Research Article

Extracting Overtaking Segments by Unsupervised Clustering and Predicting Nonmotorized Vehicle’s Trajectory

Ailing Yin, Xiaohong Chen, and Lishengsa Yue

The Key Laboratory of Road and Traffic Engineering, Ministry of Education, College of Transportation Engineering, Tongji University, 4800 Cao’an Road, Shanghai 201804, China

Correspondence should be addressed to Xiaohong Chen; chenxh@tongji.edu.cn

Received 12 January 2022; Revised 9 March 2022; Accepted 18 March 2022; Published 13 April 2022

Copyright © 2022 Ailing Yin 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.

Interpretation of flexible cycling behavior has always been a tough task. It is meaningful to understand the overtaking behavior of cyclists for its threats to safety and its high frequency on shared roads. Advanced unsupervised nonparametric clustering methods are compared to distinguish the overtaking segments from the whole trajectory based on the cycling characteristics of non-motorized two-wheelers, while the hierarchical Dirichlet process hidden Markov model (HDPHMM) outperforms the mixture model via the Dirichlet process (DP mixture model) and topic model via the hierarchical Dirichlet process (HDP topic model). HDPHMM clusters each record into different states and results in more continuous segments. Based on marked vehicle types, which state of clustering model represents the overtaking condition is deduced. The overtaking segments resulted from HDPHMM show the highest homogeneity in cycling features with actual overtaking behavior. Another practical task is to predict the overtaking trajectory and respond to overtaking behavior in advance. Comparing original trajectory and subdivided trajectory, it is found that training model with grouped data, which have homogeneous features, can improve prediction accuracy. With enough trainable samples, CNN + LSTM hybrid structure can achieve trajectory prediction with a mean absolute error of 3 cm. The segmentation produces trajectory segments with similar characteristics. The model is trained with overtaking trajectory segments. With tens of times less trainable data, the prediction on overtaking trajectory still keeps a mean absolute error of about 5 cm. Subdividing trajectory into segments with homogeneous features can improve the prediction accuracy and reduce the requirement of trainable data volume.

1. Introduction Background

Bicycles have a long story of being part of the urban transport. They once gave way to the development of motor vehicles. Emphasis on environmental protection and energy conservation, as well as the birth of the sharing economy, has led to the renaissance of bicycles. At the end of the 1990s, the emergence of electric bicycles has made two-wheeled vehicles more widely applicable, providing personalized travel services for middle- and low-income groups of people, popular in developing countries. In Chinese cities, electric bicycles are commonly used in logistics and food take-out services. According to the fifth comprehensive traffic survey in Shanghai, the average daily traffic volume of bicycles is 1.55 million and that of electric bicycles is 4.41 million. Nonmotorized vehicles are essential when discussing today’s urban traffic.

Light motorcycle is a motorcycle whose maximum designed speed is less than 50 km/h. Electric two-wheelers weighing more than 55 kilograms and traveling at speed higher than 25 km/h are classified as electric scooters (e-scooters), yet there is no specific speed management regulation. Electric bicycles (e-bicycles), regular bicycles (r-bicycles), and e-scooters share road space under the present conditions of urban transport. In this study, e-bicycles and e-scooters are considered as mopeds. At present, r-bicycles and mopeds share nonmotor lane but with a huge speed difference due to differences in power supply. Speed differences make overtaking more frequent and the urban driving environment more complicated.
Overtaking is one of the behaviors that influence riding comfort and stress the most and also brings great safety threats to road users, especially to r-bicycles. When there is no physical separation from motor vehicles, mopeds frequently take illegal lane changing to overtake slower objects, showing interference to motor vehicles. Overtaking behavior has always been a special and important content in non-motorized vehicle-related research. Identifying the overtaking segments from trajectory data provides source for the study of overtaking behavior and for the study of comparison between overtaking and other cycling behavior. Because overtaking interferes with traffic flow, it is also necessary to predict and respond to this operation in advance, but due to the flexibility of mopeds and r-bicycles, their trajectories are more difficult to predict. Facing the rising demand for traffic safety, with the help of intelligent transportation to avoid collision through communication, it is urgent to study and predict the overtaking behavior.

In this study, the overtaking trajectory segment is distinguished by unsupervised methods, and the differences between overtaking and other cycling behaviors are analyzed. Furthermore, based on the analysis of overtaking features, the high-precision prediction of overtaking trajectory is achieved. These findings demonstrate the effectiveness of unsupervised clustering in distinguishing cycling behavior. Based on segmented trajectories, major and special cycling patterns can be mined and applied to behavior modeling, risk behavior assessment, and avoidance. The prediction results prove the advantages of segmentation on trajectory prediction, indicating that trajectory prediction based on segmentation can improve the accuracy and reduce the deep learning’s requirement on training data.

2. Literature Review

Study concerning traffic flow can be traced back in the 1930s, to describe the road traffic, and the fundamental model relating volume and speed was conducted. A lot of work has been done to establish flow models and describe the riding features of nonmotorized vehicles. Models revealing the relations between volume, speed, and density have been established [1–3]. The parameter relations of nonmotorized traffic flow are different from that of motor vehicle. Nonmotorized flow has greater compressibility and congestion when speed declines to zero will hardly occur on road section. To study the impact of mopeds on traffic flow, conversion coefficients for mopeds have been proposed [4]. To describe the riding features at the microlevel, most studies focused on the description of speed and distance. With data collected in Hanoi, Vietnam, the characteristics of overtaking and paired riding on dedicated lanes and undivided roads have been described in detail, including speed, speed difference, lateral distance, overtaking lasting distance, and the speed difference threshold for pair riding [5]. With video data collected in Shanghai, China, overtaking maneuvers have been described with speed, acceleration, headway distance, and overtaking distance [6]. Another study in Denver, Colorado, described the distribution of speed, lateral spacing, and headway spacing and their correlations during overtaking and pair riding [7] but only focused on r-bicycles. A comparative study in Kunming, China, has found that the mean operating speed of e-bicycles is much faster than that of r-bicycles, which is 21.86 km/h and is 7.05 km/h faster or 47.6% higher [8]. From the perspective of cyclists, equipping testers with a laser rangefinder, GPS tracker, and camera, it is found that comprehensive factors such as lateral clearance, vehicle type, and speed are important factors that affect risk perception [9].

These studies analyzed nonmotorized vehicle riding parameters and relations from a relatively basic level. Speed, acceleration, and distance are the main factors describing riding features. Describing cycling by parameters has formed an initial understanding of cycling interaction. These parameters can be synthesized to construct indexes to measure cycling safety and cycling stress [10]. Overtaking and pair riding are the main maneuvers describing riding interactions. The cycling interaction needs to be further understood from the whole process of parameter variation, which is to achieve the purpose of trajectory generation and prediction.

Researchers have tried to predict the intention of cyclists [11–13] and generate trajectories [13–15]. Model-based studies rely largely on analytical kinematic models and provide mathematical formulas to support simulation. However, information processing in the human brain does not follow certain formulas, so deep learning is proposed to imitate human learning, which learns by examples. With the trained model, the prediction results can be instantly obtained with corresponding inputs, which is conducive to real-time applications. Researchers have tried to apply deep learning in the transportation field.

On the one hand, researchers apply deep learning to classify and predict intentions, for example, CNN to classify cycling maneuver [16], LSTM to predict crossing intentions and real-time crash risk at signalized intersections [17], and RNN to predict pedestrian intentions [18]. On the other hand, deep learning is applied to generate and predict trajectory. Using natural driving data from the extensively studied NGSIM dataset, a LSTM-based neural network is proposed to predict the behavior of the target vehicle on highways with local information of 9 cars around the target [19]. Validated by simulation results, it is found that combining CNN and LSTM can achieve better performance when predicting the steering operation [20]. The efficiency and accuracy of CNN-LSTM hybrid model are validated in other studies, such as trajectory prediction with speed and clearance information of vehicles around the target [21] and high-precision 4D trajectory prediction [22]. RNN shows advantages in dealing with trajectory data, a type of time series.

Deep learning shows an advantage in predicting trajectory of motor vehicles [19–21] and has been applied in route planning and collision prevention for automated vehicles. However, these trajectory prediction algorithms for motor vehicles are not applicable to nonmotorized vehicles because they are far more flexible and have no strict bicycle lanes to limit their trajectories. This makes their movements more arbitrary. On shared roads, it is necessary to consider the movements of nonmotorized vehicles [23, 24]. For
autonomous vehicles, a deeper understanding of the riding characteristics of these flexible two-wheelers is important, to plan trajectory and avoid collision. Current trajectory planning of autonomous vehicles rarely considers nonmotorized vehicles [25], and the assumption of unaltered speed does not conform to the real situation.

Unsupervised learning and nonparametric approaches attract researchers’ attention in behavior analysis and multi-micro-feature interpretation. The idea of cluster is used to integrate features. Manually and empirically classifying the trajectories into several simple driving scenarios according to specific maneuvers is time-consuming and subjective. The idea of machine learning is introduced to automatically identify the differences. Li et al. have done a series of studies on motor vehicle trajectory segmentation. In 2018, combining an autoencoder with k-means, driving encounters are clustered into classes [26]. Another similar study compares autoencoders, dynamic time warping (DTW), and normalized Euclidean distance (NED) in their performance in feature extraction [27]. Still, k-means clustering is then applied. In 2018, primitives are introduced that can be viewed as basic building blocks of driver behavior or principal compositions of the entire traffic [28]. Primitives are then applied in driving style analysis [29]. The results show that primitives have an apparent statistical difference in relative distance, relative velocity, and acceleration. In 2020, the research group identifies 20 kinds of traffic primitives presenting essential components of driving encounters [30]. Mohammed et al. use multivariate finite mixture model-based clustering to understand cyclists’ behavior during various interactions [31] and find that the following can be grouped into constrained and unconstrained states, and overtaking can be grouped into initiation, merging, and post-overtaking states. Segmentation, classifying, or clustering can differentiate the states in the whole database based on their hidden features.

The research on nonmotorized vehicle traffic flow parameters reveals the basic law of riding and constructs several description indexes of riding features. Deep learning imitates human instance learning process and attempts to understand and predict the driving intention and movements. Combining two, there are studies that try to understand driving behavior. Unsupervised clustering disassembles the driving process into minimum components, to study the basic features of driving behavior. These components can be further pieced together to generate trajectory. The movement of nonmotorized vehicle is more flexible, which makes it more difficult to describe and predict, but there is rarely a study that tries to combine their cycling features with trajectory prediction to achieve better performance.

Based on trajectory data, this study applies unsupervised clustering to split cycling and proposes a semiautomatic extraction method to get overtaking segments. It provides a method to reduce human work for big data collection in cycling behavior research. Then, overtaking is predicted and evidence shows that subdividing trajectory can improve prediction accuracy.

3. Methods

The flow chart of data process and analysis is shown in Figure 1. This section introduces the data process methods.

3.1. Feature Scaling. Feature scaling is to normalize the range of independent variables or features. It is an important step during the data preprocessing to reduce the scale influence of different variables on results. Feature scaling is needed to bring every feature equal importance, especially when the measurement units are different. In terms of efficiency, if the values of features are closer to each other, there are chances for the algorithm to get trained better and faster.

There are some common techniques of feature scaling, such as min-max scaler and standard scaler. The min-max scaler transforms features by scaling each value to a given range.

\[
x_{\text{new}} = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}
\]

The standard scaler assumes that data are normally distributed within each feature, and the scaler transforms the values to center around 0 with a standard deviation of 1.

\[
x_{\text{new}} = \frac{x - \mu}{\sigma}
\]

In this study, the trajectory points and other features related to space location such as distance (distance with direction, thus has a negative value) are scaled to [−1, 1] by min-max scaler, because there is no Gaussian relation between location points. Other features such as speed, acceleration, relative speed (or the speed difference between vehicles), and relative acceleration are scaled by standard scaler, for their Gaussian-like distribution.

3.2. Clustering. Clustering can help data mining and group characteristics to extract prevailing patterns. The overtaking trajectory is different from that of normal cycling. Findings [1–9] show that riding speed and spacing are important in differentiating overtaking from cycling. This study exploits the unsupervised clustering method to segment cycling based on descriptive cycling parameters and extracts the overtaking segments.

A primary goal in machine learning is to infer interpretable clusters or segmentations from complex datasets. The simple mixture model was once popular but could not deal with universal exchangeability to capture spatial, temporal, hierarchical, or relational structure. Combining structured models with the Bayesian nonparametric priors (BNPs) such as the Dirichlet process (DP) enables learning the number of clusters from data, instead of fixing the number of clusters as parametric models do [32]. The Bayesian nonparametric methods are based on a statistical framework that contains in principle an infinite number of parameters.

There are three unsupervised clustering compared in this study. (1) The Dirichlet process mixture model (DP mixture model) allows to build a mixture model when the number of
distinct clusters in the geometric structure of data is unknown; in other words, the number of clusters is allowed to grow as more data are observed. (2) A topic model is a statistical model for discovering the topics that appear in a collection of documents. It is initially a machine learning and natural language process. Topic model refers to statistical algorithms for discovering the latent semantic structures of a text body, but topic models have also been used to detect instructive structures in data such as genetic information and computer vision. (3) The hidden Markov model (HMM) is a statistical model that uses a Markov process that contains hidden and unknown parameters. The observed parameters are used to identify the hidden parameters, and these parameters are then used for further analysis. HMM is a type of Markov chain. The state cannot be directly observed but can be identified by observing the vector series. HMM links the observed events, which have no one-to-one correspondent state with states but link them through the probability distribution. It is actually a doubly stochastic process, which includes a Markov chain as the basic stochastic process and a stochastic process describing the statistical correspondence between states and observed values and state transitions. HMM has been used in research on time-series data.

There is an open package, BNPY, which provides code for training popular clustering models on large datasets. The package introduces merge, delete, and birth moves [32]. A compact set of clusters benefits the interpretability and improves algorithm speed. BNPY introduces two nonlocal proposals to remove clusters, the pairwise merges to eliminate redundancy and deletes to remove unnecessary clusters. To avoid poor initialization, the BNPY developed data-informed birth moves that can add many clusters at once, even if no single batch alone contains enough evidence for the cluster because escaping poor initialization requires adding useful clusters missing from the current model. With case study [32, 33], this platform is tested to be valid to segment sequential data into interpretable discrete states.

Every hierarchical model in this platform has two pieces, an allocation model and an observation model. The allocation model defines a probabilistic generative process for assigning or allocating clusters to data atoms. Each allocation model defines a joint distribution:

$$\log p(\pi, z) = \log p(\pi) + \log p(z|\pi).$$

(3)

Two types of variables are involved, cluster probability vectors $\pi_j$ and discrete assignments $z_n$ at each data atom indexed by $n$. A set of global cluster probabilities $\pi_0$ are firstly generated. Depending on the model, several more cluster probability vectors $\pi_j$ may next be generated. Then, cluster assignment variables $z_n$ at each data atom $n$ can be drawn. For example, the complete allocation model of a simple finite mixture model with $K$ clusters would be as follows:

$$\pi_0 \sim \text{Dirichlet distribution}_K(\alpha),$$

$$z_n \sim \text{Categorical distribution}(\pi_0, \ldots, \pi_K).$$

(4)

A stick-breaking distribution is used instead of extending this to a Dirichlet process mixture model:

$$\pi_0 \sim \text{Stick breaking distribution}(\alpha),$$

$$z_n \sim \text{Categorical distribution}(\pi_0, \ldots, \pi_K).$$

(5)

Variation inference for allocation models tries to optimize an approximate posterior:

$$\log q(\pi, z) = \log q(\pi|\theta) + \log q(z|r).$$

(6)

The optimization objective is to make this approximate posterior as close to the true posterior as possible. The objective incorporates terms from the observation model as well. The optimization finds values for the free parameters, $\mathcal{Z}$, which counts $\theta$ and assignments $r$, which make the objective function as large as possible.
In this study, the convolution layer is used to learn and extract features from the input data before trajectory prediction. Convolution layers are the major building blocks in convolutional neural networks (CNNs). CNN is a specialized type of neural network model designed for working with two-dimensional image data, although they can be applied with one-dimensional and three-dimensional data. The core of the CNN is the convolutional layer that performs convolution. The convolution is to apply a filter to an input that results in activation. It is a linear operation that involves the multiplication of a set of weights with the input. The multiplication is performed between an array of input data and an array of weights with the same size as the input. This operation is called a filter or a kernel. Repeated application of the same filter to the input results in an array of output values representing a filtering of the input. Such output array is a map of activations, called a feature map. The feature map indicates the locations and strength of some detected features of the input. Once feature map is created, each value can be passed in the feature map through nonlinearity. In general, the convolution layers can be used for the extraction of the features from the input and thus refine information from a large amount of input.

In general, CNNs are deep feedforward neural networks and are designed to process data that come in the form of multiple arrays [22, 34]. The architecture of a typical CNN composes of convolutional layers, pooling layers, flattened layers, and fully connected layers, shown in Figure 2. The pooling layers are used to reduce the dimensions of the feature maps. It reduces the number of parameters to learn and reduces the amount of computation performed, but pooling layers are not necessary.

In this study, the convolutional layer is used to learn and extract features. The final effect of adding the pooling layer or not is compared. Based on the same training data and data structure, the prediction model contains a pooling layer before a RNN layer has a mean absolute percentage error of 40.28%, while the model has a mean absolute percentage error of 2.04% without the pooling layer. Since the inclusion of the pooling layer degrades the prediction accuracy on nonmotorized vehicles’ trajectories, the pooling layer is deleted from models.

3.4. Sequence Prediction. Trajectory prediction is done by a recurrent neural network (RNN) in this study. RNN is a neural network used to process sequence data. Compared with general neural network, it can process the data of sequence changes. A time series is a series of data indexed (or listed or graphed) in order of time. Most commonly, a time series has a successively and equally spaced time gap. In this research, nonmotorized vehicles are tracked and the computer vision algorithm outputs their locations every two frames, which is a time interval of 1/12 s.

Recurrent neural networks (RNNs) are good at working with tasks that involve sequential inputs. RNNs process an input sequence one element at a time while maintaining in hidden units a “state vector” that contains information about the history of past elements of the sequence [34]. If a sequence is too long, RNNs are hard to carry information from earlier time steps to later ones. During back propagation, RNNs face the problem that gradients explode or vanish over many time steps. Gradients are values to update a neural network weight. The vanishing gradient means the gradient shrinks. As a result, if a gradient value becomes too small, it actually does not contribute much to learning. Then, in RNNs, layers that get a small gradient update stop learning. These are usually the earlier layers. These earlier layers do not learn, and the RNNs forget them in longer sequences, resulting in a short-term memory.

To get over that, long short-term memory (LSTM) as a special type of RNN is proposed mainly to solve the problem of gradient disappearance and gradient explosion in the training process of long sequences. LSTM has internal mechanisms called gates that can regulate the flow of information. These gates can learn which data in the sequence are important to keep or remove. By this, it can pass relevant information down the long chain of sequences and make predictions. It can be thought of as the memory of the network and can carry relative information throughout the processing of the sequence. LSTM has chain-like structure. Instead of having a single neural network layer, there are four interacting layers. In Figure 3, the orange boxes are learned neural network layers, and the red circles represent pointwise operations such as vector addition and matrix multiplication. As processing goes on, information gets added or removed from the cell state via gates. Based on the input of the current state $X_t$, the previous cell state $C_{t-1}$, and hidden information received from the previous state $h_{t-1}$, the current cell performs operations. The first step is to decide what information to throw away from the block. This is done by a sigmoid layer called the forget gate. Next, what new information from the input to store in the cell is decided. This is done by a sigmoid layer called the input gate, to decide which values to update and a tanh layer to create a vector of new candidate values. The next is to conditionally decide what to output based on input and the memory of the block. Compared with ordinary RNNs, LSTM can perform
better in longer sequences. The study has found that forget gate is the most important of the four gates of LSTM and input gate is the second important gate [35].

Gated recurrent unit (GRU) is a newer generation of RNNs, shown in Figure 4. GRU has gotten rid of the cell state and used the hidden state to transfer information. It has only two gates, the reset gate and the update gate. The update gate is similar to the forget and input gates of an LSTM. The update gate decides what information to remove and what new information to add. The reset gate is used to decide how much past information to forget. GRU has fewer tensor operations and is a bit faster than LSTM. The study compares LSTM and GRU and finds that GRU outperforms LSTM on all tasks except language modeling [35].

### 4. Data Collection

#### 4.1. Data Collection

This study records video data. The camera captures 4K video, 24 frames per second. The computer vision outputs objects’ locations every 2 frames.

An overpass at Yingao West Road, Shanghai, China, is chosen as the shooting site. The road section avoids the interference of entrances, exits, or intersections. The shooting view is shown in Figure 5. This section has two virtual lanes with width of 3.5 m, enough for overtaking. Clear marks were made, and the distances between the marks were recorded to create a mapping between the video image and real-world coordinates. A total of 10 hours of video were recorded, from morning to evening. During peak hours, the volume of nonmotorized vehicle reaches 3000 vehicles per hour.

In the computer vision algorithm, object tracking is applied between the cyan and blue lines in Figure 5, with a length of 26 m. Nonmotorized vehicles are tracked within this range, and their trajectories are extracted. Object tracking is done by object detection and object tracking. Due to the performance limitations of the target detection algorithm, error accumulates. Under complex situations such as object occlusion, fast moving, and lighting changing, the Kalman filter algorithm proposed by Rudolf Emil Kalman...
can help to compensate for fluctuations and missing measurements [36]. Trajectories after applying the Kalman filter are smoother as shown in Figure 6.

All the overtaking events are extracted and recorded to get the overtaking trajectories. Figure 7 shows an overtaking event. When identifying and extracting trajectories of nonmotorized vehicles that execute overtaking maneuver, trajectories of surrounding vehicles are also extracted. The surrounding vehicles are divided into the overtaken and the others. It is worth mentioning that trajectory extraction is based on the overtaking vehicles. Based on the complete trajectory of overtaking vehicles in the range of object tracking, locations of the surrounding vehicles within the frames when the overtaking vehicles show up are extracted. This means the trajectories of overtaking vehicles are complete, but trajectories of the others may be incomplete. Their trajectories may start or end in the middle of the tracking range. Although there may be large amounts of other vehicles, their trajectory points are less in amount. After extracting overtaking events, a total of 1179 nonmotorized vehicles are tracked, among which 170 are overtaking vehicles and 181 are overtaken vehicles. There is an average of 55 track points of each overtaking vehicle, 53 track points of each overtaken vehicle, and 29 track points of each other vehicle.

4.2. Statistical Analysis. Figure 8 shows the speed distribution of different types of operations. Each distribution is normalized so that the total area of the histogram equals 1. The lateral speed range of overtaking vehicles is the widest. A large proportion of lateral speed of overtaking vehicles
distributes on the right side of x-axis, representing a general swerving to the left. There is normal shaking left and right during cycling. Like overtaken and other vehicles, their lateral velocity distribution is basically symmetrical about 0, representing their even left and right shaking.

The average longitudinal speed of overtaking vehicles is the highest (5.7163 m/s for mopeds and 5.0291 m/s for r-bicycles), followed by other vehicles (4.4435 m/s for mopeds and 4.1176 m/s for r-bicycles), and the overtaken vehicles have the lowest average longitudinal speed (4.3221 m/s for mopeds and 4.0156 m/s for r-bicycles). As speed differences between different types of operations are significant, in the following study, speed is used as one of the classification indicators and proposed as one of the conditions for subdividing overtaking trajectory segments.

Figure 9 compares speed before and after overtaking. Each distribution is normalized so that the total area of the histogram equals 1. The lateral speed of vehicles taking an overtaking maneuver before and after the overtaking point (the point when the overtaking vehicle reaches the same longitudinal location as the overtaken vehicle) is different. Overtaking vehicles generally shift to the left in the forward direction before overtaking and move to the right after overtaking is completed. At the same time, after reaching the overtaking point, they usually continue to pass at a higher speed.
5. State Clustering

Unsupervised approaches can automatically cluster dataset into groups without fixed number of clusters. Each cluster represents one unique state of cycling. All models are based on unsupervised learning, and thus, there is no standard or set classification label. The effectiveness of the approaches based on their ability to differentiate three types of vehicles and their ability to learn differences in cycling preference, in other words, their ability to differentiate cycling features of different operations, are evaluated.

5.1. Segmentation Results and Feature Comparisons. A good clustering method should be able to learn the differences in features in a joint way. This section shows the result of clustering and compares the ability of three models to differentiate cycling features. Speed is one of the main factors that differentiate overtaking from normal cycling and reflect distinct cycling behavior. Clustering should be sensitive to speed variation. At the same time, speed difference within the same cluster is better to be small to show consistency. It is the same with acceleration.

Figure 10 shows the lateral and longitudinal speed distributions of different clusters based on three approaches. There are overlaps between clusters no matter in which model. The last figure of each subplot shows the speed distribution of the overtaking vehicle as reference. Speed in longitudinal direction, the direction along the road segment, attributes the most to the total velocity. Speed is normally around 4 to 5 m/s, and the difference in distribution of different states is not conspicuous. Comparing the distribution shape and value, several states are more similar to overtaking than others. For example, state 3 clustered by HDPHMM is most similar to overtaking.

Two heat maps, Figures 11 and 12, show the distribution of speed and acceleration in different clusters (states) under three clustering approaches. The darker the cell, the more feature value is located in the cell. It should be noticed that, since the amount of data in each cluster (state) is different, the heat map shows not the amount of data, but the proportion of feature value in this cluster. Speed difference in longitudinal direction between clusters is slight, but more conspicuous in lateral direction. Carefully comparing, it can be noticed that state 3 in both HDP topic model and HDPHMM has a higher proportion of speed in higher value and an even acceleration or deceleration.

The bounding boxes are positions of objects in each frame. On continuous homogeneous roads, cycling behavior is theoretically independent of location along road. On real road, there might be segments that have better and more suitable condition for overtaking. So, there may exist a preferred overtaking segment where overtaking is more likely to happen. Despite the difficulty to interpret relations between the longitudinal location and overtaking maneuver, studies have shown that overtaking on the left is preferred [6]. Assume that commonly cyclists ride in the middle of a nonmotor lane or a bit right in the middle of a nonmotor lane [37], so most of the overtaking trajectory is supposed to be left than the common trajectory. In the established coordinate axis, the positive direction of horizontal coordinate axis corresponds to the left side of the forward direction.

The distinctiveness of location in three models is not conspicuous. As shown in Figure 13, the last figure in each subplot shows the actual location of every overtaking event. The location is around the overtaking point, 5 meters ahead and behind. Again, comparing the distribution shape and value, state 3 in HDPHMM shows the highest similarity with the actual overtaking location.

5.2. Segments’ Duration. The purpose of clustering is to automatically learn and differentiate overtaking segment from the whole trajectory without prior information. Since trajectory extraction is based on frames where overtaking vehicles show up, the trajectories of overtaking vehicles are complete, but the trajectories of other vehicles may be incomplete. Their trajectories may start or end in the middle of the tracking range. On average, there are 55 track points of each overtaking vehicle, 53 track points of each overtaken vehicle, and 29 track points of each other vehicle.

With trajectory location, speed, and acceleration, three models cluster each data point into infinite latent states. The state numbers or state codes are only tokens of different
states. The magnitude of the state number has no special meaning. The main cycling states can be deduced from the frequency of fragments in each state. The ability to avoid scattered results of clustering models can be evaluated by the duration of fragments in each same continuous state.

Three models show different results of clusters. Again, the code of the state has no special meaning but only to distinguish different clusters. The state with the same code number in different models does not mean the two states are the same. As shown in Figure 14, there is only one main state in the HDP topic model and HDPHMM, and the clustering of HDP topic model is more concentrated. The DP mixture model shows two main states. In the DP mixture model, there is no cluster that contains certain amounts of records and meanwhile shows a high proportion of overtaking. In the main state of HDP topic model and HDPHMM, three types of vehicles all present a large number of records. This can be interpreted that the overtaking vehicles are basically cycling normally like other vehicles, which is the main state, but they encounter a slower subject and trigger their overtaking maneuvers, and then, they present different cycling features that represent overtaking. State 3 in the HDP topic model and HDPHMM has a highest proportion of overtaking except for state 26 in HDPHMM that have only one record of overtaking.

In terms of duration, as shown in Figure 15, HDPHMM outperforms the other two models that the clustering gets more continuous results. Average durations of the main states in the three models have the highest value. In the DP mixture model, the duration is proportional to the amount of the corresponding state. In the main states, state 6 and state 9, the duration is relatively longer. Based on the amount of records, it is inferred that there is only one main state in the clustering results of HDP topic model, and the same is true for HDPHMM. Besides main state in these two models, there is other state showing a relative long duration. For example, in the HDP topic model, while state 2 is the main state, the duration of state 3 is also long. State 3 is not the main state of cycling in terms of cluster atom amount, but it is as continuous as the main state. In the main state, state 2, the three types of nonmotorized vehicles have a similar shape of duration distribution, while in state 3, the data frequency of overtaking vehicles is higher and their duration distribution is different from the other two types of vehicles, which are longer. In HDPHMM, the results of state 2 and state 3 are similar to the HDP topic model. However, the duration of the other nonmajor states is not as discrete. In state 3, where the duration is almost as long as the main state, the duration of overtaking vehicles is also longer than the other two types of vehicles.

In general, in the DP mixture model, the overall average duration of every same state is 0.60 second, 0.61 second in HDP topic model, and 1.10 second in HDPHMM. In the DP mixture model, durations in the main state are significantly longer than that in the other states and durations in the other
states are rather scattered. In HDPHMM, the duration in every state has a relatively higher value than the other two models.

Suppose that overtaking vehicles show larger possibility to present overtaking state and last longer time in the overtaking state than other vehicles. State 3 in both HDP topic model and HDPHMM model meets the requirement. What is more, the average longitudinal speed presented in state 3 is exactly the highest, with 5.15 m/s in HDP topic model and 5.30 m/s in HDPHMM. Average longitudinal speeds in other states are all less than 4.7 m/s.

5.3. Thresholds for Identifying Overtaking. State 3 in HDP topic model and HDPHMM is identified as the overtaking state. This section compares the overtaking state and other states and calculates the threshold for identifying the overtaking state.

As shown in Figure 16, the speed and acceleration of overtaking and non-overtaking state identified by automatic clustering are compared. The range of lateral speed of overtaking state is wilder than that of non-overtaking state. The longitudinal speed of overtaking state is higher than that of non-overtaking state. The range of acceleration value shows clearer differences between overtaking and non-overtaking states.

The first three atoms of data identified as overtaking state represent the features at the beginning of an overtaking behavior, as shown in Table 1. There are significant differences between overtaking and non-overtaking states obtained by the automatic clustering algorithm. During the overtaking state, the lateral movement is more intense, the longitudinal speed is higher, and it is accompanied by higher acceleration. The cycling differences before and after reaching the overtaking point have been analyzed in Chapter 4.2 statistical analysis: there is commonly a left turn before reaching the overtaking point and gradually returns to the original virtual lane after reaching the overtaking point. What is more, cyclists tend to maintain a higher speed after reaching that point. The difference between longitudinal speed of overtaking beginning and mean velocity of whole overtaking section is slight, but clear longitudinal

Figure 11: Speed’s heatmap of different clusters in different models.
Acceleration can be observed at the beginning of overtaking. Both clustering approaches fit the above findings. The numbers in the brackets in Table 1 represent the corresponding quantiles of feature value of overtaking or overtaking beginning phrase whose value is equal to the 85% quantile feature value of non-overtaking state. For example, 6.098 is the 85% quantile of longitudinal speed of non-overtaking state in the HDP topic model and also the 63.55% quantile of overtaking state and 69.35% quantile of overtaking beginning state. In other words, if we use 6.098 to screen for overtaking state, 63.55% of the overtaking state data and 69.35% of the overtaking beginning data will also be removed when 85% of the non-overtaking state data are removed. That is to say, the smaller the value in the bracket, the more suitable the threshold is. Because under this case, 85% non-overtaking state will be removed while remain more records in overtaking beginning or during overtaking states. The absolute values of lateral offsets of each 0.25 s under different states are also compared, but the difference is not prominent. The lateral offset is not drastic at the beginning of overtaking.

In the results of HDP topic model, the beginning state of overtaking is screened, which is longitudinal speed higher than 6.098 m/s and lateral acceleration greater than 0.369 m/s$^2$ at the same time. In HDPHMM, the threshold of overtaking beginning is lateral speed higher than 0.356 m/s and lateral acceleration greater than 0.368 m/s$^2$. Under this condition, the threshold can classify whether the record is overtaking beginning state at an accuracy of 0.900 in HDP topic model and 0.897 in HDPHMM.

In this section, the effectiveness of the automatic clustering method is proved, and the HDPHMM can obtain more continuous and characteristic clustering results, while the DP mixture model is not as befitting in this task. According to the results of clustering, the trajectory segment representing overtaking features can be easily obtained. We can mark some vehicles taking overtaking actions and infer the class of overtaking state in the whole data according to the distribution of clustering results of these marked vehicles. According to the clustering results, on the one hand, it can reduce the manual work of deciding the trajectory category frame by frame. On the other hand, it can obtain

**Figure 12: Acceleration’s heat map of different clusters in different models.**
the internal characteristics of overtaking behavior and further infer the overtaking threshold.

6. Trajectory Prediction

Deep learning can make predictions by learning data features. The classified data have consistency in cycling features. In this section, it is proved that classified data are conducive to improving prediction accuracy and reducing requirements for data volume.

6.1. Prediction on Overall Data. Trajectory prediction can be achieved through feature learning from trajectory sequence. In this study, historical trajectories, speed, acceleration, and external information (density, clearance, spacing, etc.) are used to predict the position of an object after a certain time. Models are tested based on the whole data without distinguishing behavior types at first, to determine a model with better performance.

Comparing two model structures, convolution-maxpooling-LSTM/GRU and convolution-LSTM/GRU-LSTM/GRU, the prediction accuracy of the latter is much higher than the former. What is more, using the last state for each sample in a batch as the initial state for the sample at the same index in the following batch can largely increase model accuracy. In this section, the object position after 1/12 second is predicted by last 0.5 second historical information. The model structure of convolution-RNN(LSTM/GRU)-RNN(LSTM/GRU) reaches higher accuracy. The amount of trainable data will affect the accuracy of deep learning. A group of data points is packed every 0.5 s to construct a time series as input. There are a total of 28788 pieces of data, with 75% as the training set and the rest as the test set.

Table 2 shows the results of two types of RNN layers. Under the same training data and training method, the LSTM layer to learn and predict the results from CNN can achieve slightly better results than the GRU layer, though the training speed of GRU is faster. Different from the study [35] that found GRU is better than LSTM in most of the tasks, here LSTM shows better performance than GRU. As shown in Figure 17, the predicted trajectory overlaps with the actual trajectory, and the absolute error of the trajectory is about 3 cm.

---

**Figure 13:** Comparison of locations of points in each cluster.
The dataset is classified, and the objects are divided into overtaking vehicles, overtaken vehicles, and other vehicles. Again, as the extraction is determined by whether overtaking vehicle shows up within the tracking range, the trajectory of the overtaking vehicle is complete, while that of other vehicles may be incomplete. In the valid data, there are 168 overtaking vehicles, 176 overtaken vehicles, and 682 other vehicles. The amount of trajectory points of every overtaking vehicle (an average of 55 records each) is larger than that of every other vehicle (an average of 47 records for every overtaken vehicle and an average of 26 records for every other vehicle).

As shown in Table 3 the impact of trainable data amount on modeling is not as great as expected. The data are packed every 0.5 s with 6 records, and each record has 6 variables. Thus, the data structure of overtaking vehicle shows there are 8089 packages of zipped records. Here is little difference in modeling precision between the LSTM layer and the GRU layer. In general, the LSTM layer is slightly better than the GRU layer. LSTM has advantages in trajectory prediction of overtaking and other vehicles, while GRU does better in predicting the trajectory of overtaken vehicle.

According to the statistical analysis above, the overtaken vehicles cycle more stably, with lower lateral speed and lower moving velocity. Overtaking vehicles have the highest speed, and their lateral shift is more violent. Compared with the overtaken and overtaking nonmotorized vehicles, other vehicles have the shortest trajectory segments and various riding styles. Some of the lateral shifts are similar to the characteristics of the overtaken vehicles, while others are similar to overtaking. Their velocity covers the range of overtaken and overtaking nonmotorized vehicles. Therefore, it can be inferred that LSTM has a relatively stable advantage in predicting trajectory with more or less changes. LSTM shows high prediction accuracy for stable trajectory segments of overtaken vehicles, trajectory segments of overtaking vehicles with relatively severe fluctuation, and trajectory segments of other vehicles with both stability and changes. However, GRU is more suitable for the prediction of smooth trajectory segments. It has the highest accuracy in
predicting the trajectory of overtaken nonmotorized vehicles, but has disadvantages in predicting the trajectory of overtaking vehicles, which has more drastic changes.

The trajectory prediction based on vehicle classification can guarantee high precision even when the amount of data in the training set is largely reduced. The absolute error of the predicted position is 3-4 cm.

6.2. Prediction on Segmented Data. Prediction based on vehicle classification can maintain high accuracy and reduce the demand for training data. The vehicle classification is actually a simple screening of vehicle trajectories. The trajectories with similar characteristics are grouped together and predicted in groups. It is similar to clustering. The automatic clustering algorithm classifies the states of trajectory points, then obtains the state that reflects the overtaking features, and obtains the overtaking trajectory segments. Overtaking state is identified in the HDP topic model and HDPHMM. To predict trajectory of overtaking vehicle is to learn records that reflect the homogeneous cycling characteristics and predict their future location.

The results are listed in Table 4. The lite version of input variables includes speed, acceleration, and position of the overtaking vehicle. The full version of input variables includes all information in the lite version, plus speed, acceleration, and position of the overtaken vehicle, and more surrounding information, including density, average spacing, minimum spacing, and speed difference.
The results shown in Tables 4 and 5 are compared. In general, the prediction accuracy of overtaking trajectory segments screened by HDPHMM is higher because segments based on HDPHMM are more continuous. Lite and full versions of input variables show large differences. With complicated inputs, prediction accuracy deteriorates. Adding deep learning layers does not help either. Although the convolutional layer has the ability to process complex information and extracts features, it is difficult for the model to obtain key information due to overly complex input. When the focus is clear, adding too much other information only confuses the model and does not improve accuracy. The prediction on trajectories of overtaking vehicles and overtaking segments is compared, although the training set is
largely reduced, the latter shows higher accuracy. In other words, trajectory prediction based on grouped data with the same features not only improves the prediction accuracy, but also reduces the dependency on the amount of data for model training.

6.3. Prediction versus Time Intervals. Previous experiments verify the applicability of models and compare the accuracy of different clustering methods and prediction models. These predictions all use information within the last 0.5 second to predict a next track point, that is, position after 1/12 second. In practical applications, there might be special requirements for prediction intervals. The following part tests the prediction periods. Since the amount of data is fixed, the volume of batches after packaging at different time intervals is different. In other words, the amount of data available for training is different when 6 track points are taken as a group and when 12 track points are taken as a group. As the packaging capacity increases, the amount of data in the training set and test set decreases, but always guarantee 80% of the data batches for model training.
Figure 18 shows the accuracy of object position prediction in different prediction intervals using the same historical information. As the prediction interval increases, the amount of trainable data decreases, as shown in the gray background bars. At the same time, with the decrease in data amount and the increase in prediction time interval, the prediction error presents a rising trend. Among them, due to the limitation of the amount of data in the training set, the prediction error based on overtaking segment increases with the prediction interval, and the error increases more greatly from 6 cm to 27 cm. For the prediction based on whole trajectory of the overtaking vehicle, the error increases from 3 cm to 10 cm.

There are lots of research studies about trajectory planning for autonomous driving. Glaser et al. proposed a vehicle trajectory planning algorithm for autonomous vehicles, which output a new optimal trajectory every 10 milliseconds taking minimizing the risk of collision and optimizing the performance indicators into account [38], but with no consideration of nonmotorized vehicles. Li et al.
focused on the real-time trajectory planning of autonomous driving in a realistic urban environment and the trajectory planning algorithm was executed every 100 milliseconds [25]. This algorithm did consider pedestrian and other traffic participants, but with an assumption that pedestrians and other vehicles maintain the current speed and direction within 3 s. Wang et al. proposed a method for trajectory planning of autonomous vehicles that considers the motion prediction of other traffic participants, and the total time to complete trajectory planning is within 100 milliseconds [39]. The minimum prediction interval in this study is 1/12 s, approximately 0.083 s. In the above studies, trajectory planning is updated every 0.1 s. The prediction error of nonmotorized vehicle position is 3 cm after 0.083 s and 6 cm after 0.25 s, which is sufficient for updating the trajectory planning algorithm to avoid collisions.

7. Conclusion and Discussion

Two targets are achieved in this work, automatic trajectory segmentation and trajectory prediction. For the former, unsupervised learning is used to distinguish trajectory segments. Speed is found the key factor for distinguishing overtaking and non-overtaking segments. Among the three unsupervised clustering methods, HDPHMM not only maintains the continuity of clustered segments but also demonstrates high differentiation for velocity. The non-parametric clustering automatically completes trajectory segmentation. With part overtaking nonmotorized vehicles marked, the state representing overtaking features in the whole data can be inferred from the state distribution of these marked vehicles. It is no longer necessary to distinguish the overtaking segment frame by frame manually. This method can be applied to the study of cycling behavior, which can obtain a large amount of data easily while retaining the hidden characteristics of the overtaking segment and not being affected by the subjective judgment.

Using deep learning to predict trajectory of nonmotorized two-wheelers can achieve high accuracy. In the case of sufficient data, the prediction error of trained model is 2 to 3 cm, which can meet the needs of vehicle trajectory planning and obstacle avoidance. We can make several conclusions based on prediction models: (1) compared with GRU, LSTM is more suitable when combined with the convolutional layer to forecast trajectory. RNN is suitable for time-series analysis. Although GRU is a novel and faster RNN layer, it does not show advantages for trajectory prediction of nonmotorized two-wheelers. LSTM is more suitable for trajectory analysis and prediction. (2) The categorizing or subdividing of trajectory can improve the prediction accuracy and reduce the requirement of training data volume. The categorizing has two aspects, one is the vehicle type. The entire trajectory of the overtaking vehicle is adopted. Distinguishing trajectories according to cycling behavior types can maintain high accuracy on the premise of reducing the amount of trainable data, and the prediction error is 3 to 5 cm. Only part of it is actually in the overtaking process. Another aspect is to categorize on the level of trajectory points, and those points that are actually under overtaking process are chosen. The amount of data is greatly reduced, but the model can still maintain prediction accuracy and the prediction error is about 5 cm. (3) For learning, the weight of input variables is a hard task, and overly complex input variables although provide comprehensive surrounding information will reduce the accuracy of prediction. Higher accuracy can be achieved by inputting only the information of the subject to be predicted. (4) With the decrease in trainable data amount and the increase in prediction interval, prediction error increases. If the prediction of the object’s position after a longer period of time is necessary, it is needed to provide more trainable data. Existing trajectory planning algorithm updates results every 0.1 s. Under this condition, the error of segmentation plus prediction method in this study is less than 5 cm, which suits the requirements of estimating the location of obstacles in advance and updating trajectory plan.

Approaches in this study can be applied in unsupervised trajectory segmentation, and the target state from the marked objects is deduced. Limitations of this method must be acknowledged, such as scattered segments, incomplete overtaking segments, and trajectory segments of other vehicles misidentified as overtaking. In terms of accuracy, marking frame by frame manually has the highest precision, but costs too much time and cannot avoid subjectivity. The method presented in this study can greatly shorten the marking time, but it also brings error. This method can obtain a large amount of data that has similarity to overtaking and analyze the cycling behavior of nonmotorized two-wheelers from the potential characteristics of the data. Another limitation is that despite the differences between mopeds and regular bicycles, they are together identified as overtaking, overtaken, and other vehicles. Although there is little regular bicycle taking passing maneuver, subdividing overtaking into regular bicycle overtaking and moped overtaking may present other results.

Data Availability

The trajectory data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References


