the driver, the trajectory and motion feature parameters of the vehicle are generally not in a steady-state and may contain some deterministic trends. The differential method is an effective method for dealing with high sequential correlation. The ARIMA model is defined as follows:

$$D^d X_t = \sum_{i=1}^{q} \beta_i \, \epsilon_{t-i} + \sum_{i=1}^{k} \alpha_i D^d X_{t-i} + \epsilon_t \quad (3)$$

where D^d represents the d-th order difference, D is expressed as a differential symbol, X_t denotes the observation at time t. $\sum_{i=1}^{q} \beta_i \, \epsilon_{t-i}$ represents the moving average (MA) model. $\sum_{i=1}^{k} \alpha_i D^d X_{t-i}$ is the autoregressive (AR) model. ϵ_t is the zero-mean noise term. α and β are weight vectors. k, d, and q are parameterized terms.

We assume that X_t satisfies the ARIMA (k, d, q) model, the prediction of X_t over time can be achieved by an inverse differential process. The predicted value at time t+1 as $\tilde{X}_t$ can be expressed by the following equation:

$$\tilde{X}_t = D^d \tilde{X}_t + \sum_{i=0}^{d-1} D^i X_{t-1} \quad (4)$$

For the ARIMA model of Equation (3), due to the existence of a noise term ϵ_t, the existing online convex optimization technique cannot be applied to estimate the coefficient vector. Therefore, an ARIMA ($k + m$, d, 0) model that approximates the original ARIMA (k, d, q) and has no noise terms is designed. The expression is as follows,

$$\tilde{X}_t(\gamma^t) = \sum_{i=1}^{k+m} \gamma_i D^d X_{t-i} + \sum_{i=0}^{d-1} D^i X_{t-1} \quad (5)$$

where m is a constant, and γ is the new weight vector.

The loss function is:

$$\ell_t^m(\gamma^t) = \ell_t(X_t, \tilde{X}_t(\gamma^t)) = \ell_t(X_t, (\sum_{i=1}^{k+m} \gamma_i D^d X_{t-i} + \sum_{i=0}^{d-1} D^i X_{t-1})) \quad (6)$$

The goal of online ARIMA learning is to minimize the sum of losses over multiple iterations. After eliminating the noise term, we use the OGD method as an online convex optimization solver in this study [54]. More details on the OGD optimization technique can be found in [55].

The prediction algorithm needs a large dataset to update the model and cope with the rapidly changing vehicle trajectory and motion parameters to predict the future trend accurately. As the acquired data volume increases, the prediction result approaches the actual value.

3. Experimental Data

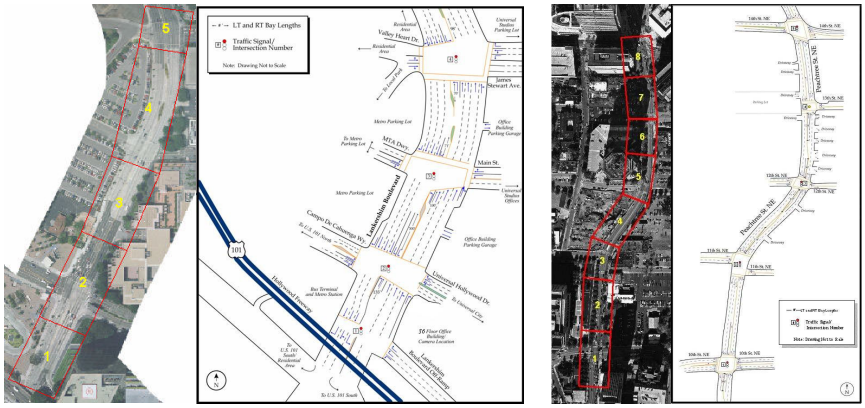
The mathematical model discussed in Section 2 requires a large amount of data for successful model training. The data must also be carefully selected to enable researchers to train models that correspond to the expected vehicle events.

3.1. Data Description

We use the open-source NGSIM dataset to verify the performance of the proposed hybrid method; the dataset was provided by the FHWA NGSIM project. This dataset has been widely used to develop and test various models [28,33,40,56–58]. A literature review indicated that most scholars investigated lane changes using the US-101 and I-80 dataset, but few had analyzed the turning behavior at intersections using this dataset. The NGSIM Lankershim and Peachtree street datasets are used in this work to develop the behavior recognition model for vehicles entering intersections. The behavior of the vehicle includes turning left (TL), turning right (TR), and going straight (GS).

Figure 5 is a schematic diagram of the intersection area investigated in this work. The red boxes and labels in the left picture of Figure 5a,b are the study area and camera coverage, and the right picture is the schematic diagram of the study area. Figure 5a shows Lankershim Boulevard, an artery running primarily north-south in Los Angeles, California. The speed limit on Lankershim Boulevard is 35 mph. Figure 5b shows Peachtree Street, which is a main road in Atlanta, Georgia; it runs from north to south, with a speed limit of 35 mph. As shown, there are four intersections on Lankershim and five intersections on Peachtree. Lankershim road is wide and is approximately 2,100 feet in length, and Peachtree Street is narrow and is approximately 1,600 feet in length. It also includes T-shaped intersections. The Lankershim data were collected on 16 June 2005 from 8:45 to 9:00 in the morning during the peak commuting period. The collection time of the Peachtree street data was from 12:45 to 1:00 and 4:00 to 4:15 on the afternoon of 8 November 2006.

The dataset contains information on the vehicle's lateral and longitudinal position, speed, acceleration, vehicle type, lane ID, and time/space headway, which were obtained from the video trajectory data at a resolution of 10 frames/s using a tracking algorithm [33]. In this work, we used some of the available variables.



(**a**) (**b**)

Figure 5. (**a**) Lankershim Boulevard. (**b**) Peachtree Street.

3.2. Data Extraction

The purpose of this work is to recognize the TL, TR, or GS behavior of the vehicle when approaching the intersection before initiating the maneuver. In other words, we predict the future

maneuver of the vehicle rather than classify the ongoing turning events to provide support for new generation ADAS or future intelligent vehicles or intelligent transportation systems. The process of the vehicle entering the intersection and turning includes the following sub-processes: first, the vehicle starts to decelerate, then the driver turns the steering wheel to change the vehicle's direction of travel, and when the direction meets the driver's needs, the driver starts to accelerate and drives away from the intersection. Figure 6 shows the process of a vehicle traveling from south to north, entering an intersection and turning left.

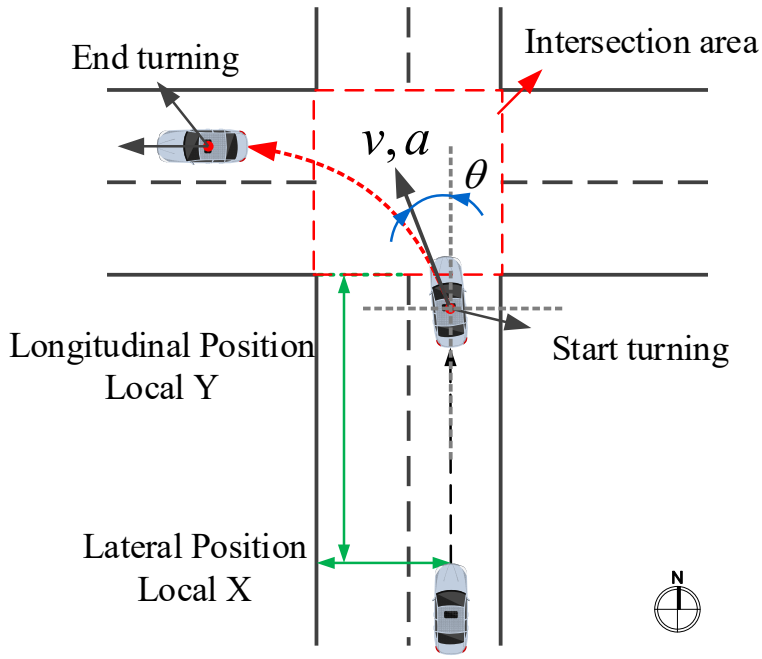


Figure 6. The process of a vehicle entering the intersection.

Due to the influence of the length of the intersection or the habits of the driver, the process of passing through the intersection has different durations. The turning process needs to be extracted to ensure that the differences between the samples are not excessive and to shorten the training time of the model. In this work, the extracted sample time for the vehicle approaching the intersection is 11 s. The "Movement" feature in the dataset is used to indicate the vehicle's current maneuvering state. When the value of movement is 1, it denotes GS, 2 denotes TL, and 3 denotes TR. However, when the trajectory information of the vehicle is combined with these data, it is found that the vehicle state indicated by this value is not accurate, and it is difficult to determine the starting time and ending time of the vehicle turning maneuver accurately. The heading angle is often regarded as a crucial parameter of the vehicle during the turning maneuver. The starting and ending time of a turning maneuver can be determined by observing the course of the heading angle. Normally, after a turning maneuver is completed, the vehicle heading angle has changed by about 90°. According to vehicle kinematics, the heading angle θ defined in this work is calculated as follows:

$$\theta = \arctan\left(\frac{\dot{y}}{\dot{x}}\right) = \arctan\left(\frac{v_y}{v_x}\right) \tag{7}$$

where x and y represent the lateral and longitudinal coordinates of the vehicle, and v_x and v_y are the lateral and longitudinal speeds of the vehicle, respectively. The starting time t_s is the initial time of the increasing or decreasing part on the curve, and the ending time t_e is the end time of the increasing or decreasing part on the curve [24], as shown in Figure 7. It should be noted that we define the heading angle as a negative value when TL and a positive value when TR (Figure 7), and the heading angles corresponding to the starting time and ending time set here are not exactly 0° and 90° or −90° because there are fluctuations. The extraction process follows the following criteria:

1. Identify the ID of the vehicle TL or TR;
2. Calculate the heading angle of the vehicle based on the trajectory information of the vehicle;
3. Search the starting time t_s when the vehicle begins to turn and mark it;
4. Using the t_s as a reference, 11 s is extracted from the time series of the entire turning process, including the time series of 10 s before t_s and 1 s after t_s.

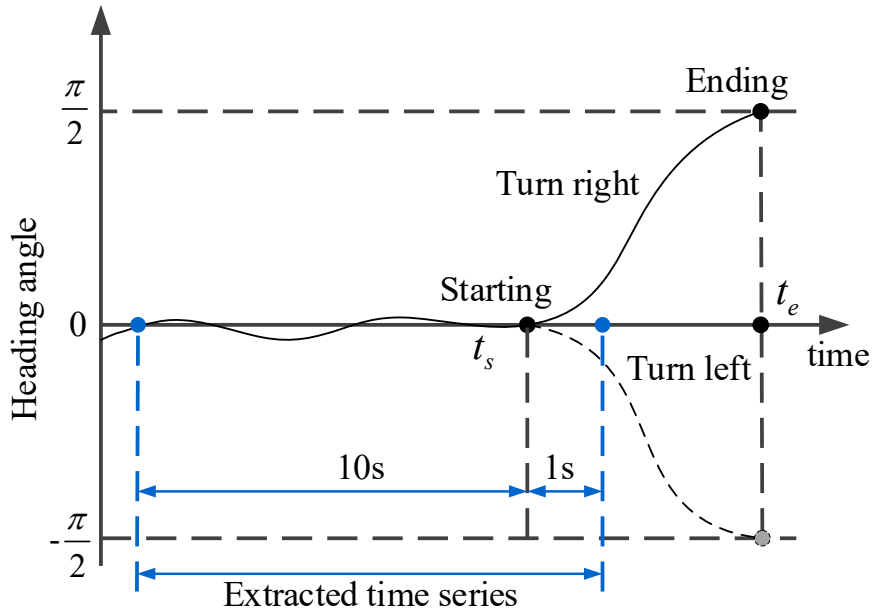


Figure 7. The process of entering the intersection.

After this extraction process and manual selection, invalid data are eliminated, such as short-duration turning or GS maneuver samples; eventually, 2993 sets of sample data are extracted. The data of the different road sections and different time periods are summarized in Table 1. The number of maneuvers in the Peachtree dataset is relatively small because we eliminated many invalid data and short-duration data.

Table 1. Summary statistics.

Dataset	Time Period	Going Straight	Left-Turn	Right-Turn	Total
Lankershim	8:30–8:45 a.m.	341	265	315	921
Lankershim	8:45–9:00 a.m.	341	302	339	982
Peachtree	2:45–1:00 p.m.	151	218	173	542
Peachtree	4:00–4:15 p.m.	143	254	151	548
Total	1 h	976	1039	978	2993

3.3. Input and Output Variable

The raw dataset of Lankershim and Peachtree cannot be used directly in the model because it has noise and errors. The locally weighted scatterplot smoother (LOWESS) algorithm was adopted to smooth the extracted data [59], as shown in Figure 8. The speed, acceleration, and the lateral and longitudinal position of the TL maneuvers are filtered. The use of filtered data does not only accelerate the convergence of the loss function during the model training phase but also makes the online prediction algorithm more stable. Note that the distance and speed units in the original dataset are feet and feet per second, which are converted into international units of meters and meters per second.

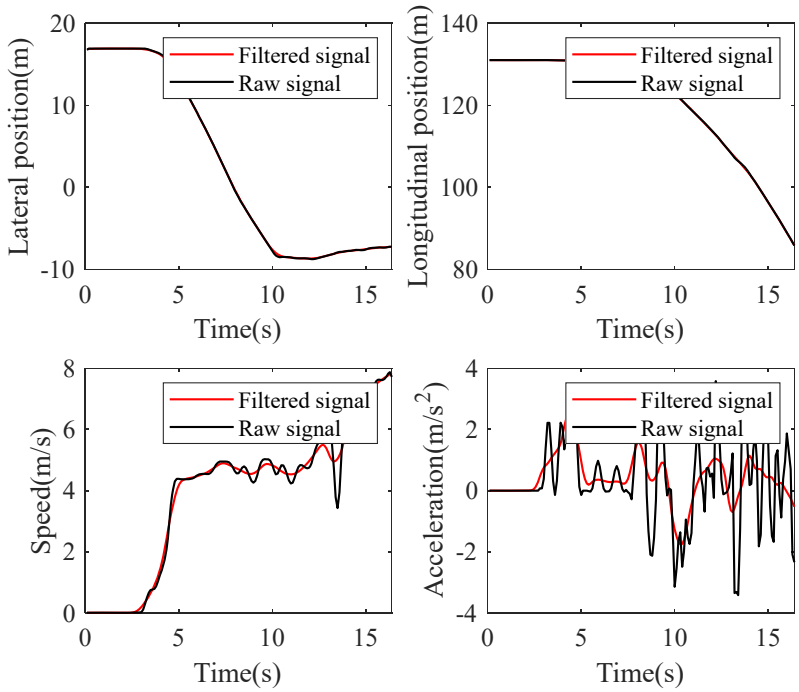


Figure 8. The process of entering the intersection.

The parameters used in this work to characterize the vehicle's motion are the vehicle's lateral and longitudinal positions, speed, and acceleration. As shown in Figure 6, Local X (x) indicates the lateral coordinate of the front center of the vehicle with respect to the left-most edge of the section in the direction of travel, Local Y (y) denotes the longitudinal coordinate of the front center of the vehicle with respect to the edge of the section in the direction of travel, v and a are the speed and acceleration of the vehicle, respectively. It should be noted that because vehicles will enter the intersection from all directions, the trajectory information of different vehicles in the dataset will be quite different.

The input data $\mathbf{X}_t$ of the prediction algorithm consists of all observations from a particular trajectory segment at time t, and $\mathbf{Y}_t$ is the output of its prediction. $\mathbf{x}_t$ contains the lateral (x_t) and longitudinal (y_t) positions, as well as the speed (v_t) and acceleration (a_t), where:

$$\begin{aligned} \mathbf{X}_t &= [\mathbf{x}_{t-h+1}, \mathbf{x}_{t-h+2}, \mathbf{x}_{t-h+3}, \dots, \mathbf{x}_t] \\ \mathbf{x}_t &= [x_t, y_t, v_t, a_t] \\ \mathbf{Y}_t &= [\mathbf{y}_{t+1}, \mathbf{y}_{t+2}, \mathbf{y}_{t+3}, \dots, \mathbf{y}_p] \\ \mathbf{y}_t &= [x_t, y_t, v_t, a_t] \end{aligned} \tag{8}$$

Here, $\mathbf{X}_t$ is the given historical observation sequence and h is the sequence length; $\mathbf{Y}_t$ is the predicted sequence, and p is the predicted length.

In addition, the vehicle's lateral motion parameters reflect the vehicle's trajectory and status. Based on this, we use the local coordinates related to the road to calculate the parameters of the vehicle's lateral motion state, including the lateral speed v_x, lateral acceleration a_x, and the key variable, i.e., the heading angle θ. The input of the Bi-LSTM behavior recognition model is $\mathbf{X}_t^B$, where:

$$\mathbf{X}_t^B = [\mathbf{Y}_t, \mathbf{x}_t^{lat}] \tag{9}$$

Here, $\mathbf{x}_t^{lat} = [v_{xt}, a_{xt}, \theta_t]$.

3.4. Data Analysis

Different drivers have different driving preferences or characteristics for different maneuvers at intersections [15]. The driving speed is often the critical parameter reflecting the driving characteristics of the driver. By observing the changing characteristics of the driving speed, it is often possible to detect driving intentions early. For turning maneuvers, the change in the heading angle, which is a crucial parameter, can often be used as a critical indicator to determine the start and end of the maneuver. The vehicle speed and heading angle of the samples extracted from different datasets are statistically analyzed to understand the vehicle's turning maneuver, as shown in Figures 9 and 10. The abscissa in the figures is time with an interval of 0.2 s, and 0 indicates the starting time of the turning maneuvers. Note that the speed and heading angle are statistically analyzed only from 10 s before t_s to 1 s after t_s, and each boxplot contains all sample subjects for each maneuver. The speed characteristics in Figure 9 show that the standard deviation starts to decrease at around −5 s, regardless of whether it is a left turn or a right turn, indicating that the driver may have started to execute his intention 5 s before the start of the turning maneuver, and that this was reflected by the change in speed. The mean values of the speed show that the driver on Peachtree is relatively stable when TL and TR, and the trend in the speed is similar. The speed of the vehicle on Lankershim before the start of the TL is significantly slower than that before the start of the TR, but when approaching the t_s, the speeds of the two maneuvers are similar. The heading angle boxplots shown in Figure 10 indicate that the heading angle has a tendency to change from −3 s to −4 s before t_s, although there are some outliers in the data. This result indicates that the driver started to control the vehicle's heading 3 to 4 s before the initiation of the actual turning maneuver, such as merging or adjusting the vehicle's position However, we can also see from the comparison of TL in Figure 10a,c, and TR in Figure 10b,d that different intersections have various impacts on the vehicle's turning maneuver. Although the change in the heading angle is about 90°, this change differs for intersections of different lengths.

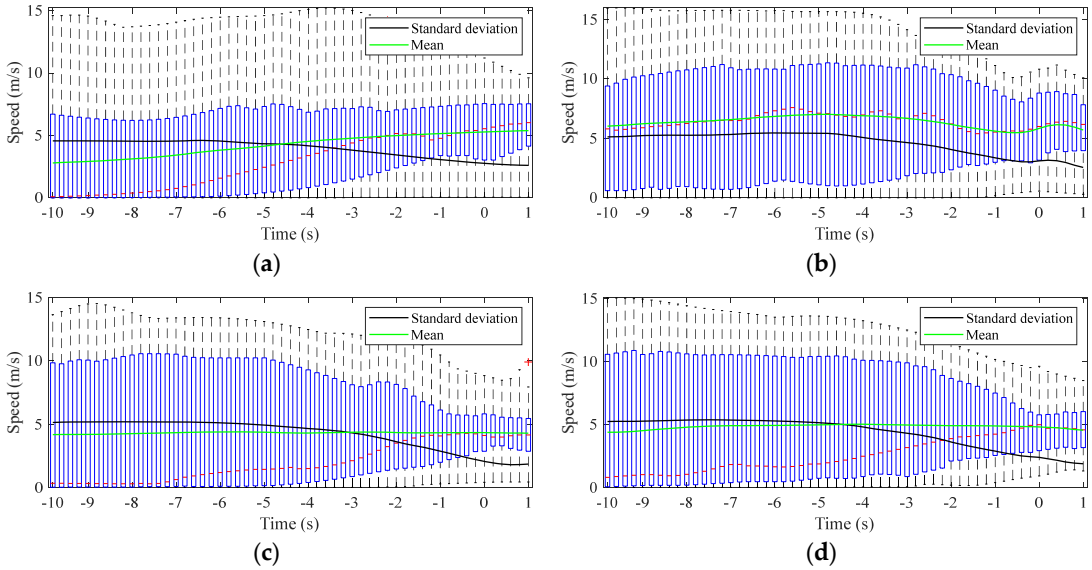


Figure 9. Boxplot of the speeds of the vehicle in the Lankershim and Peachtree datasets. (**a**) Statistics of the turn left (TL) speed in the Lankershim dataset. (**b**) Statistics of the turn right (TR) speed in the Lankershim dataset. (**c**) Statistics of the TL speed in the Pearchtree dataset. (**d**) Statistics of the TR speed in the Pearchtree dataset.

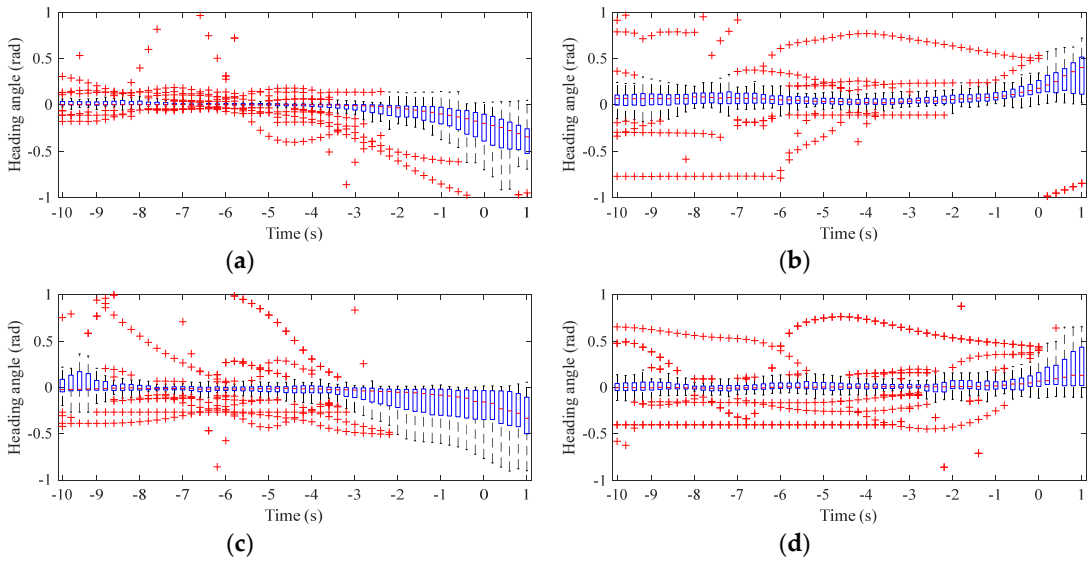


Figure 10. Boxplot of heading angles of the vehicles in Lankershim and Peachtree dataset. (**a**) Statistics of the TL heading angle in the Lankershim dataset. (**b**) Statistics of the TR heading angle in the Lankershim dataset. (**c**) Statistics of the TL heading angle in the Pearchtree dataset. (**d**) Statistics of the TR heading angle in the Pearchtree dataset.

3.5. Training and Test Procedure

3.5.1. Evaluation index for the online prediction algorithm

In this work, the mean absolute percentage error (MAPE) and the root mean square error (RMSE) are used as an indicator to reflect the forecasting accuracy of the proposed online prediction algorithm. The calculation formula of these two evaluation indexes are defined as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i} \times 100\%$$
$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (10)$$

where n is the total number of data, y_i and $\hat{y}_i$ are the real value and predicted value at the i-th time, respectively.

3.5.2. Training of the Behavior Recognition Model

The dataset was divided into training and testing datasets to develop and evaluate the model. In addition, five-fold cross-validation (CV) method is used to test the performance of the model. The extracted sample data are randomly and evenly divided into five folds; four folds are used to train the model, and one fold is used to evaluate the trained model. A total of five times are performed for such procedures. After implementing the test procedure, the model performances for detecting TL, TR, and GS are determined.

The receiver operating characteristic (ROC) curve was used to evaluate the model performance; this method evaluates the performance of a classifier by assessing the true positive rate (TPR) and false positive rate (FPR). The calculation formulas of the TPR and FPR are as follows:

$$TPR = \frac{TP}{TN + FN}$$
$$FPR = \frac{FP}{TN + FP} \quad (11)$$

where *TP*, *TN*, *FP*, and *FN* are the true positives, true negatives, false positives, and false negatives, respectively. The function provided by Matlab is used to calculate the ROC curve of the model, and *TP*, *TN*, *FP*, and *FN* are calculated after obtaining the true class labels and predicted scores of the test samples. According to the ROC curve of the model, the recognition performance of the model can be compared for different methods or premises by calculating the value of the area under the curve (AUC), where the value range of AUC is from 0 to 1. The larger the AUC value, the better the model performance is.

In this work, the developed Bi-LSTM network consists of 4 layers of stacks, the number of hidden units in each layer is 128, the activation function of the fully connected layer is ReLU, and the dropout rate is 0.9. A BP algorithm with the Adam stochastic optimization method is used to train the network over time with a learning rate of 0.001. The networks are trained using batches of size 80, and the epoch is 100. Note that all networks and algorithms are implemented in MATLAB 2019a under the Windows 10 Operation System and are evaluated on a PC with the following configuration: Intel Core i7-8700CPU at 3.20 GHz with 16 GB of RAM.

4. Results and Discussion

In this section, the sample data extracted in Section 3 are used to verify the performance of the proposed hybrid method for predicting the turning behavior. As mentioned before, we first use an online prediction algorithm to predict the vehicle's trajectory and state and use the predicted results as the input to the turning behavior recognition model to obtain the turning maneuver prediction results. The purpose is to predict the vehicle's intention as early as possible before the initiation of the maneuver. The basic parameters extracted from the dataset that characterize the vehicle's motion state include the lateral and longitudinal coordinates, as well as the speed and acceleration. Based on the above parameters, the derived parameters that intuitively characterize the vehicle's lateral state are obtained; these include the lateral speed, lateral acceleration, and heading angle. Therefore, the online prediction algorithm only predicts the basic parameters of the vehicle, and the derived parameters can be calculated based on the prediction results, thereby reducing the workload of the prediction algorithm and shortening the calculation time.

4.1. Performance of the Online Prediction Algorithm

The use of an offline model to predict the vehicle's future trajectory or motion state is often suitable for algorithm verification, but it does not meet the real-time requirements of the ADAS system, i.e., it cannot be used to evaluate the actual driving process. However, online prediction algorithms can achieve this, and they also meet people's expectations for predicting the movement of surrounding autonomous vehicles in the future. This algorithm predicts the vehicle's trajectory and movement in the future based on historical information, providing advanced prediction of the maneuver time. As mentioned in Section 2.2, in the algorithm, the model parameters are continuously updated based on the acquired data, and the future values are predicted based on the new model. In the beginning, the prediction algorithm has just received real-time data and has generated the model parameters. The data error based on the model prediction at this time will be relatively large; however, as more and more data enter the model, the prediction accuracy of the algorithm will increase. In this work, we do not start at the beginning of the extracted 11 s time series, but rather at 11 s before the t_s until the end of the sequence, and we set the prediction horizon to 1.5 s. As a result, more accurate real-time data are available for the subsequent turning behavior recognition model. Note that entering an intersection from different directions will result in large differences in the trajectories, as mentioned previously.

In this work, the lateral position, longitudinal position, speed, and acceleration of the TL and TR processes are used to verify the performance of the online ARIMA prediction algorithm. Table 2 presents the RMSEs and MAPEs of these parameters. Figures 11–13 are GS, TL, and TR processes randomly selected from the dataset, respectively. The performance of the proposed prediction algorithm by the RMSE in Table 2 and Figures 11–13 illustrates that the prediction algorithm accurately predicts the driving trajectory and motion state of the vehicle. It can be seen from Table 2

and Figures 11–13 that the algorithm can relatively accurately predict the trajectory and motion state of the vehicle in the predicted horizon, which shows that the algorithm can reflect the driving intention of the vehicle in the future to a certain extent.

Table 2. The root mean square error (RMSE) for vehicle's position, speed, and acceleration in straight through and turn scenarios.

Scenarios	Lateral Position (m)		Longitudinal Position (m)		Speed (m/s)		Acceleration (m/s²)	
	RMSE	MAPE (%)	RMSE	MAPE (%)	RMSE	MAPE (%)	RMSE	MAPE (%)
GS	0.0932	1.119	0.1093	1.028	0.1635	1.227	0.2381	0.043
TL	0.2719	0.162	0.1592	0.258	0.3674	0.184	0.1218	0.023
TR	0.1168	0.026	0.3954	0.200	0.1350	0.213	0.4007	0.058

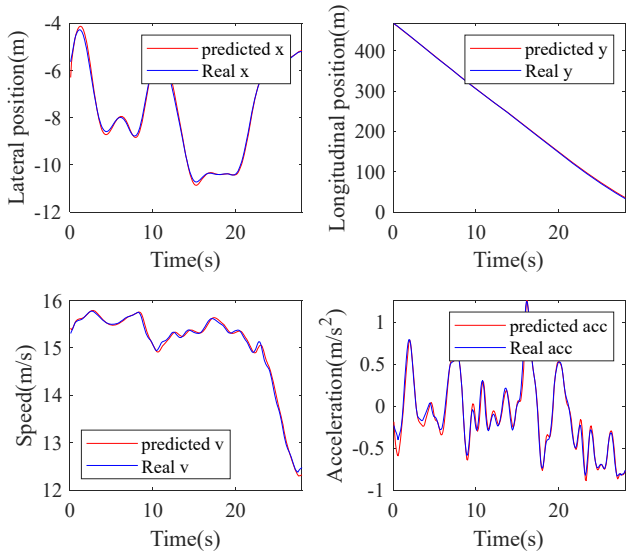


Figure 11. Going straight prediction.

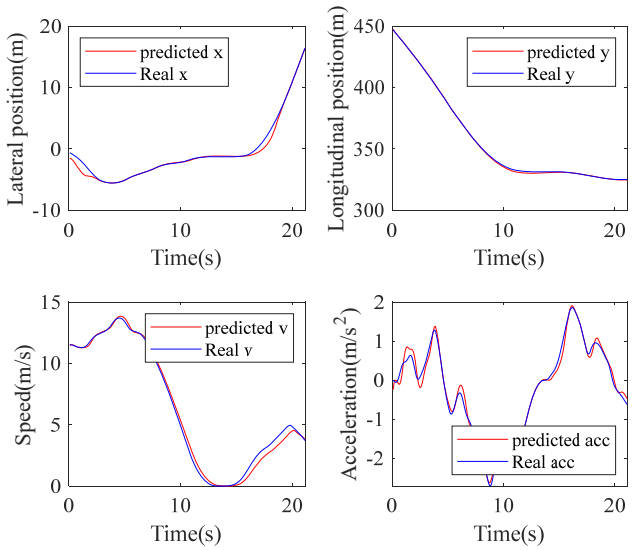


Figure 12. Turn left prediction.

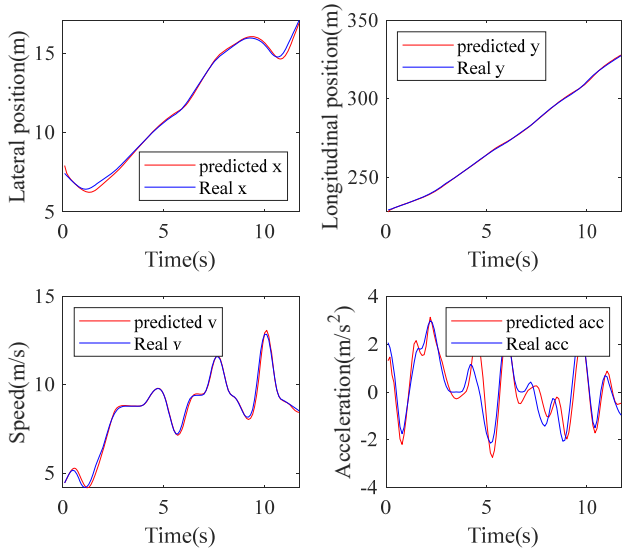


Figure 13. Turn right prediction.

4.2. Performance of the Hybrid Method for Turning Behavior Recognition

The performance of the proposed hybrid method is evaluated in terms of the recognition accuracy and behavior recognition time. A sliding time window is used to maximize the use of the data, as shown in Figure 14. The moving step is 0.1 s, i.e., each time the window moves forward, the data in the subsequent and previous windows always contains the same T_w - 0.1 s information. Since the window is relatively wide, the real-time performance of the model is affected, and large computing memory is required. After a comparative test, T_w is set to 1.5 s.

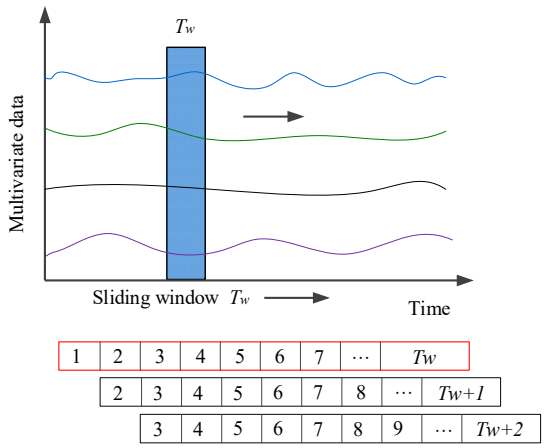


Figure 14. Sliding time window.

The performance of the proposed hybrid model for turning intention recognition combined with the online prediction algorithms and deep learning method is compared with the conventional machine learning method CHMM, and the LSTM network with the same architecture, and the LSTM with a similar structure with convolutional layer added (Conv-LSTM). Figure 15a,b shows the recognition results of the TL and TR maneuvers at the time of t_s, respectively. Because the driver has not performed the turning maneuver before t_s, the CHMM does not accurately recognize the maneuver. The ROC curves clearly show the superiority of the proposed hybrid approach over the CHMM-based, LSTM-based, and Conv-LSTM-based algorithms. Specifically, the recognition

accuracy of the proposed method at the turning moment t_s is 95.37% for TL with an AUC of 0.9733 and 96.07% for TR with an AUC of 0.9662. In comparison, the recognition accuracy of the CHMM is only 79.23% for TL with an AUC of 0.7968 and 74.33% for TR with an AUC of 0.7651. It can be seen from the figure that the Conv-LSTM network can also achieve better performance, which is better than that of the CHMM-based and LSTM-based algorithms, but is inferior to Bi-LSTM. This result reveals that the superiority of the deep learning method, and also reflects the good time series modeling capabilities of the Bi-directional LSTM in the turning case.

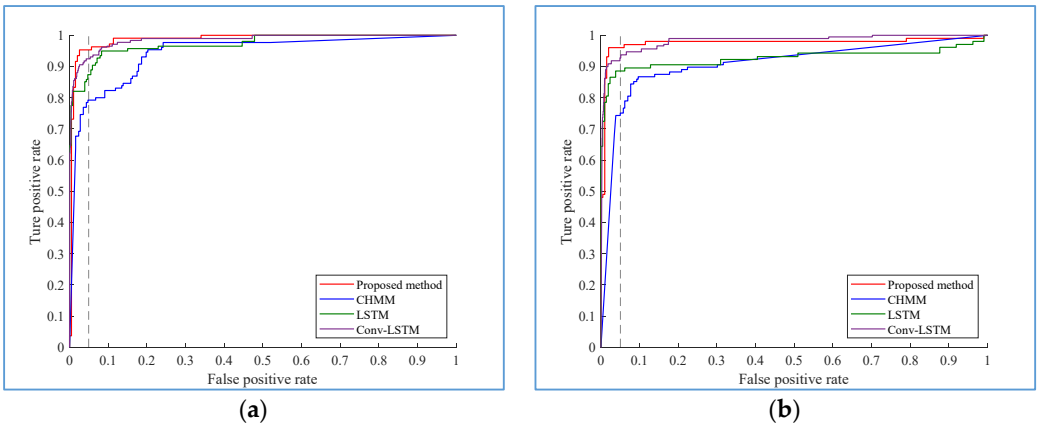


Figure 15. Receiver operating characteristic (ROC) curves for TL (**a**) and TR (**b**) based on the continuous hidden Markov model (CHMM) and the proposed hybrid algorithms at time t_s. The dashed line shows the true positive rate around a false positive rate of 5%.

The online prediction algorithm uses historical information to predict a future event; the LSTM network has strong capabilities for context modeling, sequential learning, and other nonlinear time series modeling. The Bi-LSTM analyzes data from both ends of the sequence and considers the effect of reverse timing, which improves its predictive ability. Therefore, the proposed hybrid method has strong prediction performance. Figures 16 and 17 show how quickly the proposed method can predict the intention of the driver before the initiation of the TL and TR maneuvers. The results indicate that for the TL and TR behaviors, the model has an average accuracy of 94.2% at 1 s before the maneuver, 93.5% at 2 s before the maneuver, and 74.5% at 3 s before the maneuver.

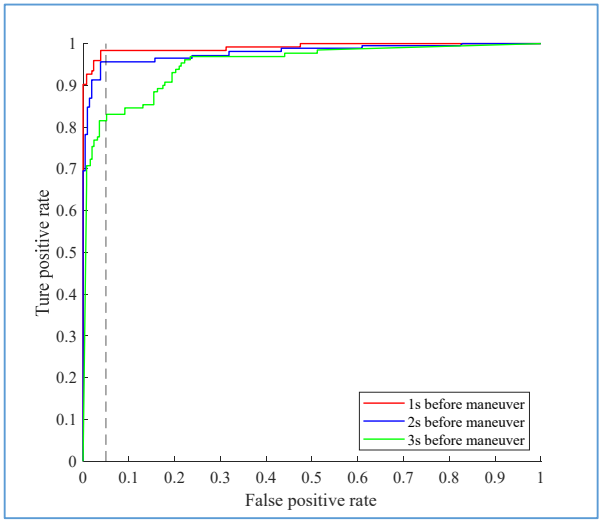


Figure 16. Performance of the proposed method at different times before the actual maneuver for the TL.

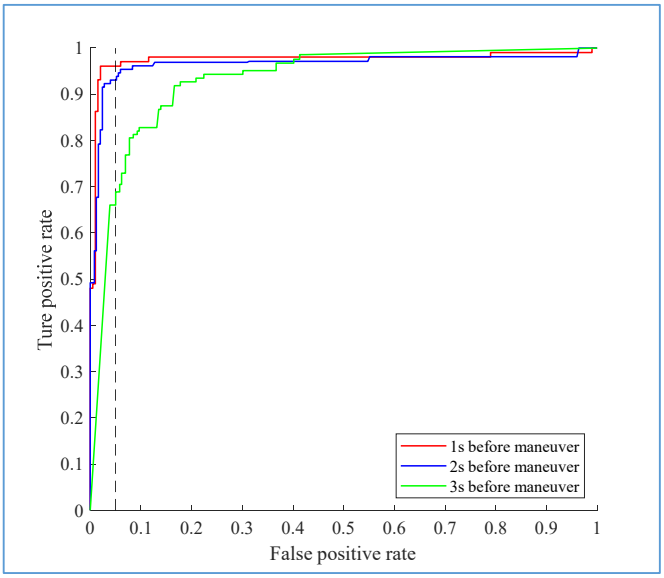


Figure 17. Performance of the proposed method at different times before the actual maneuver for the TR.

The time to recognize the turning behavior is very critical for ADAS or autonomous vehicles. The earlier the intent of the surrounding vehicles is recognized, the higher the probability is of achieving safe and comfortable driving. Figure 18 shows the driving behavior recognition time; (a) shows the recognition time for TL, (b) shows the recognition time for TR. The data are obtained using only Bi-LSTM to identify the samples. The results shown in (c) and (d) are the statistics of the recognition time of the TL and TR maneuvers, respectively, using the proposed hybrid method. For the TL maneuvers, it is observed in Figure 18a,c that the cumulative frequency of the recognition time of the Bi-LSTM is 86% at 1 s before t_s. The cumulative frequency of the proposed method is 84.8% and 96.36%, respectively at 2 s and 1 s before t_s by using the proposed hybrid method. For the TR maneuvers, we obtain similar results. The use of the hybrid method does not reduce the recognition accuracy much but increases the recognition time. The results show that the proposed hybrid approach provides early recognition of the intention of surrounding vehicles approaching the intersection.

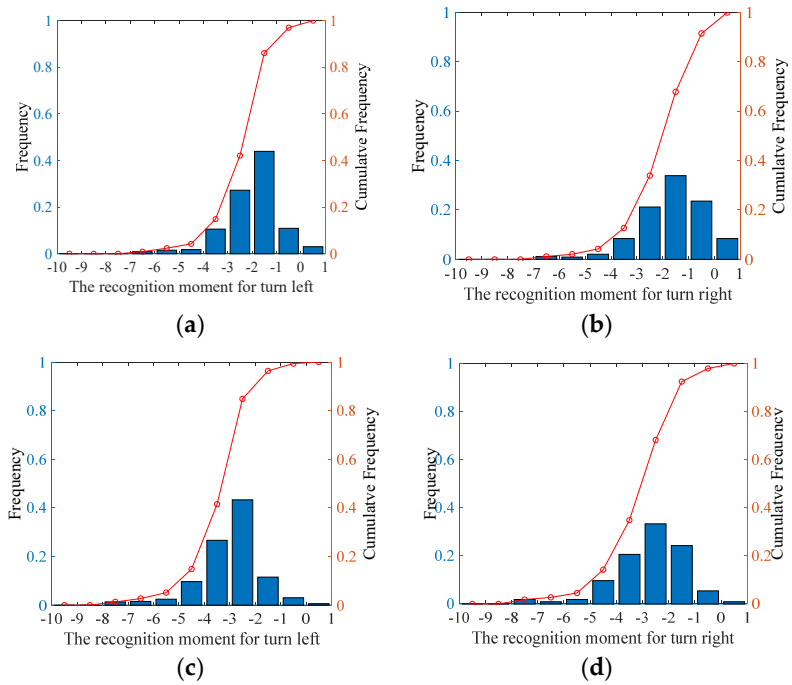


Figure 18. Statistical distribution for the correct recognition result. (**a**,**b**) shows the driving behavior recognition time: (**a**) for TL, (**b**) for TR. (**c**,**d**) are the statistics of the recognition time of the TL and TR maneuvers, respectively, using the proposed hybrid method.

The calculation time of the proposed method for a single sample is a crucial factor in determining whether the method is suitable for real vehicle experiments. Table 3 shows that the average runtime of the hybrid method, whose magnitude is 10 to the negative power of 2 s, indicating that the proposed method meets the real-time requirements of the system.

Table 3. Average runtime of the recognition.

Model	Online Prediction (s)	Bi-LSTM (s)	Total (s)
Average time	0.0013	0.0150	0.0163

5. Conclusions

In this work, a hybrid approach that combines time series prediction with deep learning networks is proposed to predict the intention of surrounding vehicles when approaching an intersection to improve driving safety. The performance of the proposed hybrid approach is verified using real natural driving data. The driving intention is predicted accurately by the proposed model with an average accuracy of 74.5%, 93.5%, and 94.2% at 3, 2, and 1 s, respectively, before the surrounding vehicles initiate the turning maneuver. The proposed approach can be used to alert drivers of human-driven vehicles of possible safety risks when entering an intersection or to plan a safe and comfortable driving path for ADAS. In the prediction stage, the trajectory, speed, and acceleration of the vehicle are predicted, and the lateral state parameters of the vehicle are derived based on the predicted value, including the lateral speed, lateral acceleration, and heading angle. In the behavior recognition stage, the output of the online prediction algorithm and its derived parameters are input into the Bi-LSTM behavior recognition model to obtain the behavior recognition result.

The use of time series prediction enables the proposed method to perceive the future driving trajectory and driving state of the vehicle. The OGD optimizes in the ARIMA algorithm allows for the online prediction of the vehicle trajectory and state. Due to the RNN, the Bi-LSTM has strong modeling ability, and the hybrid method has excellent predictive ability.

In actual intersections, Lidar or millimeter-wave radar can be used to detect the vehicle's trajectory, speed, and acceleration; therefore, the proposed hybrid approach may be achievable. In future studies, additional analysis and implementation will be performed to achieve faster detection and higher recognition rates of the intention of the surrounding vehicles after including additional information from the infrastructure, such as traffic light information.

Author Contributions: H.Z. conceived of and designed the method. H.Z. analyzed the experimental data. Finally, H.Z. wrote the paper with the help of R.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1600500, in part by the National Natural Science Foundation of China under Grant 51908054, in part by the Key Research and Development Program of Shannxi under Grant (2020GY-163，2019ZDLGY03-09-02), and in part by the Fundamental Research Funds for the Central Universities, CHD 300102220202.

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

References

1. Zhou, M.; Yu, Y.; Qu, X. Development of an Efficient Driving Strategy for Connected and Automated Vehicles at Signalized Intersections: A Reinforcement Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 433–443.
2. Sun, J.; Qi, X.; Xu, Y.; Tian, Y. Vehicle Turning Behavior Modeling at Conflicting Areas of Mixed-Flow Intersections Based on Deep Learning. *IEEE Trans. Intell. Transp. Syst.* **2019**, doi:10.1109/TITS.2019.2931701.
3. Deo, N.; Rangesh, A.; Trivedi, M.M. How would surround vehicles move? A Unified Framework for Maneuver Classification and Motion Prediction. *IEEE Trans. Intell. Veh.* **2018**, *3*, 129–140.
4. Noh, S. Decision-Making Framework for Autonomous Driving at Road Intersections: Safeguarding Against Collision, Overly Conservative Behavior, and Violation Vehicles. *IEEE Trans. Ind. Electron.* **2018**, *66*, 3275–3286.
5. Wang, J.; Peeta, S.; He, X. Multiclass traffic assignment model for mixed traffic flow of human-driven vehicles and connected and autonomous vehicles. *Transp. Res. Part B Methodol.* **2019**, *126*, 139–168, doi:10.1016/j.trb.2019.05.022.
6. Papadoulis, A.; Quddus, M.; Imprialou, M. Evaluating the safety impact of connected and autonomous vehicles on motorways. *Accid. Anal. Prev.* **2019**, *124*, 12–22, doi:10.1016/j.aap.2018.12.019.
7. Fu, R.; Li, Z.; Sun, Q.; Wang, C. Human-like car-following model for autonomous vehicles considering the cut-in behavior of other vehicles in mixed traffic. *Accid. Anal. Prev.* **2019**, *132*, 105260.
8. Xu, T.; Jiang, R.; Wen, C.; Liu, M.; Zhou, J. A hybrid model for lane change prediction with V2X-based driver assistance. *Phys. A Stat. Mech. Appl.* **2019**, *15*, 122033.
9. Ali, Y.; Zheng, Z.; Haque, M.M.; Yildirimoglu, M.; Washington, S. Understanding the discretionary lane-changing behaviour in the connected environment. *Accid. Anal. Prev.* **2020**, *137*, 105463, doi:10.1016/j.aap.2020.105463.
10. Ali, Y.; Zheng, Z.; Haque, M.M.; Wang, M. A game theory-based approach for modelling mandatory lane-changing behaviour in a connected environment. *Transp. Res. Part C Emerg. Technol.* **2019**, *106*, 220–242.
11. Ali, Y.; Zheng, Z.; Haque, M.M. Connectivity's impact on mandatory lane-changing behaviour: Evidences from a driving simulator study. *Transp. Res. Part C Emerg. Technol.* **2018**, *93*, 292–309, doi:10.1016/j.trc.2018.06.008.
12. Choi, E.H. *Crash Factors in Intersection-Related Crashes: An On-Scene Perspective*; Mathematical Analysis Division, National Center for Statistics and Analysis, National Highway Traffic Safety Administration: Washington, DC, USA, 2010.
13. Liu, C.; Ye, T.J. *Run-Off-Road Crashes: An On-Scene Perspective*; Mathematical Analysis Division, National Center for Statistics and Analysis, National Highway Traffic Safety Administration: Washington, DC, USA, **2011**.
14. National Highway Traffic Safety Administration. Fatality analysisi reporting system encyclopedia. 2009. Available online: http://www-fars.nhtsa.dot.gov/Main/index.aspx (accessed on 24 June 2020).
15. Yi, D.; Su, J.; Liu, C.; Chen, W.H. Trajectory Clustering Aided Personalized Driver Intention Prediction for Intelligent Vehicles. *IEEE Trans. Ind. Inform.* **2018**, *15*, 3693–3702.

16. Bougler, B.; Cody, D.; Nowakowski, C. *California Intersection Decision Support: A Driver-Centered Approach to Left-Turn Collision Avoidance System Design*; California Partners for Advanced Transportation Technology, UC Berkeley: Berkeley, CA, USA, 2008.
17. Ghiasi, A.; Li, X.; Ma, J. A mixed traffic speed harmonization model with connected autonomous vehicles. *Transp. Res. Part C-Emerg. Technol.* **2019**, *104*, 210–233, doi:10.1016/j.trc.2019.05.005.
18. Chen, C.; Liu, L.; Qiu, T.; Ren, Z.; Hu, J.; Ti, F. Driver's Intention Identification and Risk Evaluation at Intersections in the Internet of Vehicles. *IEEE Internet Things J.* **2018**, *5*, 1575–1587, doi:10.1109/jiot.2017.2788848.
19. Yao, Y.; Zhao, X.; Wu, Y.; Zhang, Y.; Rong, J. Clustering driver behavior using dynamic time warping and hidden Markov model. *J. Intell. Transp. Syst.* **2019**, *24*, 1–14.
20. Schorr, J.; Hamdar, S.H.; Silverstein, C. Measuring the safety impact of road infrastructure systems on driver behavior: Vehicle instrumentation and real world driving experiment. *J. Intell. Transp. Syst.* **2017**, *21*, 364–374.
21. Chen, Z.; Wu, C.; Huang, Z.; Lyu, N.; Hu, Z.; Zhong, M.; Cheng, Y.; Ran, B. Dangerous driving behavior detection using video-extracted vehicle trajectory histograms. *J. Intell. Transp. Syst.* **2017**, *21*, 409–421.
22. Chen, X.; Zhou, L.; Li, L. Bayesian network for red-light-running prediction at signalized intersections. *J. Intell. Transp. Syst.* **2019**, *23*, 120–132.
23. Li, M.; Chen, X.; Lin, X.; Xu, D.; Wang, Y. Connected vehicle-based red-light running prediction for adaptive signalized intersections. *J. Intell. Transp. Syst.* **2018**, *22*, 229–243.
24. Zhang, M.; Fu, R.; Morris, D.; Wang, C. A Framework for Turning Behavior Classification at Intersections Using 3D LIDAR. *IEEE Trans. Veh. Technol.* **2019**, *68*, 7431–7442.
25. Gadepally, V.; Krishnamurthy, A.; Ozguner, U. A Framework for Estimating Driver Decisions Near Intersections. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 637–646.
26. Jin, L.; Hou, H.; Jiang, Y. Driver intention recognition based on Continuous Hidden Markov Model. In Proceedings of the International Conference on Transportation, Mechanical, and Electrical Engineering, Changchun, China, 16–18 December 2011; pp. 739–742.
27. Li, K.; Wang, X.; Xu, Y.; Wang, J. Lane changing intention recognition based on speech recognition models. *Transp. Res. Part C Emerg. Technol.* **2016**, *69*, 497–514, doi:10.1016/j.trc.2015.11.007.
28. Zhang, Y.; Lin, Q.; Wang, J.; Verwer, S.; Dolan, J.M. Lane-Change Intention Estimation for Car-Following Control in Autonomous Driving. *IEEE Trans. Intell. Veh.* **2018**, *3*, 276–286, doi:10.1109/tiv.2018.2843178.
29. Amsalu, S.B.; Homaifar, A. Driver Behavior Modeling Near Intersections Using Hidden Markov Model Based on Genetic Algorithm. In Proceedings of the 2016 IEEE International Conference on Intelligent Transportation Engineering (ICITE), Singapore, 20–22 August 2016; pp. 193–200.
30. Kumar, P.; Perrollaz, M.; Lefevre, S.; Laugier, C. Learning-based approach for online lane change intention prediction. In Proceedings of the Intelligent Vehicles Symposium, Gold Coast, QLD, Australia, 23–26 June 2013; pp. 797–802.
31. 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.
32. Peng, J.; Guo, Y.; Fu, R.; Yuan, W.; Wang, C. Multi-parameter prediction of drivers' lane-changing behaviour with neural network model. *Appl. Ergon.* **2015**, *50*, 207–217, doi:10.1016/j.apergo.2015.03.017.
33. Yi, H.; Edara, P.; Sun, C. Situation assessment and decision making for lane change assistance using ensemble learning methods. *Expert Syst. Appl*. **2015**, *42*, 3875–3882.
34. Maier, A.; Syben, C.; Lasser, T.; Riess, C. A gentle introduction to deep learning in medical image processing. *Z. Med. Phys.* **2019**, *29*, 86–101.
35. Kumar, A.; Verma, S.; Mangla, H. A Survey of Deep Learning Techniques in Speech Recognition. In Proceedings of the 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida (UP), India, 12–13 October 2018; pp. 179–185.
36. Zyner, A.; Worrall, S.; Nebot, E. Naturalistic driver intention and path prediction using recurrent neural networks. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1584–1594.
37. Graves, A. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780.
38. Zyner, A.; Worrall, S.; Nebot, E.J.I.R.; Letters, A. A Recurrent Neural Network Solution for Predicting Driver Intention at Unsignalized Intersections. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1759–1764.
39. Zyner, A.; Worrall, S.; Ward, J.; Nebot, E. Long short term memory for driver intent prediction. In Proceedings of the Intelligent Vehicles Symposium, Stockholm, Sweden, 16–21 May 2016; pp. 1484–1489.

40. Phillips, D.J.; Wheeler, T.A.; Kochenderfer, M.J. Generalizable intention prediction of human drivers at intersections. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017.
41. Jain, A.; Singh, A.; Koppula, H.S.; Soh, S.; Saxena, A.; Jain, A.; Singh, A.; Koppula, H.S.; Soh, S.; Saxena, A. Recurrent Neural Networks for Driver Activity Anticipation via Sensory-Fusion Architecture. In Proceedings of the IEEE International Conference on Robotics and Automation, Stockholm, Sweden, 16–21 May 2016; pp. 699–706.
42. Liebner, M.; Klanner, F.; Baumann, M.; Ruhhammer, C.; Stiller, C. Velocity-Based Driver Intent Inference at Urban Intersections in the Presence of Preceding Vehicles. *IEEE Intell. Transp. Syst. Mag.* **2013**, *5*, 10–21, doi:10.1109/mits.2013.2246291.
43. Tan, Y.V.; Elliott, M.R.; Flannagan, C.A. Development of a real-time prediction model of driver behavior at intersections using kinematic time series data. *Accid. Anal. Prev.* **2017**, *106*, 428–436.
44. Jain, A.; Koppula, H.S.; Raghavan, B.; Soh, S.; Saxena, A. Car that Knows Before You Do: Anticipating Maneuvers via Learning Temporal Driving Models. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 13–16 December 2015.
45. Lethaus, F.; Baumann, M.R.K.; Köster, F.; Lemmer, K.J.N. A comparison of selected simple supervised learning algorithms to predict driver intent based on gaze data. *Neurocomputing* **2013**, *121*, 108–130.
46. Deo, N.; Trivedi, M.M. Multi-Modal Trajectory Prediction of Surrounding Vehicles with Maneuver based LSTMs. In Proceedings of the IEEE intelligent vehicles symposium, Changshu, China, 26–30 June 2018; pp. 1179–1184.
47. Woo, H.; Yonghoon, J.I.; Kono, H.; Tamura, Y.; Kuroda, Y.; Sugano, T.; Yamamoto, Y.; Yamashita, A.; Asama, H.J.I.R.; Letters, A. Lane-Change Detection Based on Vehicle-Trajectory Prediction. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1109–1116.
48. Lee, S.; Khan, M.Q.; Husen, M.N. Continuous Car Driving Intent Detection Using Structural Pattern Recognition. *IEEE Trans. Intell. Transp. Syst.* **2020**, 1–13, doi:10.1109/TITS.2019.2961928.
49. Fawaz, H.I.; Forestier, G.; Weber, J.; Idoumghar, L.; Muller, P.A.J.D.M.; Discovery, K. Deep learning for time series classification: A review. *Data Min. Knowl. Discov.* **2019**, *33*, 917–963.
50. Ye, R.; Dai, Q. MultiTL-KELM: A multi-task learning algorithm for multi-step-ahead time series prediction. *Appl. Soft Comput.* **2019**, *79*, 227–253, doi:10.1016/j.asoc.2019.03.039.
51. Liu, G.; Guo, J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* **2019**, *337*, 325–338.
52. Wang, S.; Wang, X.; Wang, S.; Wang, D. Bi-directional long short-term memory method based on attention mechanism and rolling update for short-term load forecasting. *Int. J. Electr. Power Energy Syst.* **2019**, *109*, 470–479.
53. Liu, C.; Hoi, S.C.H.; Zhao, P.; Sun, J. Online ARIMA algorithms for time series prediction. In Proceedings of the national conference on artificial intelligence, Phoenix, AZ, USA, 12–17 February 2016; pp. 1867–1873.
54. Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In Proceedings of the 20th International Conference on Machine Learning (ICML-03), Pittsburgh, PA, USA; pp. 928–936.
55. Biehl, M.; Schwarze, H. Learning by online gradient descent. *J. Phys. A* **1995**, *28*, 643–656.
56. Yang, D.; Zhu, L.; Ran, B.; Pu, Y.; Hui, P. Modeling and Analysis of the Lane-Changing Execution in Longitudinal Direction. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 2984–2992, doi:10.1109/tits.2016.2542109.
57. Yang, D.; Zhu, L.; Liu, Y.; Wu, D.; Ran, B. A Novel Car-Following Control Model Combining Machine Learning and Kinematics Models for Automated Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 1991–2000.
58. Zhang, X.; Sun, J.; Qi, X.; Sun, J. Simultaneous modeling of car-following and lane-changing behaviors using deep learning. *Transp. Res. Part C Emerg. Technol.* **2019**, *104*, 287–304.
59. Yang, D.; Zhu, L.; Yang, F.; Pu, Y. Modeling and analysis of lateral driver behavior in lane-changing execution. *Transp. Res. Rec. J. Transp. Res. Board* **2015**, *2490*, 127–137.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).