

Personalized Vehicle Trajectory Prediction Based on Joint Time-Series Modeling for Connected Vehicles

Yang Xing , Chen Lv , Member, IEEE, and Dongpu Cao , Member, IEEE

Abstract—Motion prediction for the leading vehicle is a critical task for connected autonomous vehicles. It provides a method to model the leading-following vehicle behavior and analysis their interactions. In this study, a joint time-series modeling approach for leading vehicle trajectory prediction considering different driving styles is proposed. The proposed method enables a precise and personalized trajectory prediction for the leading vehicle based on limited inter-vehicle communication signals, such as the vehicle speed and acceleration of the front vehicles. Three different driving styles are first recognized based on an unsupervised clustering algorithm, namely, Gaussian Mixture Model (GMM). The GMM generates a specific driving style for each vehicle based on the speed, acceleration, jerk, time, and space headway features of the leading vehicle. The feature importance of driving style recognition is also evaluated based on the Maximal Information Coefficient (MIC) algorithm. Then, a personalized joint time series modeling (JTSM) method based on the Long Short-Term Memory (LSTM) Recurrent Neural Network model (RNN) is proposed to predict the front vehicle trajectories. The JTSM contains a common LSTM layer and different fully connected regression layers for different driving styles. The proposed method is tested with the Next Generation Simulation (NGSIM) data on the US101, and I-80 highway dataset. The JTSM is tested for making predictions one to five seconds ahead. Results indicate that the proposed personalized JTSM approach shows a significant advantage over the baseline algorithms.

Index Terms—Trajectory prediction, time-series modeling, driving style recognition, RNN, LSTM, connected vehicle.

I. INTRODUCTION

A. Motivation

CONNECTED vehicles and vehicle-to-vehicle (V2V) communication techniques are designed for safer and more efficient transportation, which plays a key role in constructing next-generation road mobilities and transportations [1]–[3]. Also, the V2V technique will significantly influence the traffic mode design of future intelligent and automated driving vehicles [4], [5]. The final vision of V2V is to benefit all road entities such as buses, vehicles, motorcycles, and pedestrians to safely use the

Manuscript received August 20, 2019; revised November 3, 2019; accepted December 12, 2019. Date of publication December 16, 2019; date of current version February 12, 2020. This work was supported in part by the SUG-NAP Grant M4082268.050 of Nanyang Technological University, Singapore, and in part by the A*Star Grant 1922500046. The review of this article was coordinated by Prof. S. Kamalasadan. (*Corresponding authors: Chen Lv; Dongpu Cao.*)

Y. Xing and C. Lv are with the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore 639798 (e-mail: xing.yang@ntu.edu.sg; lyuchen@ntu.edu.sg).

D. Cao is with the Mechanical and Mechatronics Engineering, University of Waterloo, Waterloo, ON N2L3G1, Canada (e-mail: dongpu.cao@uwaterloo.ca).

Digital Object Identifier 10.1109/TVT.2019.2960110

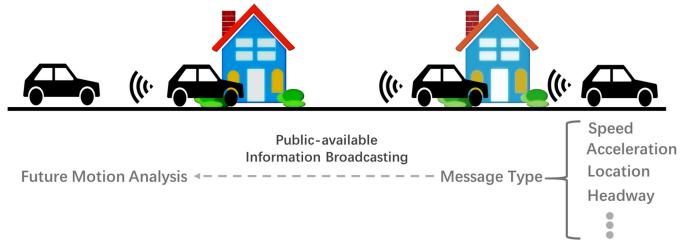


Fig. 1. Leading-following pair communication example.

public road and transport based on short-range and long-range broadcast [6].

V2V techniques are expected to primarily increase road safety and traffic efficiency in the next decade. However, how to maximize the influence of V2V techniques based on the limited short-range signal transfer capability and the varying quantities of signals to enhance the scene understanding ability of the connected vehicles are still an open question [7]. For example, as shown in Fig. 1, the following cars are interested in predicting the future trajectories of the leading vehicles so that to understand the future motion of the leading vehicles better. Conventional methods to predict the leading vehicle trajectory is to estimate the motion based on vehicle perception methods. However, this approach is more complex and less efficient than the V2V technique which allows the leading-following pairs to share brief status information.

This study aims to provide an efficient and personalized motion prediction algorithm based on the joint estimation of driving styles and future vehicle trajectory for the connected vehicles. One of the natures of the human learning process is always solving similar tasks based on similar experience and knowledge. In this study, the same learning scheme for the connected vehicles towards high precise motion prediction is developed. Different driving styles are unsupervised learned in the first; then, it can be used to group vehicles with similar behaviors together. Separate prediction networks with a shared temporal feature extraction layer can be used to improve the final prediction accuracy. Based on the proposed system, the connected vehicles will obtain a more informative understanding of the nearby vehicles based on the limited transferable vehicle data through the V2V network. Moreover, the prediction accuracy for the future trajectory prediction of the leading vehicle can also be further improved. The literature review part in the next will focus on the systems and methodologies used for future motion prediction.

B. Literature Review

In [8], Lefèvre et al. proposed a well-organized survey study on vehicle motion modeling and prediction, which separated the motion prediction methodologies into three different groups, namely, physics-based motion models [9], [10], maneuver-based models [11], [12], and interaction-aware motion models [13], [14]. Specifically, physics-based motion prediction models represent the vehicle dynamic according to the laws of physics and make trajectory prediction based on dynamic and kinematic models. Maneuver-based models, on the other hand, represent the vehicles as independent entities, and the motion consists of a series of discrete maneuvers. Last, the interaction-aware models describe the vehicles as maneuver entities, which can interact with each other according to the traffic laws and the dependence between different vehicles. Considering the traffic context into motion prediction enables a longer-term and more precise motion prediction of the surrounding vehicles compared with the physics-based and maneuver-based methods. Therefore, most of current motion and trajectory prediction studies focus on the interaction-aware based approaches.

Most of the early interaction-aware algorithms use Dynamic Bayesian Network (DBN) to learn the mutual dependencies between the multiple entities. In [15], an integrated physics-based and maneuver-based trajectory prediction method were proposed based on the Bayesian network. A generalized time-to-critical-probability (TTCCP) was used to validate arbitrary uncertainty multi-agents driving context and long-term prediction. In [16], a hierarchical DBN model which consists of three dynamic layers, namely, observation layer, dynamic layer, and context layer, was proposed. For each vehicle, the dynamic layer stores the low-level information such as the pose, orientation, and velocity based on the onboard sensors. Then, the context layer captures the high-level semantic information such as the relative location concerning nearby vehicles.

In [17], an interaction-aware intention and maneuver prediction framework were proposed for sophisticated traffic scene understanding. The motion intentions of surrounding vehicles were estimated iteratively based on the game-theoretic algorithm. The spatiotemporal cost maps are used for predicting surrounding vehicle intention considering the traffic context. Then a Bayesian Network was applied to discriminate the possible future maneuvers further and improve the algorithm robustness for the unseen scenarios. In [18], Three types of predictive features were designed for the DBN model to recognize future vehicle maneuvers, which were road structure-based feature (existence of lane and lane curvatures), interaction-aware feature (state of adjacent and leading vehicles), and physics-based features (vehicle dynamics). The DBN model achieved an average of 80% F1 score with an advanced prediction time of 3.75 s. In [19], an agent-sensitive motion prediction method, namely, General Agent Motion Prediction Model for Autonomous Driving (GAMMA) was developed based on the Bayesian inference learning. The GAMMA can predict the multi-agents' motion by integrating different kinematic and geometry features and generate hypotheses trajectories by inferring the human intention with velocity and acceleration characteristics.

Some of the current studies have analyzed the effectiveness of LSTM to the surrounding vehicle motion prediction. Inspired by the social LSTM architecture given by Alexandre *et al.* [20], many studies have successfully transferred this method into the vehicle motion prediction task. For example, in [21], an LSTM encoder-decoder model was proposed for surrounding vehicle maneuver and trajectory prediction. An improved convolutional social pooling scheme was developed for the maneuver-based decoder. The model generated the multi-modal distributions for the future motion of the surrounding vehicles based on the predicted maneuvers. In [22], a multi-agent tensor fusion (MATF) network was proposed to model the social interaction behavior between a varying number of vehicles on the highway. The MATF network was capable of encoding both the scene context image and the past vehicle's trajectories together to make a multi-step trajectory prediction for multiple agents based on the improved social LSTM framework. Similar work can be found in [23], where a stacked LSTM based method was proposed to predict the vehicle trajectories at unsignalized urban intersections. Despite using the mean average error to evaluate the system, the worst 5% and the worst 1% of scores were also used to indicate when the algorithm may fail.

Despite the time-series modeling methods, some studies also prove the efficiency of other neural network models on track prediction. In [24], a driver behavior modeling method was proposed based on the generative adversarial imitation learning (GAIL) methods and NGSIM dataset. The GAIL network can learn emergent driving behaviors properties from the reinforcement learning scheme and allocate a high likelihood to expert actions. In [25], an integrated physics-based and interaction-aware-based motion prediction system, namely, interaction-aware Kalman neural network (IaKNN), was proposed based on a hierarchical neural network framework. Several convolutional layers and encode-decode LSTM were used to construct the interaction layer and generate interactive effects among vehicles. In [26], a vehicle trajectory prediction method based Kalman Filter (KF) was developed under the framework of the connected vehicle. Results indicate that trajectory prediction, especially longer-term prediction, can perceive some threats sooner than onboard sensors.

C. Contribution

Although studies have been proposed for interaction-aware based motion prediction, these studies mainly try to solve this problem with a unified model without considering any different patterns that may exist within the traffic model. One of the major behaviors behind the human task solving process is to solve the problems that have similar features with similar methods. Therefore, in this study, a personalized future trajectory prediction is developed to enhance the scene understanding capability for the connected vehicles. The contribution of this study can be summarized as follows.

First, an unsupervised driving style recognition method for connected vehicles on the highway is proposed. This study focuses on the analysis of the driving style of the leading vehicle with V2V communication, which uses the limited and easy to

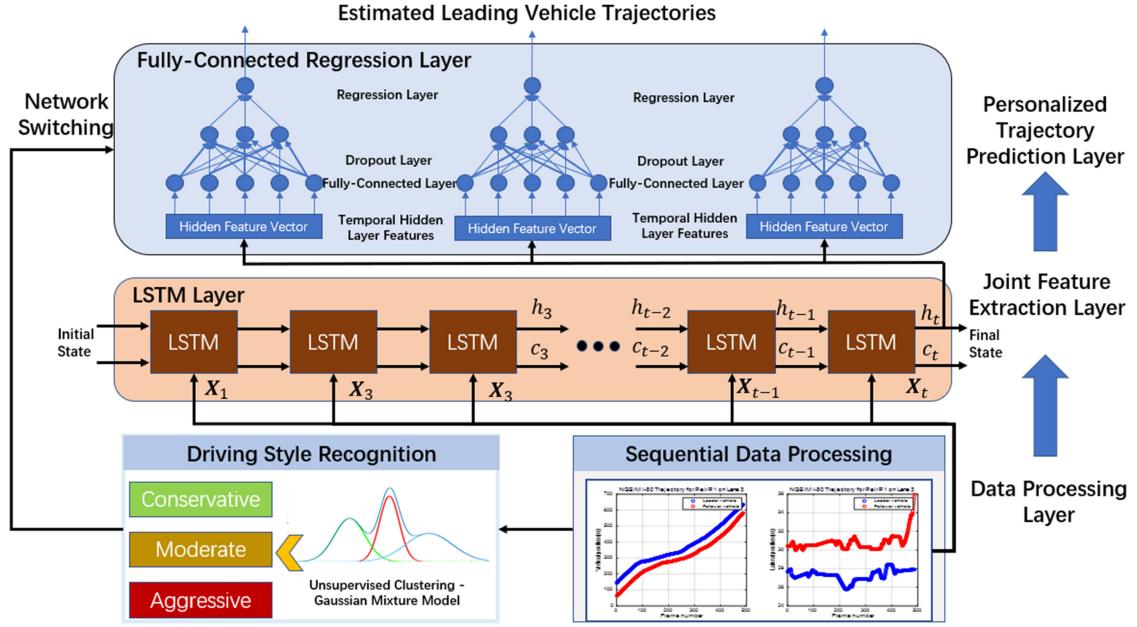


Fig. 2. Illustration of the high-level system architecture.

access signals to measure the driving styles. Then, multiple features that can be used to estimate the driving styles on the highway are selected and quantitatively analyzed. The feature importance for determining the driving style is evaluated based on the MIC algorithm, which can be implemented easily and effectively in a real-world application. Last, a shared deep LSTM-based RNN network is proposed for trajectory prediction. Different fully-connected regressions networks are proposed for different styles with a shared LSTM layer. The joint framework can decrease model complexity and increase prediction accuracy. The personalized JTSMS method is comprehensively evaluated and tested with different prediction horizons and multiple metrics.

D. Paper Organization

The remainder of this paper is organized as follows. Section II introduces a general framework for personalized JTSMS. The GMM based driving style generation and joint deep learning scheme based on the RNN model are described in Section III and Section IV, respectively. In Section V, the experimental design and results are proposed. Finally, this study is concluded in Section VI.

II. THE HIGH-LEVEL FRAMEWORK OF PERSONALIZED TRAJECTORY PREDICTION

In this section, the high-level architecture of the proposed personalized JTSMS is proposed. The algorithm is comprised of three layers, which are the data processing layer, joint temporal feature extraction layer, and personalized trajectory prediction layer. The details of the system architecture and functions are described as follows.

The high-level system architecture with the proposed methodologies is shown in Fig. 2. The general system architecture can

be split into three layers. Specifically, the data processing layer mainly focuses on generating the leading-following pairs of the vehicle trajectory for each lane. Then, the leading vehicle information will be collected and used for unsupervised driving style learning. Since in the beginning, it is hard to generate the driving style labels for each vehicle, which can be applied to describe the difference in the driving styles between each group. The GMM unsupervised clustering algorithm is used to obtain the optimal cluster label for each vehicle based on the hand-craft features. Then, the essential features that highly related to the learned group labels will be evaluated based on the MIC method. With the feature importance analysis, it will be easier to design a more efficient driving style recognition algorithm, which relies on fewer features but can be more robust. A detailed description of the data processing layer that consists of NGSIM data processing, GMM-based driving style clustering, and feature importance estimation with MIC will be introduced in the next section.

Based on the driving style learned in the first layer, a personalized trajectory prediction module for each driving style can be developed. The joint feature extraction layer contains an LSTM layer to extract the time-sequence patterns. Then, three different regression networks are trained separately based on the driving style patterns and the time sequence patterns given by the LSTM layer. The regression networks can learn more temporal patterns by sharing the same LSTM layer as it enables a precious temporal pattern memory for different driving dynamics. It is not necessary to train a separate sequence feature extraction network for each class as the time sequence processing layer should learn as many temporal interaction behaviors as possible. Also, sharing the same temporal feature extraction layer can significantly decrease the model training cost and increase the prediction accuracy compared with training multiple models for multiple classes. A detailed description of the JTSMS modeling

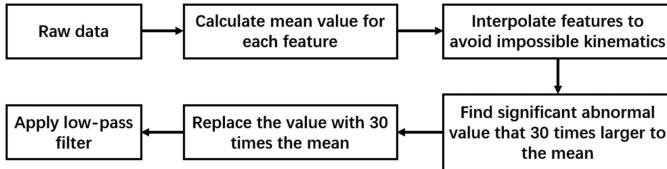


Fig. 3. Data pre-processing procedure.

based on LSTM-RNN will be described in Section IV, and the model evaluation and comparison will be made in Section V.

III. DATA PROCESSING

This section describes the data processing layer for the personalized JTSIM system, which contains NGSIM data processing, GMM-based driving style recognition, and representative feature evaluation for the driving styles with the MIC method.

A. NGSIM Data Processing

In this study, the I-80 and US-101 freeway data are used to generate vehicle trajectory pairs. NGSIM vehicle trajectory data were collected from a different region at a different time, which can reflect congested and moderate traffic conditions. The NGSIM data are well-labeled with the vehicle identity and frame identity, which are suitable for the analysis of car-following behaviors. In this study, the leading-following pairs among different vehicles at different locations are collected. The NGSIM data sampling rate is 10Hz, and the minimum length of each leading-following pair is set as 80, which can cover at least eight seconds of the car leading and following behaviors.

A major issue of the raw NGSIM data is some significant data errors exist. For example, based on the natural NGSIM trajectory, the vehicle can generate a collision path, which is abnormal in the real case [27], [28]. Therefore, in this part, a data pre-processing is applied to clean the raw dataset. The data cleaning procedure given in [27] was partially adopted. A simplified data pre-processing procedure is shown in Fig. 3. As shown in Fig. 3, the impossible kinematics such as suddenly zero speed, impossible high accelerations values, or suddenly zero headway space will be interpolated based on the nearby data. Then, the significant abnormal points which are 30 times larger than the mean value will be regarded as the unusual case and will be replaced by the value based on the mean value. This process is placed before applying a low-pass filter as it was found that the low-pass filter still cannot eliminate such abnormal signals.

B. Driving Style Recognition Based on GMM

In this part, the GMM unsupervised clustering method is applied to generate the most distinctive driving styles for the connected vehicles. GMM can be represented by a weighted sum of sub-Gaussian components [29]. Each cluster is modeled according to different Gaussian distribution function.

The GMM can be described as the following merged distribution function:

$$p(x_i|\theta) = \sum_{k=1}^K \pi_k N\left(x_i|\mu_k, \Sigma_k\right) \quad (1)$$

TABLE I
STATISTICS OF THE FEATURES USED FOR DRIVING STYLE RECOGNITION

Features	Statistic Values	
Velocity	Velocity STD	Mean Velocity
Acceleration	Acceleration STD	Mean Acceleration
Jerk	Max Jerk Jerk STD	Min Jerk Mean Jerk
Space Headway	Max Space Headway Time Headway STD	Min Space Headway Mean Time Headway
Time Headway	Max Time Headway Time Headway STD	Min Time Headway Mean Time Headway

where x_i is the data point and can be multiple dimensions, θ is the parameters of the GMM, which can be represented as $\theta = \{\pi_k, \mu_k, \Sigma_k\}$, π_k are the weight of each component Gaussian distribution function and the sum of π_k equals to one. μ and Σ are the mean and covariance parameter of multivariate Gaussian function, K is the total number of components in the model, $N(x_i|\mu_k, \Sigma_k)$ is the univariate Gaussian distribution function in this case with the form as follows.

$$N\left(x|\mu, \Sigma\right) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu)\right] \quad (2)$$

The GMM model can be trained with the Expectation-Maximization (EM) maximum likelihood estimation algorithm, which computes the maximization of the cost function iteratively. Two performance metrics are used to evaluate the GMM model performance on the unsupervised clustering model, which are the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC), the model parameters are selected according to the joint evaluation of these two metrics.

Most of the existing driving styles recognition systems were designed to recognize the host vehicles [30], [31]. While, in this study, we analyze the impact of the GMM on the recognition of surrounding vehicle driving styles based on vehicle-to-vehicle communication and mainly focus on the leading vehicles. The GMM-based driving style recognition system will be capable of recognizing the leading or the surrounding vehicle styles with limited driving behavior data. Based on the most publicly available data that is allowed to be transferred through vehicles, vehicle velocity, acceleration, space headway, and time headway were extracted. Moreover, the jerk information, which is the derivative of acceleration along time, is also adopted. The vehicle jerk is calculated as follows.

$$Jerk(t) = \frac{acc(t + \Delta t) - acc(t)}{\Delta t} \quad (3)$$

To increase the diversity as well as the robustness of the driving style recognition algorithm, the statistical measurement such as the standard deviation (STD), mean, maximum, and minimum for these features are used. The whole feature vector for the GMM is a 16-dimensional vector, as shown in Table I. The maximum and minimum speed and acceleration values are not used to describe the driving styles as it is found that these signals are usually similar within the candidates, which lacks of the representative to reflect the driving styles.

TABLE II
COMPARISON OF KEY PARAMETERS OF DIFFERENT DRIVING STYLES AND ENERGY CONSUMPTION

Styles	Speed STD	Speed Mean	Acc STD	Acc Mean	Jerk Max	Jerk Min	Jerk STD	Jerk Mean
Conservative	3.607	30.808	5.122	3.855	9.635	-7.845	3.394	-0.069
Moderate	2.844	31.747	5.315	4.006	9.266	-8.486	3.618	-0.078
Aggressive	1.580	33.823	4.242	2.970	7.318	5.862	2.818	-0.066
Styles	SH Max	SH Min	SH STD	SH Mean	TH Max	TH Min	TH STD	TH Mean
Conservative	39.240	23.420	4.356	31.403	6.432	2.516	0.995	3.804
Moderate	27.971	23.544	1.395	25.745	3.373	2.406	0.288	2.822
Aggressive	19.851	17.586	0.713	61.386	18.715	1.704	0.098	1.855

The rear vehicle will take three seconds to observe the dynamics of its leading vehicle and generate the corresponding style label based on the GMM. Hence, the GMM is trained with the statistical values of the above feature vector in the length of three seconds. An assumption made here is that the driving style for each vehicle will not change during the driving cycle or a certain period. This is normally true in the real-world as the driving style is the reflection of driving habits, which can hardly be changed rapidly and frequently [32]. However, the driving style can change due to some emergency scenarios, such as suddenly cut-in and braking of a leading vehicle, etc. When the predicted trajectories are significantly different with the observed position of the learning vehicle, the algorithm has to re-estimate the driving style and predict the vehicle track with a more proper sub-network.

A statistic for the three driving styles is illustrated in Table II above, all features have the same units with the NGSIM dataset. The three different driving styles are named as conservative, moderate, and aggressive based on the mean velocity, and other dynamics. It should be mentioned that the driving style recognition is not the major task for this task, and the dynamics for the three driving styles can be different from some existing methods. It is only the reflection of the distinctive clustering results of the GMM and it can be defined with other category names. Besides, the name definition of the three groups will not influence further analysis on the personalized track predictions.

From Table II it can be seen that the defined aggressive vehicles have faster speed, smaller space headway, and lower time headway than the other two groups based on the GMM classification, which shows aggressive drivers may tend to maintain a smaller space with front vehicles with faster speed. On the other hand, the acceleration and jerk features do not significantly distinct from each other based on the GMM method except the variation of the acceleration and jerk for the aggressive driver is lower than the rest two groups.

C. Feature Importance Analysis Based on MIC

In this part, the MIC algorithm is used to analyze the mutual dependence between the GMM learned driving styles and the hand-crafted feature vector. By evaluating the mutual association between the specific feature vector and the style labels, the most influential features can be obtained. MIC can measure the linear or nonlinear correlation between two variables by applying an optimal data binning technique to assess the mutual information for continuous variables [33]. The mutual information measures the mutual dependence between two variables,

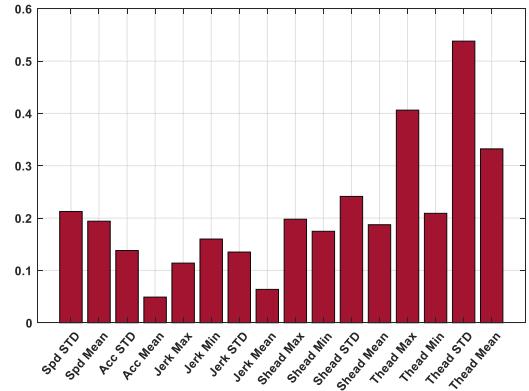


Fig. 4. Importance analysis of the driving style-related features.

which generates the amount of information for one variable by observing another variable. The mutual information for two discrete vectors is defined as:

$$MI_D(X, Y) = \sum_{y \in Y} \sum_{x \in X} p(x, y) \log \left(\frac{p(x, y)}{p(x)p(y)} \right) \quad (4)$$

where MI_D is the mutual information of two discrete vectors, $p(x, y)$ is the joint probabilistic mass function of x and y . $p(x)$ and $p(y)$ are the marginal mass function of x and y . Similarly, for continuous variables, the mutual information format is described as:

$$MI_C(X, Y) = \int \int p(x, y) \log \left(\frac{p(x, y)}{p(x)p(y)} \right) dx dy \quad (5)$$

where MI_C is the mutual information for the two continuous vectors, and $p(x, y)$, $p(x)$, and $p(y)$ represent the corresponding joint probabilistic density and marginal probabilistic density functions, respectively.

It is not always easy to access the mutual dependency between two continuous variables by directly calculating the probabilistic mass function. Thus, MIC was designed to solve this problem by introducing a maximal mutual information searching technique and optimal data binning method [34]. Meanwhile, MIC enables the mutual information score to be normalized into the range between zero and one, which makes it more convenient to assess the dependency and co-relationship between two variables.

Based on the MIC algorithm, the co-relationship between the features and the driving style label can be generated. The results are shown in Fig. 4. The feature with the most massive value indicates the most importance and co-related properties. As shown in Fig. 4, the three most important features are the standard deviation value for the time headway, maximum of

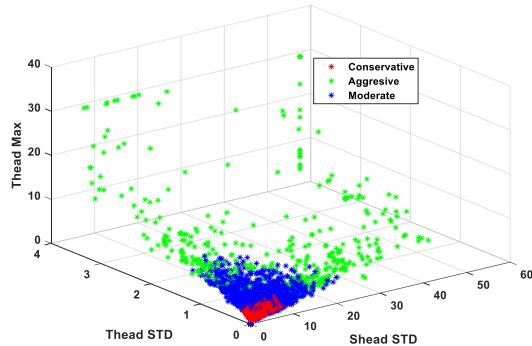


Fig. 5. Visualization of the distribution for the three driving styles concerning time headway STD, time headway Max, and space headway STD.

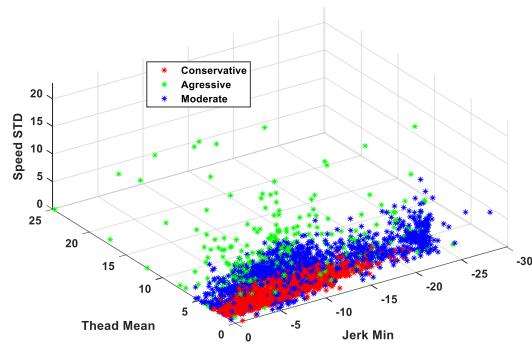


Fig. 6. Visualization of the distribution for the three driving styles concerning time headway means, speed STD, and Jerk Min.

time headway, and meantime headway. It shows that the time headway of the leading vehicle is an important feature to evaluate the driving styles.

Based on the analysis of feature importance, the different driving styles are visualized based on the six most important features. The visualization for different driving styles is shown in Fig. 5 and Fig. 6, respectively. As shown in the two figures, the three driving styles show different significant distributions in the selected domain. The MIC module can be used to calculate the feature importance so that more important and representative features can be selected to increase the precise and computational efficiency. In the real-world application, the most distinctive features are usually not available and can be different in different situations. For example, if the V2V communication is not allowed, only the estimated speed and acceleration of the learning vehicle are available. If the feature importance results show that the importance of space/time headway is similar to the velocity and acceleration, a more robust and transferable system can be designed based on the easily acceptable features.

IV. JOINT FEATURE LEARNING AND PERSONALIZED TRAJECTORY PREDICTION

In this section, the recurrent neural network and the Long Short-Term Memory unit basics are introduced. The RNN model is used to learn the temporal dependency within the time sequence data, while the LSTM increases the performance of the RNN model by adding the memory and forget gate unit

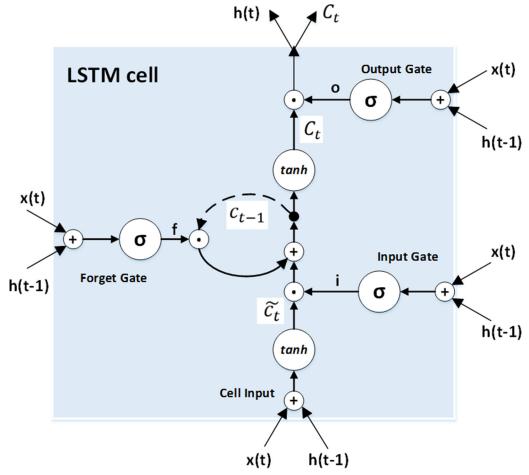


Fig. 7. LSTM cell structure.

to control the information flow. The personalized JTSM model concatenates the shared LSTM layer and personalized regression layer together.

A. Recurrent Neural Network and LSTM

The LSTM based RNN is applied in this study to process the sequential trajectory data and generate a future track prediction. RNN exhibits the dynamic temporal pattern of a sequence of data by forming a directed connection between previous states and the current state [35]–[37]. Current state stores the information of the previous hidden states and the past inputs. A major problem of vanilla RNN is the so-called gradient vanishing or exploding, which makes RNN unable to capture the long-term dependency and information for the sequential data [38].

Hochreiter and Schmidhuber developed the LSTM cell to overcome the drawbacks of RNN [39]. The LSTM-RNN solves the long-term dependency problem by introducing three extra gates, known as the input gate, forget gate, and output gate. The gates cooperate to control the information flow. Fig. 7. shows the LSTM architecture [40]. The chain-like LSTM RNN model replaces the hidden states with the LSTM cell.

The forget gate in the LSTM cell controls what information to throw away while the input gate controls what new information to be updated and stored. The output gate controls the candidate layer output.

$$f_t = \sigma(\mathbf{U}_f x_t + \mathbf{W}_f h_{t-1} + b_f) \quad (6)$$

$$i_t = \sigma(\mathbf{U}_i x_t + \mathbf{W}_i h_{t-1} + b_i) \quad (7)$$

$$o_t = \sigma(\mathbf{U}_o x_t + \mathbf{W}_o h_{t-1} + b_o) \quad (8)$$

The value \tilde{c}_t is the candidate cell state which can be represented as:

$$\tilde{c}_t = \tanh(\mathbf{U}_c x_t + \mathbf{W}_c h_{t-1} + b_c) \quad (9)$$

The cell state c_t of the LSTM unit is the combination of previous c_{t-1} and current candidate states.

$$C_t = f_t * c_{t-1} + i_t * \tilde{c}_t \quad (10)$$

TABLE III
FEATURE VECTORS FOR THE CONSTRUCTION OF JTSM

Vehicles	Features
Following	Lateral position, Longitudinal position, Speed, Acceleration, Space headway, Time headway
Leading	Lateral position, Longitudinal position, Space, Acceleration

Finally, the cell outputs the products of the cell state C_t and the candidate output from the output gate.

$$h_t = o_t * \tanh(C_t) \quad (11)$$

where σ in the above equations represents the *sigmoid* function. $*$ is the element-wise production, x_t and h_{t-1} , are the current input vector and the previous hidden state. f, i, o are the forget gate, input gate, and output gate. U, W, b are corresponding model parameters.

B. JTSM Model Construction

The JTSM model is trained based on the following-leading pairs extracted from the NGSIM dataset. The data contains the trajectory data from lane one to lane six of the US-101 and I-80 freeway. There are 5574 following-leading pairs used in this study. The sequential data contain the six follow vehicle features and four leading vehicle features. The features used in this study are shown in Table III. It should be noticed that the features used for trajectory prediction are not necessarily the same with the GMM clustering shown in Table I as trajectory prediction should be determined in each step, and fewer features can improve the real-time computation efficiency. While the GMM-based driving style recognition does no need to be calculated frequently as the driving style does not change rapidly. The driving style can be updated if the predicted trajectories show a significant cumulative error along with time, which means the previous estimation of the driving style is not representative in the current stage.

The sequential feature extraction layer of JTSM is constructed with LSTM cells, while the personalized trajectory prediction layer contains three different fully-connected (FC) regression neural networks. The training objective of the customized JTSM is to estimate the leading vehicle trajectories based on the previous paths and the easily accessible vehicle states of the leading vehicle based on real-time communication. The trajectory is defined by the longitudinal and lateral position of the leading vehicle. The lateral position follows the same format of NGSIM dataset, while the longitudinal position is the relative longitudinal distance between the leading and following vehicles. The common temporal pattern extraction layer based on the LSTM is first trained with the whole set of training data to increase the diversity of the state. Then, according to the different driving styles, different FC layers are trained and fine-tuned separately with the common LSTM layer based on transfer learning. The whole leading-following pairs are used to generate the driving style categories.

Regarding the construction of the JTSM model, 80% of the data are randomly selected as the training data, and the rest 20%

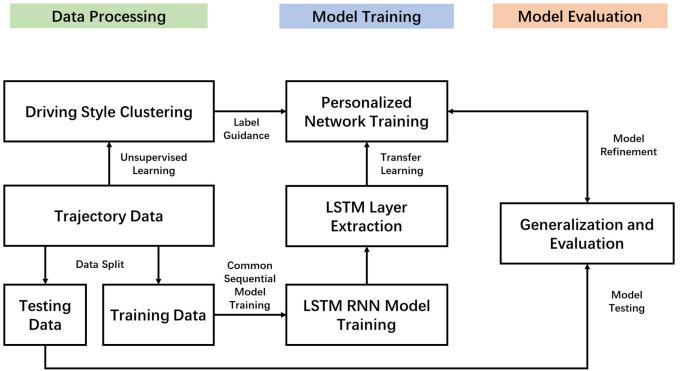


Fig. 8. Personalized JTSM training process.

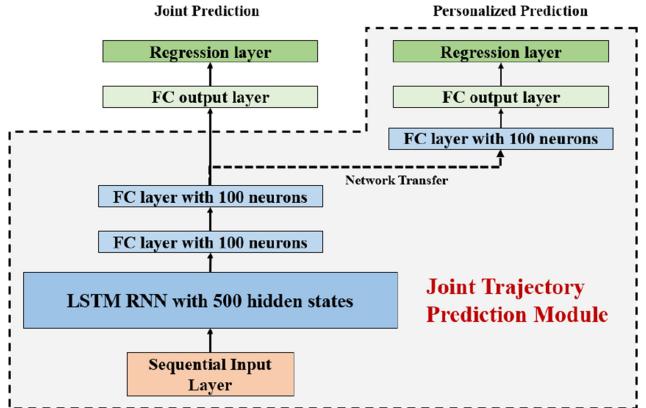


Fig. 9. Model structure of the personalized JTSM network.

are used for model testing. Among the 4457-training data, 1644 cases are classified into the moderate group, 2429 cases belong to the aggressive group, and the rest 384 samples are classified into the conservative group. For the third group model training, the 362 samples are too few to train a precise model. Therefore, data augmentation is applied to the third group, which uses a random sampling method with a random sampling rate selected between 0.75 to 0.9 for each iteration. For each sequence, the sampling unit randomly generates a number within the range of the sampling rate and randomly extracts the data to form a sub-sequence of the original series. The augmentation is taken three times, and the final data volume of the third group is four times of the original data. The whole networks are developed with MATLAB 2019a. A detailed training and testing process are illustrated in Fig. 8 above. The testing data are only used for model testing and evaluation to prevent data leakage issues.

The model structure is shown in Fig. 9. The left part is the joint prediction module, which uses all the training data to generate the LSTM sequential layer and two fully connected layers without considering the different driving styles. Once the joint time-series model is built, this part will be used to construct the personalized models based on the sub-training dataset according to the driving styles. The joint model is trained with stochastic gradient descent (SGD) with a learning rate of 0.01. The max epoch is 150, and the mini-batch size is 32. Regarding the personalized models, the first four layers of the joint model are adopted with an initial learning rate 0.0001 to

slow down the learning in the transfer layer, while the weighted learning rate of the FC layer is 40 to speed up the learning process of the new layers. The max epoch of the personalized networks is 200.

V. EXPERIMENT RESULTS

In this section, the statistical analysis of the trajectory prediction is made based on different evaluation metrics and different prediction horizons.

A. Evaluation Metrics and Baselines

The model prediction results are compared based on multiple metrics. Specifically, the Euclidean distances between the predicted and the ground truth trajectories are evaluated with the root of the mean square error (RMSE) format. The worst 5% and 1% of cases reported for the tracks are also analyzed. Also, the Modified Hausdorff Distance (MHD) method is adopted [41], following with the worst 5% and 1% records for the MHD scores. The general MHD is the mean of the MHD for the lateral position and longitudinal position. The general RMSE measures for the tracks are calculated as:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1,2,\dots,N} ((\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2)} \quad (12)$$

where n is the total number of steps, \hat{x}_i and \hat{y}_i are the lateral and longitudinal values of the predicted trajectory.

Several baselines are used to make a comparison between the existing algorithms and proposed methods. The Baselines include:

- 1) **Constant Kalman Filter (CKF):** The path of the leading vehicle is predicted with a constant velocity Kalman filter model [42].
- 2) **LSTM:** A single LSTM-RNN model is trained to predict future tracks without considering the driving styles.
- 3) **Multiple LSTM (MLSTM) for multiple classes:** Three different LSTM-RNN models are trained based on the driving styles without using the pre-trained common LSTM layer. The training data for each group is used separately to train the models.
- 4) **JTSM:** A joint time-series model based on the shared LSTM layer and further FC regression networks for the different driving styles. First, a common LSTM temporal pattern network is trained. Then, for each driving style, the concatenate networks are fine-tuned.

B. Comparison Between Different Driving Styles

In this part, the multi-steps track prediction is proposed and compared within the baseline algorithms listed above. First, the track prediction results for different driving styles are proposed with a one-second prediction horizon. Then, the predictions that are made with different horizons are proposed. Fig. 10. indicates the trajectory prediction on some exemplary cases for the three different driving styles with a one-second prediction horizon. It is shown that the proposed JTSM method can generate a good trajectory estimation for different kinds of driving styles. It is

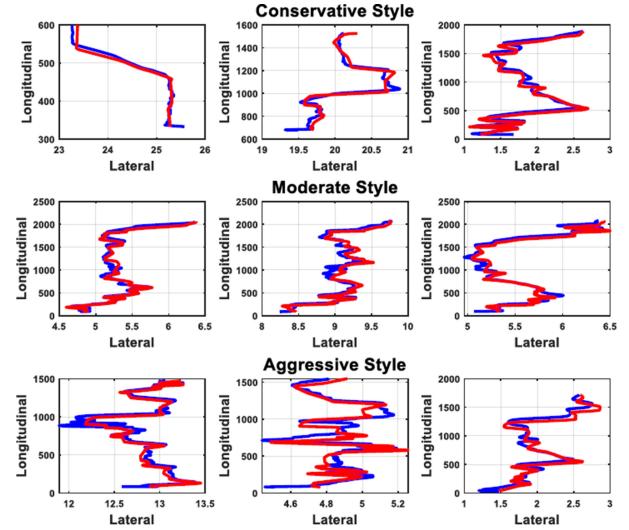


Fig. 10. Track prediction for different driving styles with x-axis and y-axis for the longitudinal and lateral position in meter. The red lines are the target tracks, while the blue lines are the predicted tracks.

TABLE IV
COMPUTING COST FOR DIFFERENT ALGORITHMS

Time Cost	LSTM	MLSTM	JTSM
Training	~1800 s	~2400 s	~3300 s
Testing	~0.022 s	~0.025 s	~0.025 s

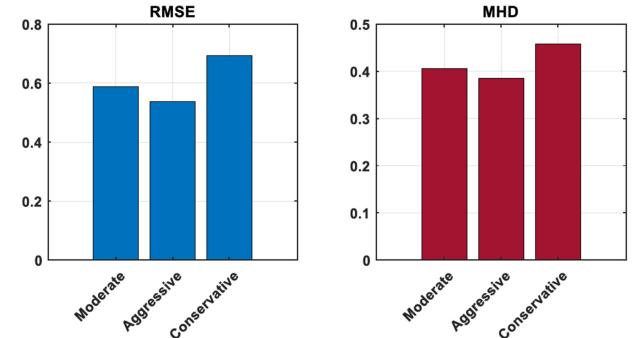


Fig. 11. Prediction errors for different driving styles. The blue bars are the RMSE estimation for the three driving styles, and the red bars are the MHD.

also essential to understand whether different driving styles have a different impact on the estimation precision of future trajectories. Therefore, track prediction results for different driving styles are compared.

The training and testing cost for the three different algorithms are shown in Table IV. The training cost for the proposed JTSM method is the largest one, which mainly due to the data augmentation is applied to the aggressive driving class. Besides, the testing cost for the MLSTM and JTSM is slightly higher than the vanilla LSTM method. The testing rate of the JTSM achieved about 40 fps, which satisfy the real-time computational requirement. All the training and testing algorithms are evaluated with a low-cost NVIDIA MX150 GPU.

As shown in Fig. 11, the RMSE and MHD of the track prediction for different driving styles are illustrated. The prediction

TABLE V
COMPARISON OF THE LATERAL POSITION PREDICTION RESULTS FOR DIFFERENT METHODS

Prediction Horizons	Evaluation Metrics	CKF	LSTM	MLSTM	JTSM
1 s	Eu RMSE	1.419	0.764	0.682	0.568
	Eu 5% Worst	3.604	1.783	1.950	1.661
	Eu 1% Worst	5.595	3.150	3.443	3.029
	MHP Mean	0.058	0.175	0.189	0.132
	MHP 5% Worst	0.281	0.400	0.514	0.322
	MHP 1% Worst	0.505	0.514	0.718	0.469
2 s	Eu RMSE	2.547	1.489	1.569	1.384
	Eu 5% Worst	6.784	3.188	3.615	3.120
	Eu 1% Worst	10.613	4.984	5.516	4.994
	MHP Mean	0.143	0.293	0.310	0.219
	MHP 5% Worst	0.956	0.753	0.910	0.585
	MHP 1% Worst	1.729	0.999	1.356	0.869
3 s	Eu RMSE	3.517	2.376	2.441	2.210
	Eu 5% Worst	9.886	4.684	5.236	4.723
	Eu 1% Worst	15.747	6.871	7.963	7.111
	MHP Mean	0.282	0.465	0.403	0.320
	MHP 5% Worst	2.200	1.280	1.238	1.065
	MHP 1% Worst	4.218	1.795	1.902	1.651
4 s	Eu RMSE	4.367	3.469	3.233	3.026
	Eu 5% Worst	12.899	6.863	6.982	6.465
	Eu 1% Worst	20.796	9.870	9.782	8.915
	MHP Mean	0.498	0.685	0.614	0.467
	MHP 5% Worst	4.235	2.457	2.292	1.855
	MHP 1% Worst	8.594	3.581	3.611	2.870
5 s	Eu RMSE	5.119	3.929	4.015	3.788
	Eu 5% Worst	15.774	8.731	9.142	8.479
	Eu 1% Worst	25.773	12.447	12.981	11.575
	MHP Mean	0.788	0.839	0.749	0.639
	MHP 5% Worst	7.110	3.293	3.369	3.014
	MHP 1% Worst	14.390	5.325	5.520	4.887

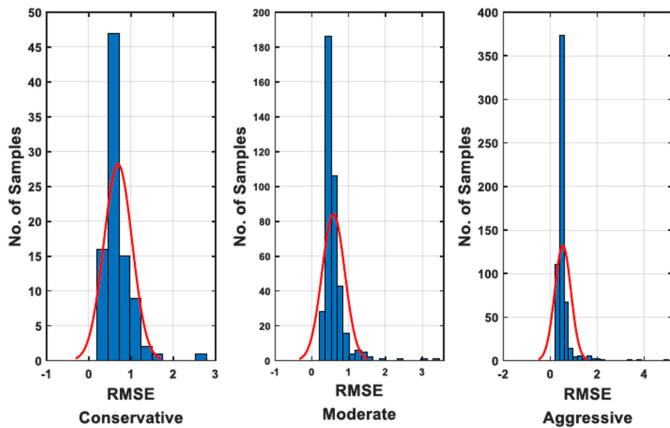


Fig. 12. RMSE histogram with the distribution fit of the three different groups.

errors of the conservative driver achieved the highest. The data quantity of the conservative class is smaller than the rest two sets, which can cause worse prediction results than the other two groups due to over-fitting issues. Also, it can be caused by some unnecessary actions or hesitation for path selection during driving as the conservative driver usually take more time and space to finish the maneuver. The high randomness of the conservative driving style can make it difficult to estimate the vehicle trajectory precisely. The histogram distribution of the three groups is illustrated in Fig. 12. The prediction error distribution of the three groups can be represented with a normal

distribution. The prediction error of the aggressive drivers is more consistent than the rest two groups, which show a more precise track prediction of this class.

C. Multi-Steps Ahead Trajectory Prediction

In this part, the comparison between multiple algorithms with multiple prediction horizons of the tracks is proposed. The main results are shown in Table V. The mean Euclidean distance-based RMSE value and the mean Modified Hausdorff distance are calculated. Also, the worst 5% and worst 1% cases for the track prediction are recorded to give a detailed analysis of the proposed method. The RMSE is transformed from feet to meters for easier comparison with other studies. The RMSE for the MLSTM and JTSM methods are calculated as the average RMSE of the three classes.

As shown in Table V, the proposed JTSM method achieved the most accurate results than the other methods. It is shown that the joint temporal pattern extraction layer along with the personalized prediction layer achieved more precise prediction than the separate LSTM networks. The CKF methods can generate reasonable estimation results within two seconds; however, the prediction ability the worse among the four methods. Another interesting finding is the performance of the LSTM and MLSTM do not show significant differences with each other. Separately training different LSTM models with sub-training data group is not an efficient solution for improving the model performance since each class is trained with fewer data with less temporal patterns.

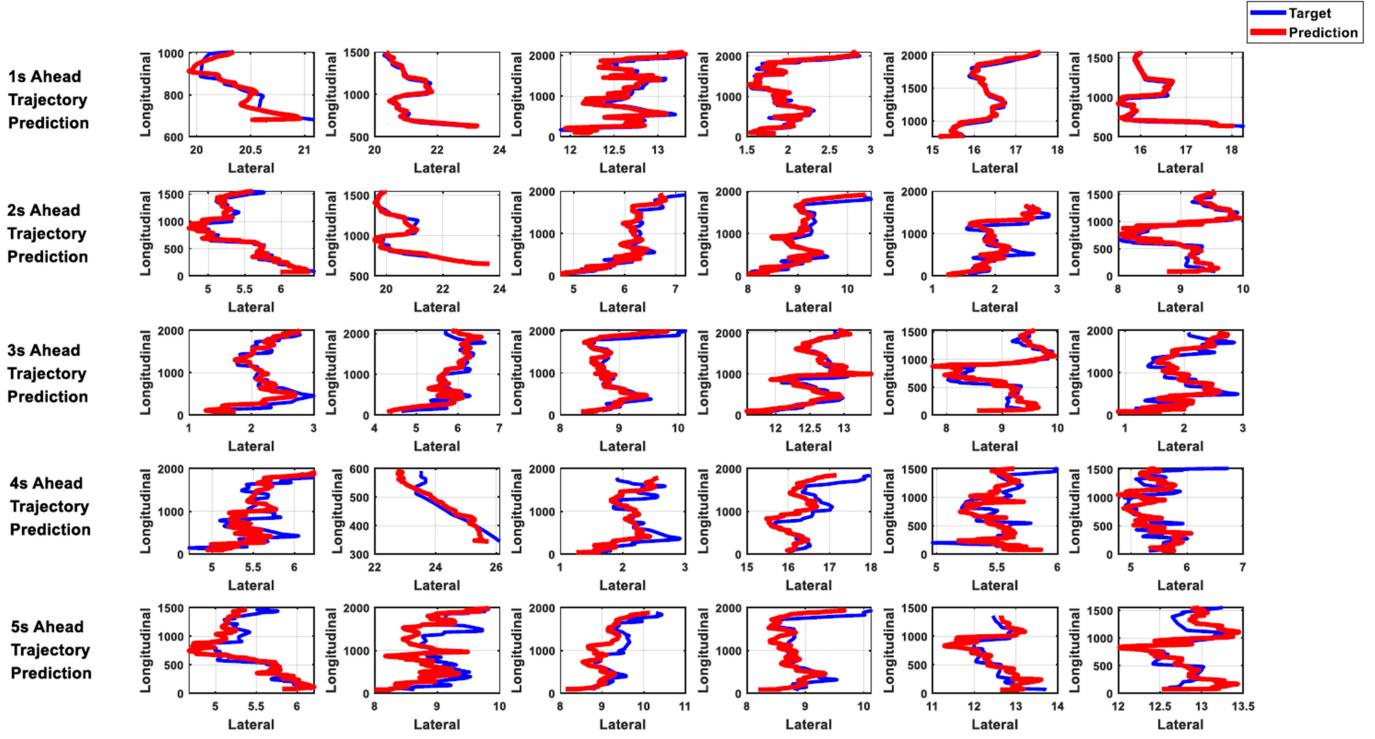


Fig. 13. Exemplar track predictions of the JTSM method with multiple prediction horizons x-axis and y-axis for the longitudinal and lateral position in meter, the left two most columns are the prediction for the moderate styles, the middle two columns are the aggressive styles, while the right two columns are case examples for conservative styles. The red lines are the target tracks, while the blue lines are the predicted tracks.

An illustration of the track predictions with different prediction horizons is shown in Fig. 13. It is shown that the JTSM method can achieve a precise prediction within three seconds. The predictions that are made with four seconds and five seconds prediction horizon will not follow the target trajectory very well. The RMSE for the four seconds and five seconds ahead prediction are over three meters, which is a relatively large prediction error in the real-world scenarios. Hence, the prediction horizon within three seconds is more suitable for the personalized track prediction. Based on the results statistics, it can be found that the proposed JTSM algorithm is more precise than the vanilla and separate multiple LSTM methods, which shows the effectiveness of both the driving style based personalized track prediction and the joint sequential pattern learning.

D. Results Discussion

The model performance on the leading vehicle trajectory prediction is analyzed and compared in the last two parts of this study. The results indicate that the shared temporal pattern learning network based on the LSTM layer and the personalized fully-connected prediction layers can improve the prediction accuracy. The model achieved consistent and better results compared with the study in [21] and [22]. It shows that integrating driving style into the model design can generate a more precise track prediction model for the leading vehicles. Although reasonable estimation for the track prediction is achieved, we can estimate the limitation of the proposed method into the

following aspects. First, the GMM model is designed based on a set of easy acceptable data through V2V communication. A more comprehensive driving style recognition model can be created based on more vehicle dynamics such as the observed lateral/longitudinal tactical behaviors of the leading vehicle. The current model only based on longitudinal features without using lateral behaviors where the representative of the driving styles recognition model can be further improved. Second, only the leading vehicle track is predicted in this study. A more holistic approach for interaction-aware and driving style-aware trajectory prediction model is expected to be designed in the future so that to construct an informative traffic context-aware system.

VI. CONCLUSION

In this study, a personalized leading vehicle trajectory prediction based on joint time series modeling is proposed for connected vehicles. By analyzing the leading vehicle status through V2V communication and driving style recognition, the future trajectories prediction of the surrounding vehicle can be improved. The personalized JTSM algorithm generates three different driving styles for the leading vehicle; then, a joint LSTM temporal pattern extraction layer is used by the three different track prediction networks. With the transfer learning-based JTSM algorithm, the track prediction achieved better results than the separate multiple networks and single LSTM based methods. The track prediction for the leading vehicle

achieved a 0.53 m RMSE, which shows the effectiveness of the proposed personalized JTSM method. Results indicate that predict the future vehicle status based on the different groups is more precise than predict the state with a uniform model.

This study focused on the leading and following vehicle interaction. The main objective of this study is to show the impact of the personalized joint time-series modeling on the trajectory prediction for connected vehicles. The personalized track prediction is believed to be also effective when considering multiple surrounding vehicle interaction. Future works will concentrate on the construction of personalized status estimation for multiple-agents based interaction-aware models for the large-scale transportation system.

REFERENCES

- [1] J. Harding *et al.*, “Vehicle-to-vehicle communications: Readiness of V2V technology for application,” National Highway Traffic Safety Administration, Washington, DC, USA, DOT HS 812 014, 2014.
- [2] B. Zhao *et al.*, “Driving-cycle-aware energy management of hybrid electric vehicles using a three-dimensional Markov chain model,” *Automot. Innov.*, vol. 2, pp. 146–156, 2019.
- [3] Z. Wanzhong, W. Gang, W. Chunyan, Y. Leiyang, and L. Yufang, “Energy transfer and utilization efficiency of regenerative braking with hybrid energy storage system,” *J. Power Sources*, vol. 427, pp. 174–183, 2019.
- [4] S. I. Guler, M. Menendez, and L. Meier, “Using connected vehicle technology to improve the efficiency of intersections,” *Transp. Res. Part C, Emerg. Technol.*, vol. 46, pp. 121–131, 2014.
- [5] J. E. Siegel, D. C. Erb, and S. E. Sarma, “A survey of the connected vehicle landscape—Architectures, enabling technologies, applications, and development areas,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2391–2406, Aug. 2018.
- [6] S. R. K. Narla, “The evolution of connected vehicle technology: From smart drivers to smart cars to ... self-driving cars,” *ITE J.*, vol. 83, no. 7, pp. 22–26, 2013.
- [7] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, “Connected vehicles: Solutions and challenges,” *IEEE Int. Things J.*, vol. 1, no. 4, pp. 289–299, Aug. 2014.
- [8] S. Lefèvre, D. Vasquez, and C. Laugier, “A survey on motion prediction and risk assessment for intelligent vehicles,” *ROBOMECH J.*, vol. 1, no. 1, pp. 1–14, 2014.
- [9] Z. Wanzhong and Z. Han, “Coupling control strategy of force and displacement for electric differential power steering system of electric vehicle with motorized wheels,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8118–8128, Sep. 2018.
- [10] H. Guo, C. Shen, H. Zhang, H. Chen, and R. Jia, “Simultaneous trajectory planning and tracking using an MPC method for cyber-physical systems: A case study of obstacle avoidance for an intelligent vehicle,” *IEEE Trans. Ind. Inf.*, vol. 14, no. 9, pp. 4273–4283, Sep. 2018.
- [11] C. Lv *et al.*, “Hybrid-learning-based classification and quantitative inference of driver braking intensity of an electrified vehicle,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5718–5729, Jul. 2018.
- [12] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, and F. Wang, “Driver activity recognition for intelligent vehicles: A deep learning approach,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5379–5390, Jun. 2019.
- [13] M. Bahram *et al.*, “A combined model-and learning-based framework for interaction-aware maneuver prediction,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 6, pp. 1538–1550, Jun. 2016.
- [14] N. Deo and M. Trivedi Mohan, “Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTM,” in *Proc. IEEE Intell. Vehicles Symp.*, 2018, pp. 1179–1184.
- [15] M. Schreier, V. Willert, and J. Adamy, “An integrated approach to maneuver-based trajectory prediction and criticality assessment in arbitrary road environments,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2751–2766, Oct. 2016.
- [16] G. Agamennoni, J. I. Nieto, and E. M. Nebot, “Estimation of multivehicle dynamics by considering contextual information,” *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 855–870, Aug. 2012.
- [17] M. Bahram *et al.*, “A game-theoretic approach to replanning-aware interactive scene prediction and planning,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 3981–3992, Jun. 2016.
- [18] J. Li *et al.*, “A dynamic Bayesian network for vehicle maneuver prediction in highway driving scenarios: Framework and verification,” *Electronics*, vol. 8, no. 1, pp. 40–59, 2019.
- [19] Y. Luo and P. Cai, “GAMMA: A general agent motion prediction model for autonomous driving,” 2019, *arXiv:1906.01566*.
- [20] A. Alahi *et al.*, “Social LSTM: Human trajectory prediction in crowded spaces,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 961–971.
- [21] N. Deo and M. Mohan Trivedi, “Convolutional social pooling for vehicle trajectory prediction,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, Salt Lake City, UT, USA, 2018, pp. 1549–15498.
- [22] T. Zhao *et al.*, “Multi-agent tensor fusion for contextual trajectory prediction,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 12126–12134.
- [23] A. Zyner, S. Worrall, and E. Nebot, “Naturalistic driver intention and path prediction using recurrent neural networks,” *IEEE Trans. Intell. Transp. Syst.*, to be published, doi: [10.1109/TITS.2019.2913166](https://doi.org/10.1109/TITS.2019.2913166).
- [24] A. Kuefeler, J. Morton, T. Wheeler, and M. Kochenderfer, “Imitating driver behavior with generative adversarial networks,” in *Proc. IEEE Intell. Vehicles Symp.*, Los Angeles, CA, USA, 2017, pp. 204–211.
- [25] C. Ju *et al.*, “Interaction-aware Kalman neural networks for trajectory prediction,” 2019, *arXiv:1902.10928*.
- [26] R. Zhang *et al.*, “A method for connected vehicle trajectory prediction and collision warning algorithm based on V2V communication,” *Int. J. Crashworthiness*, vol. 22, no. 1, pp. 15–25, 2017.
- [27] B. Coifman and L. Li, “A critical evaluation of the next generation simulation (NGSIM) vehicle trajectory dataset,” *Transp. Res. Part B, Methodological*, vol. 105, pp. 362–377, 2017.
- [28] V. Punzo, M. T. Borzacchiello, and B. Ciuffo, “On the assessment of vehicle trajectory data accuracy and application to the next generation simulation (NGSIM) program data,” *Transp. Res. Part C, Emerg. Technol.*, vol. 19, no. 6, pp. 1243–1262, 2011.
- [29] C. E. Rasmussen, “The infinite Gaussian mixture model,” in *Proc. Advances Neural Inf. Process. Syst.*, 2000, pp. 554–560.
- [30] G. Li, S. E. Li, B. Cheng, and P. Green, “Estimation of driving style in naturalistic highway traffic using maneuver transition probabilities,” *Transp. Res. Part C, Emerg. Technol.*, vol. 74, pp. 113–125, 2017.
- [31] M. Van Ly, S. Martin, and M. M. Trivedi, “Driver classification and driving style recognition using inertial sensors,” in *Proc. IEEE Intell. Vehicles Symp.*, 2013, pp. 1040–1045.
- [32] C. Marina Martinez, M. Heucke, F. Wang, B. Gao, and D. Cao, “Driving style recognition for intelligent vehicle control and advanced driver assistance: A survey,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 666–676, Mar. 2018.
- [33] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 2012.
- [34] Y. Xing *et al.*, “Driver lane change intention inference for intelligent vehicles: Framework, survey, and challenges,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4377–4390, May 2019.
- [35] D. N. Reshef *et al.*, “Detecting novel associations in large data sets,” *Science*, vol. 334, no. 6062, pp. 1518–1524, 2011.
- [36] Y. Xing and C. Lv, “Dynamic state estimation for the advanced brake system of electric vehicles by using deep recurrent neural networks,” *IEEE Trans. Ind. Electron.*, to be published, doi: [10.1109/TIE.2019.2952807](https://doi.org/10.1109/TIE.2019.2952807).
- [37] J. Morton, T. A. Wheeler, and M. J. Kochenderfer, “Analysis of recurrent neural networks for probabilistic modeling of driver behavior,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1289–1298, May 2017.
- [38] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
- [39] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [40] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A search space odyssey,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2016.
- [41] M.-P. Dubuisson and A. K. Jain, “A modified Hausdorff distance for object matching,” in *Proc. 12th Int. Conf. Pattern Recognit.*, 1994, vol. 1, pp. 566–568.
- [42] S. E. Li *et al.*, “Kalman filter-based tracking of moving objects using linear ultrasonic sensor array for road vehicles,” *Mech. Syst. Signal Process.*, vol. 98, pp. 173–189, 2018.



Yang Xing received the Ph.D. degree from Cranfield University, England, U.K., in 2018. He is currently a Research Fellow with the Department of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore. His research interests include machine learning, driver behavior modeling, intelligent multi-agent collaboration, and intelligent/autonomous vehicles. His work focuses on the understanding of driver behaviors using machine-learning methods and intelligent and automated vehicle design. He was the recipient of the IV2018 Best Workshop/Special Issue Paper Award. Dr. Xing was a Guest Editor for the IEEE INTERNET OF THING, and he is an Active Reviewer for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, and IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS.



Chen Lv received the Ph.D. degree from Tsinghua University, Beijing, China, in 2016. He is currently an Assistant Professor with Nanyang Technology University, Singapore. From 2014 to 2015, he was a joint Ph.D. Researcher with EECS Department, University of California, Berkeley. His research focuses on cyber-physical system, hybrid system, advanced vehicle control and intelligence, where he has contributed more than 40 papers and obtained 11 granted China patents. Dr. Lv was a Guest Editor for the IEEE/ASME TRANSACTIONS ON MECHATRONICS, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, and *International Journal of Powertrains*, and an Associate Editor for *International Journal of Electric and Hybrid Vehicles*, *International Journal of Vehicle Systems Modelling and Testing*, *International Journal of Science and Engineering for Smart Vehicles*, and *Journal of Advances in Vehicle Engineering*. He received the Highly Commended Paper Award of IMechE UK in 2012, the National Fellowship for Doctoral Student in 2013, the NSK Outstanding Mechanical Engineering Paper Award in 2014, the Tsinghua University Graduate Student Academic Rising Star Nomination Award in 2015, the China SAE Outstanding Paper Award in 2015, the 1st Class Award of China Automotive Industry Scientific and Technological Invention in 2015, and the Tsinghua University Outstanding Doctoral Thesis Award in 2016.



Dongpu Cao received the Ph.D. degree from Concordia University, Montreal, QC, Canada, in 2008. He is currently an Associate Professor and Director of Driver Cognition and Automated Driving (DC-Auto) Lab, University of Waterloo, Waterloo, ON, Canada. His research focuses on vehicle dynamics and control, driver cognition, automated driving and parallel driving, where he has contributed more than 150 publications and 1 US patent. He was the recipient of the ASME AVTT'2010 Best Paper Award and 2012 SAE Arch T. Colwell Merit Award. Dr. Cao

was an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE/ASME TRANSACTIONS ON MECHATRONICS, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, and *ASME Journal of Dynamic Systems, Measurement and Control*. He has been a Guest Editor for *Vehicle System Dynamics*, and IEEE TRANSACTIONS ON SMC: SYSTEMS. He has been serving on the SAE International Vehicle Dynamics Standards Committee and a few ASME, SAE, IEEE technical committees, and was a Co-Chair of IEEE ITSS Technical Committee on Cooperative Driving.