Research Article

Learning-Based Lane-Change Behaviour Detection for Intelligent and Connected Vehicles

Luyao Du,1 Wei Chen,1 Zhonghui Pei,2 Hongjiang Zheng,3,4 Shuaizhi Fu,1 Kang Chen,1 and Di Wu5,6

1School of Automation, Wuhan University of Technology, Wuhan 430070, China
2School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China
3Shanghai Engineering Technology Research Center for Intelligent and Connected Vehicle Terminals, Shanghai 200030, China
4Shanghai PATEO Electronic Equipment Manufacturing Co., Ltd., Shanghai 200030, China
5Key Laboratory of Environment Change and Resources Use in Beibu Gulf, Nanning Normal University, Ministry of Education, Nanning 530001, China
6GNSS Research Center of Wuhan University, Wuhan 430000, China

Correspondence should be addressed to Di Wu; 29649243@qq.com

Received 17 June 2020; Revised 17 August 2020; Accepted 31 August 2020

Academic Editor: Nian Zhang

Copyright © 2020 Luyao Du et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Detection of lane-change behaviour is critical to driving safety, especially on highways. In this paper, we proposed a method and designed a learning-based detection model of lane-change behaviour in highway environment, which only needs the vehicle to be equipped with velocity and direction sensors or each section of the highway to have a video camera. First, based on the Next Generation Simulation (NGSIM) Interstate 80 Freeway Dataset, we analyzed the relevant features of lane-changing behaviour and preprocessed the data and then used machine learning algorithms to select the suitable features for lane-change detection. According to the result of feature selection, we chose the lateral velocity of the vehicle as the lane-change feature and used machine learning algorithms to learn the lane-change behaviour of the vehicle to detect it. From the dataset, continuous data of 14 vehicles with frequent lane changes were selected for experimental analysis. The experimental results show that the designed KNN lane-change detection model has the best performance with detection accuracy between 89.57% and 100% on the selected dataset, which can well complete the vehicle lane-change detection task.

1. Introduction

Over the past few years, with the rapid development of artificial intelligence and communication technology, intelligent vehicles based on intelligence and networking have become a major trend in the development of the automotive industry. From the perspective of technological development, intelligent vehicles are divided into three development directions: connected vehicle (CV), autonomous vehicle (AV), and the integration of the former two, namely, connected and automated vehicle (CAV) or intelligent and connected vehicle (ICV) [1].

ICVs play an important role in improving driving safety and reducing driver burden, contribute to energy conservation and environmental protection, and improve traffic efficiency. Research shows that, in the initial stage of ICVs, advanced intelligent driving assistance technology can help reduce traffic accidents by about 30%, improve traffic efficiency by 10%, and reduce fuel consumption and emissions by 5% [2]. Entering the ultimate stage of the ICVs, that is, the fully automatic driving stage, it could avoid traffic accidents, improve traffic efficiency, and finally liberate people from boring driving tasks.

Driving behaviour detection plays a significant role in ICVs’ decision-making system. During the driving of the vehicle, due to the driver’s lack of attention or the obstruction of the surrounding large vehicles, it is likely that the driver will not be able to fully understand the driving
conditions of the surrounding vehicles, thus causing great safety risks. Many methods of lane-change behaviour detection have been proposed by researchers in recent years, including hidden Markov model (HMM) [3–5], multi-view convolutional neural network model (MV-CNN) [6], and vision-based deep residual neural network (RNN) [7]. Detection of lane-changing behaviour in different scenarios, including highways [8–12] and signalized intersections [13–16], has also been studied by many researchers. Steering behaviour recognition [17] and prediction [18] methods have been proposed, too. Besides, some new deep learning and machine learning methods have also been proposed in recent years. Xie et al. comprehensively modeled lane-change using deep learning approaches including deep belief network and long short-term memory [19]. Xing et al. proposed an ensemble bi-directional LSTM model for driver intention inference [20]. Gao et al. proposed a data-driven lane-change detection system using deep learning techniques [21]. Zhang et al. modeled the car following and lane-changing behaviours simultaneously using hybrid retraining constrained long short-term memory neural networks [22]. Zhao et al. proposed a new quantitative discriminant model based on deep belief networks algorithm and the classification analysis method based on support vector machine [23]. Dang and Dai established a lane-change model based on improved Bayesian network [24]. These methods, however, need prior knowledge, or the structure is complex and the real-time performance can be improved. In practical application scenarios, there is usually lack of prior knowledge of data distribution, and a simpler classification method is easier to implement.

In this paper, we proposed a method and designed a learning-based detection model of lane-change behaviour on highways, which only needs the vehicle to be equipped with velocity and direction sensors or each section of the highway to have a video camera. The main contributions of this paper can be summarized as follows:

1. Based on the NGSIM Interstate 80 Freeway Dataset, the vehicle lane-change behaviour characteristics were analyzed and selected, and the data, including non-lane-change, single lane-change, and sequential lane-change, was pre-processed and reconstructed.

2. Based on the analysis of the vehicle lane-change process, and considering the real-time requirements in the application of ICVs, the vehicle lane-change detection model based on K-Nearest Neighbor (KNN) is proposed and compared with extra tree (ET) and random forest (RF).

3. Through feature selection, the lateral speed, which is combined with speed and driving direction and is easy to be collected, is excavated as a feature for lane-change detection. The continuous data of 14 vehicles with frequent lane changes were tested and analyzed experimentally and performed well with accuracy between 89.57% and 100% on lane-change behaviour detection.

The rest of the paper is organized as follows. Section 2 explains the details of the dataset. Section 3 describes the methodology of lane-change behaviour detection, including feature selection and lane-change detection method. Section 4 presents the experiments and results of lane-change behaviour detection. Section 5 concludes this paper and discusses the future work.

2. Data Processing

In order to verify the lane-change detection method, NGSIM Interstate 80 Freeway Dataset initiated by the United States Department of Transportation (US DOT) Federal Highway Administration (FHWA), which is freely available at the NGSIM web site at http://ngsim.fhwa.dot.gov, is used and processed. The dataset contains 45 minutes, divided into three periods: 4:00 p.m. to 4:15 p.m.; 5:00 p.m. to 5:15 p.m.; and 5:15 p.m. to 5:30 p.m., which represent the buildup of congestion, the transition between uncongested and congested conditions, and full congestion during the peak period, respectively [25]. As shown in Figure 1, the six-lane study area with a length of 1650 feet is divided into seven sub-areas. In each sub-area, a video detector is installed on the high-rise building near the lane, and the traffic of the sub-area is photographed and recorded.

The original dataset contains many attributes, including some attributes that are not highly relevant to the lane-change detection. In order to establish a dataset suitable for vehicle lane-change detection, the attributes in the dataset that are not highly relevant to lane-change detection were deleted, Vehicle_ID, Lane_ID, V_Length, and V_Width remained the same as those in the original dataset, LX_m, LY_m, Vel_m/s, and Acc_m/s² changed the unit in the original dataset from feet to meters (1 foot = 0.3048 meters), the average lateral velocity of vehicle and instantaneous lateral acceleration of vehicle were, respectively, calculated by LX_m and Acc_m/s² and added to the dataset, and the lane-change behaviour of the vehicle was calculated by Lane_ID, forming a new dataset. In the Lane_changing attribute, 0 means to keep the current lane, 1 denotes a single lane change to the right, −1 stands for a single lane change to the left, 2 represents sequential lane change to the right, and −2 represents sequential lane change to the left. The composition of processed data is shown in Table 1.

The instantaneous lateral acceleration of vehicle Acc_X can be calculated as

\[
\text{Acc}_X = \text{Acc}_m/s^2 \times \sin \left( \arctan \left( \frac{\text{LX}_m - \text{LX}_{m-1}}{\text{LY}_m - \text{LY}_{m-1}} \right) \right),
\]

where Acc_Xt represents the value of Acc_X at time t, Acc_m/s² denotes the value of Acc_m/s² at time t, LX_m, and LX_m−1, respectively, mean the value of LX_m at times t and t−1, while LY_m and LY_m−1 stand for the value of LY_m at times t and t−1, respectively.

The average lateral velocity of vehicle Vel_X can be calculated as
where \( \text{Vel}_X_t \) represents the value of \( \text{Vel}_X \) at time \( t \), \( \text{LX}_m \) means the value of \( \text{LX}_m \) at time \( t \), \( \text{LX}_m(t-1) \) denotes the value of \( \text{LX}_m \) at time \( t-1 \), and \( t \) is the sampling period of the dataset, which is 0.1 seconds.

Current lane-change behaviour of vehicle \( \text{Lane}_{\text{changing}} \) can be calculated as

\[
\text{Lane}_{\text{changing}}_t = \text{Lane}_{\text{ID}}_t - \text{Lane}_{\text{ID}}_{t-1},
\]

where \( \text{Lane}_{\text{changing}}_t \) represents the value of \( \text{Lane}_{\text{changing}} \) at time \( t \), \( \text{Lane}_{\text{ID}}_t \) means the value of \( \text{Lane}_{\text{ID}} \) at time \( t \), and \( \text{Lane}_{\text{ID}}_{t-1} \) denotes the value of \( \text{Lane}_{\text{ID}} \) at time \( t-1 \).

\section{Methodology}

\subsection{Feature Selection}

In order to accurately detect vehicle lane changes, the relationship between various attributes and vehicle lane changes was analyzed. By analyzing the changing trends of various attributes when the vehicle changes lanes in the dataset, we found that the vehicle’s lateral velocity has the most obvious correlation with the lane-change behaviour. The relationship between lateral velocity and lane change is shown in Figure 2, from which we can see that when the vehicle changes lanes, there will be a very obvious change in lateral velocity.

To further analyze and verify the relationship between each attribute and the lane-change behaviour of the vehicle, machine learning models, including KNN [26], extra trees
and random forest, were performed on each attribute. A total of 27287 lane-changing data were extracted from the dataset. At the same time, in order to balance the number of samples of lane-changing (the moment when the vehicle changes lanes) and non-lane-changing (the moment when the vehicle does not change lanes) data, 27287 non-lane-changing data were selected to form a feature selection dataset. The result of feature selection is shown in Table 2, from which we can see that the detection accuracy with Vel_X as the feature is significantly higher than other features, which can reach more than 90%. In the detection using Vel_X, the accuracy of KNN, extratree, and random forest is 94.85%, 91.73%, and 92.31%, respectively; KNN has the highest accuracy.

In order to represent the contribution of each feature to the lane-change behaviour detection more intuitively, feature importance analysis, which can be applied to random forest and extra trees, was performed on the dataset. The Gini index was used to measure the feature importance, which can be defined as

$$G_m = \sum_{k=1}^{K} \sum_{k' 
eq k} p_{mk} p_{mk'} = 1 - \sum_{k=1}^{K} p^2_{mk},$$  \hspace{1cm} (4)$$

where $K$ means that there are $K$ categories and $p_{mk}$ denotes the proportion of category $k$ in node $m$. VIM is used to represent the variable importance measures. The importance of feature $X_i$ at node $m$, that is, the change in Gini index before and after the branch of node $m$, can be defined as

$$VIM^{(\text{Gini})}_{jm} = G_m - G_l - G_r,$$  \hspace{1cm} (5)$$

where $G_l$ and $G_r$ represent the Gini index of the left and right nodes after the $m$ branch. If the node where the feature $X_j$ appears in the decision tree $i$ is in the set $M$, then the importance of $X_j$ in the $i$-th tree is

$$VIM^{(\text{Gini})}_{ij} = \sum_{m \in M} VIM^{(\text{Gini})}_{jm}.$$  \hspace{1cm} (6)$$

Assuming there are $n$ trees, then

$$VIM^{(\text{Gini})}_j = \sum_{i=1}^{n} VIM^{(\text{Gini})}_{ij}.$$  \hspace{1cm} (7)$$

Finally, supposing there are $c$ features, all the obtained importance scores are normalized:

$$VIM_j = \frac{VIM_j}{\sum_{i=1}^{c} VIM_i}.$$  \hspace{1cm} (8)$$

The feature ranking based on feature importance is shown in Table 3, which illustrates that Vel_X has the highest feature importance scores in both random forest and extra tree and is significantly higher than the other six features. Therefore, Vel_X can be selected as a feature of lane-change detection for ICVs.

### 3.2. Lane-Change Detection

#### 3.2.1. Lane-Change Model

The data used in feature selection is discontinuous, so the learned features are relatively independent and have no relationship with the adjacent data. In the actual driving process, the data of vehicle lane-changing behaviour often only takes up a small part of the entire dataset. Therefore, in order to further analyze and establish a lane-change detection model during vehicle driving, we have selected continuous data from 14 vehicles with frequent lane changes for analysis, training, and testing.

Lane-change behaviour includes single lane change and sequential lane change. The single lane changes to the left and right are denoted as $-1$ and $1$, while the sequential lane
changes to the left and right are denoted as −2 and 2, respectively. The lateral velocity of single lane change intercepted from selected data is shown in Figure 3, from which we can see that there is a significant peak/valley when the vehicle changes lanes, and the threshold of peak/valley can be learned to determine if the vehicle is changing lanes. The lateral velocity of sequential lane change intercepted from selected data is shown in Figure 4; similar to single lane change, there is also a significant peak/valley when the vehicle changes lanes. Besides, there will be a continuous peak/valley or a larger peak/valley of lateral velocity in the sequential lane change, which also can be learned to determine if the vehicle is changing lanes sequentially.

### 3.2.2. Detection Model

KNN is a simple classification method that can perform effective classification in the absence of prior knowledge of data distribution, and the same are the ET and RF.

RF is an algorithm that integrates multiple trees through the idea of integrated learning and its basic unit is the decision tree [29]. The construction process of the random forest model is mainly divided into four steps:

1. Initialize training sets and categories
2. Calculate the Euclidean distance between the test set sample and the training set sample
3. Sort the training set samples in ascending order according to the Euclidean distance
4. Select the first K training samples with the smallest Euclidean distance and count their frequency in each category
5. The category with the highest return frequency, that is, the test set sample, belongs to this category

Table 4 shows the step of KNN algorithm, in which the list $I_k$ of its nearest neighbors is determined by calculating the similarity distance between the training object $(x, y) \in I$ and the test object $(x_t, y_t)$:}

$$Euclidean \ distance \ (a, b) = \sqrt{\sum_{i=1}^{n} (a_i - b_i)^2}.$$  \hspace{1cm} (9)

When KNN is used for classification, the output can be calculated as the class with the highest frequency from the K most similar instances. Each instance in essence votes for their class and the class with the most votes is taken as the prediction.

Class probabilities can be calculated as the normalized frequency of samples that belong to each class in the set of K most similar instances for a new data instance. For example, in a binary classification problem (class is 0 or 1),

$$p(\text{class} = 0) = \frac{\text{count(\text{class} = 0)}}{\text{count(\text{class} = 0)} + \text{count(\text{class} = 1)}}$$  \hspace{1cm} (10)

The KNN algorithm can be described as follows:

1. Initialize training sets and categories
2. Calculate the Euclidean distance between the test set sample and the training set sample
3. Sort the training set samples in ascending order according to the Euclidean distance
4. Select the first K training samples with the smallest Euclidean distance and count their frequency in each category
5. The category with the highest return frequency, that is, the test set sample, belongs to this category

Table 2: Precision result of feature selection.

| Selected features | KNN | Extra trees (%) | Random forest (%) |
|-------------------|-----|-----------------|------------------|
| L_X_m; L_Y_m      | 60.28 | 70.63           | 74.28            |
| V_Length; V_Width | 62.38 | 65.26           | 65.29            |
| Vel_m/s           | 49.14 | 50.26           | 50.67            |
| Acc_m/s²          | 43.28 | 43.62           | 43.52            |
| Vel_X             | 94.85 | 91.73           | 92.31            |

Table 3: Features ranked based on importance.

| Rank | Feature | Importance | Rank | Feature | Importance |
|------|---------|------------|------|---------|------------|
| 1    | Vel_X   | 0.712      | 1    | Vel_X   | 0.788      |
| 2    | Vel_m/s | 0.094      | 2    | Vel_m/s | 0.061      |
| 3    | Acc_m/s²| 0.052      | 3    | Acc_m/s²| 0.042      |
| 4    | L_X_m   | 0.044      | 4    | L_X_m   | 0.032      |
| 5    | L_Y_m   | 0.040      | 5    | L_Y_m   | 0.030      |
| 6    | V_Length| 0.031      | 6    | V_Length| 0.024      |
| 7    | V_Width | 0.027      | 7    | V_Width | 0.022      |
and the test object \( z = (\tilde{x}, \tilde{y}) \), where \( x \) represents the training object, \( y \) represents the class to which it belongs, and \( \tilde{x} \) and \( \tilde{y} \) represent the test object and the class to which it belongs.

4. **Experimental Results**

4.1. **Evaluation Indicators.** When performing machine learning, the confusion matrix of prediction results can be described as shown in Table 5.

The number of pairs of samples divided by the number of all samples is the accuracy (ACC), which can be defined as

\[
ACC = \frac{TP}{TP + FN + FP + FN}
\]

(11)

Generally, the higher the accuracy, the better the classifier. However, in the case of imbalance between positive and negative samples, the accuracy is a big flaw as an evaluation indicator. It is not scientific and comprehensive to evaluate a model based on accuracy alone.

To evaluate the performance of machine learning more scientifically and comprehensively, precision \((P)\), recall \((R)\), and \(F1\) score can be used.
4.2. Experiments and Analysis.

The precision, which represents the proportion of positive examples that are actually classified as positive examples, can be defined as

\[ P = \frac{TP}{TP + FP} \]  

(12)

The recall, which measures how many positive examples are classified as positive examples, can be defined as

\[ R = \frac{TP}{TP + FN} \]  

(13)

\( P \) and \( R \) indicators sometimes conflict, so they need to be considered comprehensively. The most common method is \( F \)-Measure, which is the weighted harmonic average of \( P \) and \( R \), and can be defined as

\[ F = \frac{\alpha^2 + 1)P \cdot R}{\alpha^2 (P + R)} \]  

(14)

When the parameter \( \alpha = 1 \), it is \( F1 \):

\[ F1 = \frac{2 \cdot P \cdot R}{P + R} \]  

(15)

\( F1 \) combines the results of \( P \) and \( R \). When \( F1 \) is higher, it can indicate that the model is more effective.

In addition, receiver operating characteristic (ROC) curve, in which the abscissa is False Positive Rate (FPR) and the ordinate is True Positive Rate (TPR), is also an important evaluation indicator. The definition of TPR is the same with the ordinate is True Positive Rate (TPR), is also an important evaluation indicator. The definition of TPR is the same with\[ \text{TPR} = \frac{TP}{TP + FN} \]

4.2. Experiments and Analysis. The model is built on vscode using python and uses the scikit-learn framework. The experiments were performed on a server with a single-core CPU, 2.6 GHz, 2G memory, and Ubuntu 18.04.

KNN was performed on the selected dataset first. In order to choose the most appropriate number of neighbors, we trained and tested different numbers of neighbors on the dataset consisting of the data of all 14 vehicles and obtained their accuracy, respectively. The result of varying number of KNN neighbors is shown in Figure 5, from which we can see that the accuracy of the training set decreases as the number of neighbors increases, and at the same time the accuracy of the test set increases as the number of neighbors increases.

When the number of neighbors increases to 9, the accuracy of both the training and test sets remains stable. Therefore, 9 is appropriate to be determined as the number of neighbors.

After determining the number of neighbors, KNN model was designed and performed on the dataset consisting of the data of all 14 vehicles, compared with ET and RF. The ROC of designed models including KNN, ET, and RF is shown in Figure 6 and the AUC values of the three models are shown in Table 6. Obviously, from the ROC curves, the performance of KNN is better than RF and obviously better than ET. It can be seen more clearly in Table 6 that the AUC value of KNN is 97.73%, while the AUC values of ET and RF are 92.55% and 96.69%, respectively, showing that the performance of KNN is the best in these three models.

After experimental testing on the dataset of all 14 vehicles, in order to analyze the effect of continuous lane-change detection of vehicles in real scenes, the designed KNN model was used to perform experiments on the respective datasets of the 14 vehicles and compared with ET and RF.

The detailed sample sizes of lane-change behaviour on 14 selected vehicles are shown in Table 7. In the dataset, we can see that, in the real scene, the number of lane-keep samples is generally larger than the number of left and right lane-change samples. Among the 14 vehicles, only the total number of the lane changes to left (LCL) and the lane changes to right (LCR) of the vehicle numbered 2791 is greater than the number of lane-keep (LK), and the number of each item is still less than the number of LK. Besides, there are continuous lane changes in vehicles numbered 2795 and 2825. A single lane change to the left and right is recorded as LCL-1 and LCR-1, while a continuous lane change to the left and right is recorded as LCL-2 and LCR-2, respectively.

The dataset combining all 14 selected vehicles is divided into training and testing sets according to a ratio of 0.75 to 0.25. The confusion matrix of the detection results is shown in Figure 7, which illustrates that the KNN model has the highest detection accuracy, followed by RF, and ET has the lowest detection accuracy. In addition, detection errors mainly occur between the non-lane-change and the single lane-change behaviour, the probability of false detection between sequential lane-change behaviour and other behaviours is small, and the probability of false detection between left lane-change and right lane-change behaviour is also small. It is worth noting that, in the KNN and RF models, there is no misdetection between the left lane-change, right lane-change, and non-lane-change behaviour.

The experimental results of lane-change detection performed on 14 selected vehicles are shown in Table 8, in which the evaluation indicator mACC denotes mean accuracy of the detection model on all lane-change behaviours. From the experimental results, KNN performed best in the lane-change detection results of all 14 vehicles, while in the lane-change detection results of 14 vehicles, ET performed better than RF in 4 vehicles, RF performed better than ET in 7 vehicles, and ET and RF performed the same in the remaining 3 vehicles. KNN’s lane-change detection accuracy ranges from 89.57% to 100%, ET's
### Table 6: AUC values of designed models.

|        | KNN (%) | ET (%) | RF (%) |
|--------|---------|--------|--------|
| AUC values | 97.73   | 92.55  | 96.69  |

### Table 7: Sample size of lane-change behaviour on selected vehicles.

| ID  | Behaviour | Sample size |
|-----|-----------|-------------|
| 3365| LCL       | 131         |
|     | LK        | 444         |
|     | LCR       | 133         |
| 3362| LCL       | 79          |
|     | LK        | 299         |
|     | LCR       | 79          |
| 2826| LCL       | 39          |
|     | LK        | 123         |
|     | LCR       | 39          |
| 2804| LCL       | 45          |
|     | LK        | 267         |
|     | LCR       | 45          |
| 2795| LCL-2     | 36          |
|     | LCL-1     | 46          |
| 2782| LCL       | 152         |
|     | LK        | 935         |
|     | LCR       | 151         |
| 2778| LCL       | 91          |
|     | LK        | 620         |
|     | LCR       | 91          |
| 3363| LCL       | 334         |
|     | LK        | 127         |
|     | LCR       | 127         |
| 3063| LCL       | 23          |
|     | LK        | 99          |
|     | LCR       | 22          |
| 2791| LCL       | 207         |
|     | LK        | 379         |
|     | LCR       | 206         |
| 2800| LCL       | 100         |
|     | LK        | 300         |
|     | LCR       | 99          |
| 2825| LCL-2     | 12          |
|     | LCL-1     | 13          |
| 2779| LCL       | 134         |
|     | LK        | 526         |
|     | LCR       | 135         |
| 2774| LCL       | 100         |
|     | LK        | 379         |
|     | LCR       | 102         |
Figure 7: The confusion matrix of detection results: (a) ET, (b) RF, and (c) KNN.
Table 8: Experimental results of lane-change detection on selected vehicles.

| ID   | Model | Behaviour | P (%) | R (%) | F1 (%) | mACC (%) |
|------|-------|-----------|-------|-------|--------|----------|
| 3365 | ET    | LCL       | 91    | 94    | 93     | 97.18    |
|      |       | LK        | 98    | 97    | 98     |          |
|      |       | LCR       | 100   | 100   | 100    |          |
|      |       | LCL       | 91    | 94    | 93     |          |
| 3365 | RF    | LK        | 97    | 97    | 97     | 96.61    |
|      |       | LCR       | 100   | 97    | 99     |          |
|      |       | LCL       | 97    | 97    | 97     |          |
|      | KNN   | LK        | 99    | 99    | 99     | 98.87    |
|      |       | LCR       | 100   | 100   | 100    |          |
| 3362 | ET    | LCL       | 68    | 89    | 77     | 84.35    |
|      |       | LK        | 89    | 84    | 86     |          |
|      |       | LCR       | 88    | 82    | 85     |          |
|      |       | LCR       | 71    | 89    | 79     |          |
| 3362 | RF    | LK        | 91    | 85    | 88     | 86.09    |
|      |       | LCR       | 89    | 86    | 87     |          |
|      |       | LCL       | 76    | 100   | 86     |          |
|      | KNN   | LK        | 97    | 85    | 91     | 89.57    |
|      |       | LCR       | 87    | 93    | 90     |          |
| 2826 | ET    | LCL       | 89    | 89    | 89     | 94.12    |
|      |       | LK        | 94    | 97    | 96     |          |
|      |       | LCR       | 100   | 86    | 92     |          |
|      |       | LCR       | 80    | 89    | 84     |          |
| 2826 | RF    | LK        | 92    | 94    | 93     | 90.20    |
|      |       | LCR       | 100   | 71    | 83     |          |
|      |       | LCL       | 89    | 89    | 89     |          |
| 2826 | KNN   | LK        | 97    | 97    | 97     | 96.08    |
|      |       | LCR       | 100   | 100   | 100    |          |
| 2804 | ET    | LCL       | 73    | 80    | 76     | 93.33    |
|      |       | LK        | 97    | 94    | 96     |          |
|      |       | LCR       | 90    | 100   | 95     |          |
|      |       | LCR       | 89    | 80    | 84     |          |
| 2804 | RF    | LK        | 97    | 96    | 96     | 94.44    |
|      |       | LCR       | 82    | 100   | 90     |          |
|      |       | LCL       | 90    | 90    | 90     |          |
| 2804 | KNN   | LK        | 97    | 97    | 97     | 95.55    |
|      |       | LCR       | 89    | 89    | 89     |          |
| 2795 | ET    | LCL-2     | 91    | 100   | 95     |          |
|      |       | LCL-1     | 67    | 77    | 71     |          |
|      |       | LK        | 99    | 95    | 97     | 94.36    |
|      |       | LCR-1     | 81    | 100   | 90     |          |
|      |       | LCR-2     | 100   | 83    | 91     |          |
|      |       | LCL-2     | 91    | 100   | 95     |          |
|      |       | LCL-1     | 91    | 77    | 83     |          |
| 2795 | RF    | LK        | 99    | 98    | 98     | 96.41    |
|      |       | LCR-1     | 81    | 100   | 90     |          |
|      |       | LCR-2     | 100   | 83    | 91     |          |
|      |       | LCL-2     | 100   | 100   | 100    |          |
|      |       | LCL-1     | 100   | 77    | 87     |          |
| 2795 | KNN   | LK        | 98    | 100   | 99     | 98.46    |
|      |       | LCR-1     | 100   | 100   | 100    |          |
|      |       | LCR-2     | 100   | 100   | 100    |          |
| ID  | Model | Behaviour | P (%) | R (%) | F1 (%) | mACC (%) |
|-----|-------|-----------|-------|-------|--------|----------|
|     | ET    | LCL       | 100   | 98    | 99     | 99.03    |
|     |       | LK        | 99    | 100   | 99     | 99.03    |
|     |       | LCR       | 100   | 95    | 97     | 97.00    |
|     |       | LCL       | 100   | 98    | 99     | 99.03    |
| 2782| RF    | LK        | 98    | 100   | 99     | 98.71    |
|     |       | LCR       | 100   | 92    | 96     | 96.00    |
|     |       | LCL       | 100   | 98    | 99     | 99.03    |
|     | KNN   | LK        | 100   | 100   | 100    | 99.68    |
|     |       | LCR       | 100   | 100   | 100    | 99.68    |
|     |       | LCL       | 96    | 100   | 98     | 98.01    |
|     |       | LK        | 100   | 97    | 99     | 98.01    |
|     |       | LCR       | 88    | 100   | 94     | 94.00    |
|     |       | LCL       | 96    | 100   | 98     | 98.01    |
|     | KNN   | LK        | 99    | 100   | 99     | 99.00    |
|     |       | LCR       | 100   | 91    | 95     | 95.00    |
|     |       | LCL       | 87    | 77    | 82     | 82.00    |
|     | ET    | LK        | 87    | 90    | 89     | 87.07    |
|     |       | LCR       | 86    | 89    | 88     | 88.00    |
|     |       | LCL       | 90    | 74    | 81     | 81.00    |
| 2778| RF    | LK        | 86    | 90    | 88     | 86.39    |
|     |       | LCR       | 83    | 89    | 86     | 86.00    |
|     |       | LCL       | 96    | 77    | 86     | 86.00    |
|     | KNN   | LK        | 90    | 95    | 92     | 91.16    |
|     |       | LCR       | 90    | 96    | 93     | 93.00    |
|     |       | LCL       | 50    | 33    | 40     | 40.00    |
|     |       | LK        | 84    | 96    | 90     | 90.00    |
|     |       | LCR       | 100   | 40    | 57     | 57.00    |
|     |       | LCL       | 50    | 33    | 40     | 40.00    |
| 3363| RF    | LK        | 87    | 96    | 92     | 86.11    |
|     |       | LCR       | 100   | 60    | 75     | 75.00    |
|     |       | LCL       | 100   | 67    | 80     | 80.00    |
|     | KNN   | LK        | 97    | 100   | 98     | 97.22    |
|     |       | LCR       | 100   | 100   | 100    | 100.00   |
|     |       | LCL       | 100   | 77    | 87     | 87.00    |
|     |       | LK        | 82    | 98    | 89     | 88.89    |
|     |       | LCR       | 95    | 87    | 91     | 91.00    |
|     |       | LCL       | 100   | 84    | 91     | 91.00    |
| 3063| RF    | LK        | 84    | 97    | 90     | 89.90    |
|     |       | LCR       | 93    | 84    | 88     | 88.00    |
|     |       | LCL       | 96    | 85    | 91     | 91.00    |
|     | KNN   | LK        | 89    | 96    | 92     | 92.42    |
|     |       | LCR       | 96    | 96    | 96     | 96.00    |
|     |       | LCL       | 88    | 93    | 90     | 90.00    |
|     |       | LK        | 94    | 85    | 90     | 90.00    |
|     |       | LCR       | 72    | 90    | 80     | 80.00    |
|     |       | LCL       | 91    | 97    | 94     | 94.00    |
| 2791| RF    | LK        | 96    | 91    | 93     | 92.00    |
|     |       | LCR       | 82    | 90    | 86     | 86.00    |
|     |       | LCL       | 97    | 97    | 97     | 97.00    |
|     | KNN   | LK        | 97    | 95    | 96     | 95.20    |
|     |       | LCR       | 86    | 95    | 90     | 90.00    |
lane-change detection accuracy ranges from 83.33% to 99.50%, while RF’s lane-change detection accuracy ranges from 86.09% to 99.50%. Besides, combined with the sample sizes, in the vehicles with the numbers of 3063 and 2825, the number of samples of the lane-change left and the lane-change right is small, and the detection results are relatively poor, indicating that too few samples will affect the accuracy of the lane-change detection.

5. Conclusions

This paper proposed a lane-change detection method for intelligent and connected vehicles. Based on the feature selection of vehicle lane-change behaviour, the detection model based on machine learning was designed, and the effect verification and comparison were performed on the selected dataset. The dataset based on NGSIM Interstate 80 Freeway Dataset was processed for lane-change detection first. After that, feature selection for lane-change detection was performed on the processed dataset, and the lateral velocity was selected as the feature for lane-change detection. Then, the lane-change model was analyzed based on the real data in the processed dataset and the detection model was designed. Finally, the number of KNN neighbors was determined based on experiment, and the performance of KNN, ET, and RF was analyzed by the evaluation indicators. From the experimental results, the designed KNN model performed best in all datasets of the selected 14 vehicles, with detection accuracy ranging from 89.57% to 100%, indicating that it can well complete the task of lane-change behaviour detection for ICVs.

As for future work, the lane-changing scene can be extended by the measured data from the vehicle sensors to establish a more widely adaptable dataset, and the detection model can be further optimized and then implemented on embedded hardware to achieve a lane-change real-time detection system for ICVs.

Data Availability

The related data are available online at https://github.com/WHUT-DLY/Processed-data-based-on-the-Next-Generation-Simulation-NGSIM-Interstate-80-Freeway-Dataset.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.
Acknowledgments

The research work was supported in part by the National Key R&D Program of China under Grant 2018YFB0105205, in part by Major Technological Innovation Project of Hubei Province under Grant 2019AAA025, and in part by the Fundamental Research Funds for the Central Universities (WUT: 2019-JL-023).

References

[1] K. Q. Li, Y. F. Dai, S. B. Li, and M. Y. Bian, “State-of-the-art and technical trends of intelligent and connected vehicles,” Journal of Automotive Safety and Energy, vol. 8, no. 1, pp. 1–14, 2017.
[2] S. A. E. China–, “Technology Roadmap for Energy Saving and New Energy Vehicles,” Mechanical Industry Press, Beijing, China, 2016, in Chinese.
[3] R. Hanada, T. Kubo, K. Ikeda et al., “Modeling and prediction of driving behaviors using a nonparametric Bayesian method with AR models,” IEEE Transactions on Intelligent Vehicles, vol. 1, no. 2, pp. 131–138, 2016.
[4] Q. Deng and D. Soffker, “Improved driving behaviours prediction based on fuzzy logic-hidden Markov model (FL-HMM),” in Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), pp. 2003–2008, Suzhou, China, June 2018.
[5] E. Yurtsever, S. Yamazaki, C. Miyajima et al., “Integrating driving behavior and traffic context through signal symbolization for data reduction and risky lane change detection,” IEEE Transactions on Intelligent Vehicles, vol. 3, no. 3, pp. 242–253, 2018.
[6] Y. Zhang, J. Li, Y. Guo, C. Xu, J. Bao, and Y. Song, “Vehicle driving behavior recognition based on multi-view convolutional neural network with joint data augmentation,” IEEE Transactions on Vehicular Technology, vol. 68, no. 5, pp. 4223–4234, 2019.
[7] Z. S. Wei, C. Wang, P. Hao, and M. J. Barth, “Vision-based lane-changing behaviour detection using deep residual neural network,” in Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pp. 3108–3113, Auckland, New Zealand, October 2019.
[8] X. Y. Xu, J. D. Yu, Y. M. Zhu, Z. C. Wu, J. D. Li, and M. L. Li, “Leveraging smartphones for vehicle lane-level localization on highways,” IEEE Transactions on Mobile Computing, vol. 17, no. 8, pp. 1894–1907, 2018.
[9] T. Rehder, A. Koenig, M. Goehl, L. Louis, and D. Schramm, “Lane change intention awareness for assisted and automated driving on highways,” IEEE Transactions on Intelligent Vehicles, vol. 4, no. 2, pp. 265–276, 2019.
[10] N. Motamedidehkordi, S. Amini, S. Hoffmann, F. Busch, and M. R. Fitriyanti, “Modeling tactical lane-change behaviour for automated vehicles: a supervised machine learning approach,” in Proceedings of the 2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS), pp. 268–273, Naples, Italy, June 2017.
[11] D. Augustin, M. Hofmann, and U. Konigorski, “Prediction of highway lane changes based on prototype trajectories,” Forschung im Ingenieurwesen, vol. 83, no. 2, pp. 149–161, 2019.
[12] S. Seelam, M. Arpan, V. Praveen, and K. G. Naga, “Simulation of traffic flow to analyze lane changes on multi-lane highways under non-lane discipline,” Periodica Polytechnica: Transportation Engineering, vol. 48, no. 2, pp. 109–116, 2018.
[13] T. Wang, L. J. Xu, G. J. Chen, and W. Zhao, “A guidance method for lane change detection at signalized intersections in connected vehicle environment,” in Proceedings of the 2019 5th International Conference on Transportation Information and Safety, pp. 32–38, Liverpool, UK, July 2019.
[14] Q. Gao, J. Zhang, Z. Sheng, and L. Dong, “A lane-changing BML model considering the influence of both lane information and turn signals,” Luxue Xuebao/Chinese Journal of Theoretical and Applied Mechanics, vol. 52, no. 1, pp. 283–291, 2020.
[15] Y. Xing, J. Wang, W. Liu, L. Sun, and F. Chong, “Study on the characteristics of vehicle lane-changing in the intersection,” Advances in Intelligent Systems and Computing, vol. 890, pp. 371–381, 2019.
[16] F. L. Wei, Z. Y. Wang, and J. Liu, “Exploring factors contributing to lane changes during left turns on quadruple left-turn lanes at signalized intersections,” Advances in Mechanical Engineering, vol. 9, no. 5, pp. 171–179, 2018.
[17] Z. Ouyang, J. Niu, and M. Guizani, “Improved vehicle steering pattern recognition by using selected sensor data,” IEEE Transactions on Mobile Computing, vol. 17, no. 6, pp. 1383–1396, 2018.
[18] L. X. Li and P. H. Li, “Analysis of drivers’ steering behaviour for lane change prediction,” in Proceedings of the 2019 11th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), pp. 71–75, Hangzhou, China, August 2019.
[19] D.-F. Xie, Z.-Z. Fang, B. Jia, and Z. He, “A data-driven lane-changing model based on deep learning,” Transportation Research Part C: Emerging Technologies, vol. 106, pp. 41–60, 2019.
[20] Y. Xing, C. Lv, H. J. Wang, D. P. Cao, and E. Velenis, “An ensemble deep learning approach for driver lane change intention inference,” Transportation Research: Part C, vol. 115, 2020.
[21] J. Gao, Y. L. Murphy, J. G. Yi, and H. H. Zhu, “A data-driven lane-changing behavior detection system based on sequence learning,” Transportmetrica B: Transport Dynamics, 2020.
[22] X. Zhang, J. Sun, X. Qi, and J. Sun, “Simultaneous modeling of car-following and lane-changing behaviors using deep learning,” Transportation Research Part C: Emerging Technologies, vol. 104, pp. 287–304, 2019.
[23] W. Zhao, L. J. Xu, B. Ran, and J. Z. Wang, “Dangerous lane-change detecting model on highway based on deep learning DBN algorithm,” Journal of Southeast University (Natural Science Edition), vol. 47, no. 4, pp. 832–838, 2017.
[24] T. Dang and F. Dai, “Vehicle lane change model based on improved bayesian network structure learning,” International Journal of Intelligent Technologies and Applied Statistics, vol. 11, no. 4, pp. 255–270, 2018.
[25] Federal Highway Administration (FHWA), “Next Generation Simulation (NGSIM) Interstate 80 Freeway Dataset,” U.S. Department of Transportation Intelligent Transportation Systems Joint Program Office (JPO), Washington, D.C., USA, 2016.
[26] Z. Yu, H. Chen, J. Liuxs, J. You, H. Leung, and G. Han, “Hybrid,” IEEE Transactions on Cybernetics, vol. 46, no. 6, pp. 1263–1275, 2016.
[27] V. John, Z. Liu, C. Z. Guo, S. Mita, and K. Kidono, “Real-time lane estimation using deep features and extra trees regression,” Image and Video Technology, vol. 9431, pp. 721–733, 2016.
[28] G. Robin, J. M. Poggi, T. M. Christine, and V. V. Nathalie, “Random forests for big data,” Big Data Research, vol. 9, pp. 28–46, 2017.
[29] L. Breiman, “Random forests,” Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
[30] P. Geurts, D. Ernst, and L. Wehenkel, “Extremely randomized trees,” Machine Learning, vol. 63, no. 1, pp. 3–42, 2006.