Development and Evaluation of Two Learning-Based Personalized Driver Models for Pure Pursuit Path-Tracking Behaviors

Zirui Li, Boyang Wang, Jianwei Gong, Tianyun Gao, Chao Lu, Gang Wang

▶ **To cite this version:**

HAL Id: lirmm-03124363

https://hal-lirmm.ccsd.cnrs.fr/lirmm-03124363v1

Submitted on 28 Jan 2021

Development and Evaluation of Two Learning-Based Personalized Driver Models for Pure Pursuit Path-Tracking Behaviors

Zirui Li[1], Boyang Wang[1,2], Jianwei Gong[1] *Member, IEEE,* Tianyun Gao[1], Chao Lu[1] and Gang Wang[3]

Abstract— Establishing a personalized driver model to predict the driving behavior plays a significant role in the promotion of driver assistance system and automated driving system. In this paper, we propose two different learning-based path-tracking personalized driver models to predict the lookahead distance based on pure pursuit algorithm using the naturalistic driving data collected from BIT intelligent vehicle platform. Based on Gaussian Mixture Model (GMM), one stochastic driver model is the velocity-based Gaussian Mixture Regression (GMR) approach established by combining Gaussian classification process and Gaussian mixture regression, while another driver model is the general GMR approach. The predicting results obtained from the stochastic models are analyzed based on the numbers of GMM components. Statistical analyses show that both personalized driver models perform well, and the velocity-based GMR approach demonstrate higher accuracy than general GMR approach in predicting lookahead distance with the preferred number of the GMM components 10-12 and better performance in tracking the given path.

I. INTRODUCTION

Comprehending human driver behavior and development of coupled driver-vehicle system by establishing the personalized driver model is critical for the advanced driver assistance systems(ADAS) [1, 2], decision making [3, 4], automated driving system [5] and human-centered vehicle control systems [6-8]. For instance, [2] proposed that personalized driver model combining the characteristics of an individual driver/vehicle can perform better in driver assistance system for lane change scenario. Driver assistance system developed by learning from drivers' preferences and characteristics improves the fuel economy and reduces waiting time [3]. [9] developed the personalized driver model based on identify maneuvers performed by drivers to provide personalized driving assistance, which develops the widespread of LKA system and ACC system. All examples above indicates that establishing personalized driver model could make intelligent vehicles more efficient and less intrusive.

On the whole, the methods used to modeling the human driver are divided into two categories: physical-based and learning-based. As for the physical-based model, the unknown parameters in formulations are used to model the structure of drivers' driving characteristics in path-tracking, car following, lane change, overtaking. Most physical-based parameters have their own meanings, which can be translated with ease. For example, economy-oriented headway control algorithm [10] and optimal velocity model (OVM) [11] are frequently used in automated vehicle (AV) evaluation and vehicle control [12]. Although the physical-based model can express the physical meanings in some scenarios, the uncertainties and nonlinearities of the driver-vehicle-road system are hard to be fully involved in the driver model. Fortunately, learning-based driver models have been developed to overcome the problems of physical-based method [13]. Some familiar approaches are proposed to improve the learning-based models such as Stochastic Switched AutoRegressive eXogenous model (SS-ARX) [14, 15], Hidden Markov Model [3], Neural network [16, 17] and Gaussian Process [18]. But the learning-based approaches are insufficient to describe the physical meanings of parameters with a highly nonlinear function after modeling the drivers' behavior [18]. Besides, some researchers have developed the models combining the physical-based model and learning-based model. [1] proposed a two layers driver model, in which the physical-based method is used to describe vehicle's kinematics (sinusoidal lane change model) and learning-based method is developed to describe the distribution of physical parameters by Gaussian Mixture Model.

The above mentioned studies demonstrate the contribution of GMM methods due to its ability to describe the uncertainty of the driver-vehicle-road system [1] and integrate with traditional control methods [18]. However, numerous research have focused on the driver data obtained from simulation methods to establish the driver model [4, 6, 7] but very few research pays close attention on both naturalistic driving data and the impacts of different GMM-based approaches. Moreover, most of these application are not concerned with systematic evaluation and the comparison of different GMM-based approaches.

In this paper, two learning-based approaches are proposed for pure pursuit path-tracking scenario [19], in which the influence of different numbers of GMM component on model's performance is analyzed. In addition, we have established a learning-based personalized driver model for pure pursuit path-tracking scenario based on naturalistic driving data. This paper provides the systematic evaluation and the comparison of two learning-based approaches for modeling the pure pursuit path-tracking behavior, and sheds light on the impacts

[1]Zirui Li, Boyang Wang, Jianwei Gong, Chao Lu and Tianyun Gao are with the Intelligent Vehicle Research Center, Beijing Institute of Technology, Beijing, 10081, China. (Corresponding author: Jianwei Gong). `gongjianwei@bit.edu.cn`

[2]Boyang Wang is also with the Interactive Digital Human group of CNRS-UM LIRMM, UMR5506, Montpellier, France, 34095. `wbythink@hotmail.com`

[3]Gang Wang is with Department of Mechanical and Automation Engineering, Chinese University of Hong Kong. `gang.wang@siat.ac.cn`

of GMM components in the establishment of a pensonalized pure pursuit path tracking driver model.

This paper is organized as follows. Section II shows the formulation of pure pursuit path tracking driver model. Section III presents the essential methods to develop the personalized driver model. Section IV shows the data collection system and model training approach of two different learning-based driver models. Evaluation of two models and analysis of the model prediction results are given in Section V. Finally, in section VI we summarize our current work and propose the future work accordingly.

II. PROBLEM FORMULATION

A. Personalized Driver Model

In this paper, we propose to investigate the personalized driver model for pure pursuit behaviors. A personalized pure pursuit driver model can be referred to a model that can promote or predict a personalized lookahead distance according to the desired path and velocity with the same environment situation.
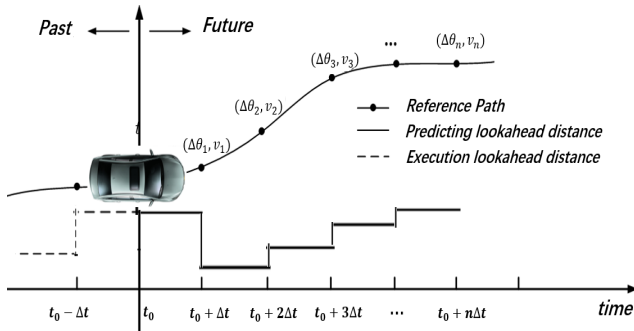

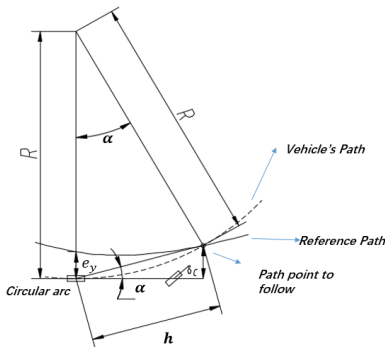Fig. 1. Pure pursuit path tracking scenario

Fig. 2. Pure pursuit algorithm

B. Pure pursuit path tracking Scenario

The Path Tracking Scenario based on pure pursuit algorithm can be demonstrated by Fig.1. The pure pursuit algorithm is geometrically calculates the steering angle based on the α between the heading angle of vehicle and look ahead vector. In the Ackerman two degree of vehicle model, the steering angle δ are given below:

$$\delta = tan^{-1}(\frac{2sin(\alpha)L}{h}) \qquad (1)$$

where h is lookahead distance and L is wheel base. The goal path point is determined by the look ahead distance. The desired lookahead distance is determined by the reference path point. The following variables are defined to represent the path point.

- $\boldsymbol{\xi}_p = [\theta_t, v_t]^T \in \mathbb{R}^{2 \times 1}$ is the state parameter of the path point at time t, in which v_t is the velocity of the path point, θ_t is the course angle of the path point at time t.
- $\boldsymbol{O}_t = [\Delta\theta_t, v_t]^T \in \mathbb{R}^{2 \times 1}$ is another parametric definition of the path point at time t. The difference is that is the course deviation between the time t and $t - 1$.

In the offline model training process, the lookahead distance need to be obtained as the input sequences of training data. But the optimal lookahead distance for each time t can not be obtained directly from original driving data. We use following method to get driver's lookahead distance $h(t)_{optimal}$ based on distance $d_{t+1:t+N}$ between forward path point and current path point.

$$h(t)_{optimal} = P(d_{t+1:t+N}, \delta_t) \qquad (2)$$

$$P(d_{t+1:t+N}, \delta_t) : min\left\{\delta - \delta_0\left(d_{t+n}\right)\right\} \to d_{t+n}, n = 1, 2, ..., N \qquad (3)$$

with $\delta_0\left(d_{t+n}\right)$ is the steering angle calculated by pure pursuit algorithm based on d_{t+n}. The driver data obtained from the vehicle is defined as accurate training data, and each time t corresponds a desired lookahead distance or goal path point by equation (2-3) which is calculated based on the whole forward path information after the current path point.

After the driver model is trained, at time t, the model can generate a predicted lookahead distance. The general process of the proposed pure pursuit driver model is presented as

$$\lambda(\boldsymbol{O}_{1:t}, h_{1:t}) : \boldsymbol{O}_{t_0:t_0+\Delta t} \mapsto \boldsymbol{h}_{t_0:t_0+\Delta t} \qquad (4)$$

Based on the λ model, the equation (3) is to generate the predicted lookahead distance sequences $\boldsymbol{h}_{t_0:t_0+\Delta t}$ for pure pursuit algorithm according to the history vehicle data, where $n \in \mathrm{N}^*$, $\Delta t = 0.1s$.

III. METHODS

In this section, two statistical learning approaches are presented to model a personalized driver pure pursuit behavior (i.e., the Gaussian Mixture Model (GMM) and Gaussian Mixture Regression (GMR)). The GMM and GMR are separately discussed in the following sections.

A. Gaussian Mixture Model

In order to describe the probability density function of course deviation, velocity and decision of look ahead distance, the Gaussian Mixture Models are used. multivariate Gaussian Mixture Models is represented as

$$P(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{i=1}^{k} \pi_i \frac{1}{(2\pi)^{d/2} \left|\boldsymbol{\Sigma}_i\right|^{1/2}}$$
$$\times \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_i)\right] \qquad (5)$$

where $A = \boldsymbol{x} - \boldsymbol{\mu}_i$, $\boldsymbol{\theta} = \{\boldsymbol{\theta}\}_{i=1}^n$, $\boldsymbol{\theta}_i = \{\pi_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}$ and $\boldsymbol{x}$ is a set of d-dimension sequence, $\boldsymbol{x} = \{\boldsymbol{x}\}_{i=1}^n$ with $\boldsymbol{x}_i \in \mathbb{R}^{d \times 1}$, which can be calculated in a combination of N Gaussian models. $\boldsymbol{\mu}_i \in \mathbb{R}^{d \times 1}$ and $\boldsymbol{\Sigma}_i \in \mathbb{R}^{d \times d}$ are mean and covariance vector of ith single Gaussian Model, π_i is the prior probability with $\Sigma_{i=1}^n \pi_i = 1$.

For the pure pursuit model, the joint distribution between O_t and h_t can be rewritten as

$$P(\boldsymbol{O}_t, h_t, \boldsymbol{\theta}) \sim \sum_{i=1}^N \pi_i N_i(\boldsymbol{O}_t, h_t, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \quad (6)$$

In order to estimate the parameters $\boldsymbol{\theta}$ of the GMM, the expectation maximization (EM) algorithm are used. The maximum iteration steps are set in advance. to gain the estimated optimal parameters until the likelihood function value meet the maximum or convergent.

The optimal objection for the log-likelihood function is shown as:

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \arg \max_{\boldsymbol{\theta}} \log(G(\boldsymbol{x}; \boldsymbol{\theta})) \quad (7)$$

Similar to the k-means clustering method, an unavoidable problem with GMM is to determine the optimal number of Gaussian Mixture Model. So Bayesian information criterion (BIC) is used to determine the optimal number of the Gaussian Mixture Model.

$$BIC(\boldsymbol{x}|\tau) \approx \Sigma_{i=1}^n \log G(x_i) - \frac{1}{2}(2k + kd) \log n \quad (8)$$

The cluster label of data x_i being in Gaussian Model is represented as

$$DataLabel = \arg \max_{1 \le l \le k} \{\Pr(l|x_i)\} \quad (9)$$

B. Gaussian Mixture Regression

Gaussian Mixture Regression (GMR) is an approach to obtain the estimated output and evaluate the uncertainty based on the probability density function of the GMM in the process of predicting the lookahead distance. The formulation to get the estimated output by maximizing the probability is shown as

$$\hat{h}(t + \Delta t) = \arg \max_{\boldsymbol{\theta}} M(\boldsymbol{O}_t; \hat{\boldsymbol{\theta}}) \quad (10)$$

where $\hat{\boldsymbol{\theta}}$ is the output of Gaussian Mixture Model. This estimated parameter is calculated on the basis of (7).

When GMR is used to predict the lookahead distance, the probability $\hat{\sigma}(t + \Delta t)$ of the output $\hat{h}(t + \Delta t)$ at time t in i^{th} Gaussian model is represented by following:

$$\hat{\sigma}(t + \Delta t) = \frac{\pi_i N_i(\boldsymbol{O}_t, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}{\sum\limits_{i=1}^N \pi_i N_i(\boldsymbol{O}_t, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)} \quad (11)$$

With the $\hat{\sigma}(t + \Delta t)$ and $\hat{h}(t + \Delta t)$ above, average error and standard deviation are selected to statistically analyze the performance of two learning-based models.

IV. DATA COLLECTION AND MODEL TRAINING

A. Data Collection and Sensor Setting

In order to collect the drivers' pure pursuit data, several drivers with different driving personalities and experience participated to manipulate the sensor setting vehicle on the roads of Beijing, China. The driving conditions include intersection, ring roads and highway, etc. The situation is constituted by lane keeping, lane changing, overtaking and other representative situations.

Fig. 3. BIT intelligent vehicle platform

The BIT intelligent vehicle platform was applied to collect the sensor-based vehicles data. The platform BYD Tang with automatic ability is shown in Fig.3. Several sensors equipped are as following:

- OxTs integrated navigation unit. The GPS information including latitude and longitude, heading angle, yaw velocity, horizontal velocity, and timestamp of all data are collected by this sensor.
- BYD's CAN bus network. The vehicles CAN bus network provides vehicles steering wheel angle, throttle pedal position, brake pressure, speed, timestamp of all data, etc.
- Two Mako cameras. The cameras provide the front vision of the vehicle labeled with same timestamp. The camera information is used to recognize different driving situation.

The Velodyne LIDAR HDL-32E is equipped but not actually used for the vehicle's pure pursuit database. All the driving data collected are synchronized with the same timestamp. Data from OxTs and CAN bus is used and analyzed in this paper. Finally, a total length of over 81 hrs vehicle's pure pursuit database has been established.

B. Data Training Process

1) Preprocessing: Moving average filter is used to smooth the raw data collected from vehicle with a window size W = 5. The general data (common situations) is selected manually as training data, in which the extreme situations are not included, because extreme situations are few and diverse from each other resulting in lower modeling performance. Each single driver data is divided into M groups, with $M-1$ groups randomly selected as the training data and the left one group as the test data to evaluate the model, which is called the cross-validation method. All the divided data groups meet

the following regulation:

$$\bigcup_{m=1} \boldsymbol{D}^{k,m} = \boldsymbol{D}^k and \bigcap_{m=1} \boldsymbol{D}^{k,m} = \varnothing, m = 1, 2...10 \quad (12)$$

where $\boldsymbol{D}^{k,m}$ is the mth group's data for the kth single driver data $\boldsymbol{D}^k$. $\boldsymbol{D}$ is the data of the training process, $\boldsymbol{D} = [\{\boldsymbol{O}_1, h_1\}\{\boldsymbol{O}_2, h_2\}, \ldots, \{\boldsymbol{O}_t, h_t\}]$,
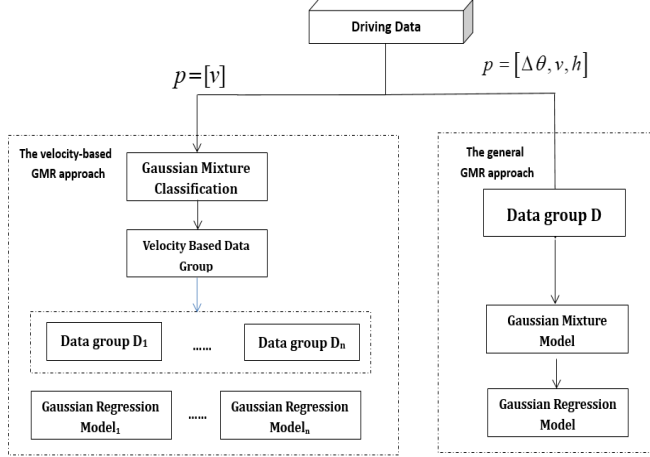

Fig. 4. The model training process of the two different GMR based approaches

2) Driver Model structures: In order to discover the influence of different GMR methods on model results, as shown in Fig.4, two different GMR methods are used to predict the lookahead distance,i.e., the general GMR approach, the velocity based GMR approach.

- The general GMR approach
  The input training data of the general GMR approach are the lookahead distance calculated above and the path point data (course deviation and the velocity) directly obtained from vehicle. Shown as Fig.4, only one GMR model is used to estimate the lookahead distance. The process of obtaining the predicting output data is shown as follow:

$$P(\boldsymbol{D}; \boldsymbol{\theta}_{general}) = \sum_{i=1}^{n_1} p_i g_i(\boldsymbol{D}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \quad (13)$$

$$gdl_i = \arg \max_{1 \le l \le n_1} \{Pr(l|\boldsymbol{D}_i)\} \quad (14)$$

$$\bigcup_{gdl=1} \boldsymbol{D}_{gdl} = \boldsymbol{D} and \bigcap_{gdl=1} \boldsymbol{D}_{gdl} = \varnothing, gdl = 1, 2...n_1 \quad (15)$$

$$\hat{h}(t + \Delta t) = \arg \max_{\boldsymbol{\theta}_{general}} M(\boldsymbol{O}_t; \boldsymbol{\theta}_{general}) \quad (16)$$

$$\hat{\sigma}(t + \Delta t) = \frac{\pi_i N_i(\boldsymbol{O}_t, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}{\sum_{i=1}^{N} \pi_i N_i(\boldsymbol{O}_t, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)} \quad (17)$$

- The velocity based GMR approach
  The input training path point data consist of course deviation and the velocity, and velocity is a major factor of vehicle dynamic. So, shown as Fig.4, a two layer velocity based GMR approach is proposed.

First layer: The input training data are clustered by the Gaussian classification process based on the velocity. The output after divided procedure are shown as following:

$$P(\boldsymbol{v}; \boldsymbol{\theta}_{velocity}) = \sum_{i=1}^{n_2} p_i g_i(\boldsymbol{v}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \quad (18)$$

$$vdl_i = \arg \max_{1 \le l \le n_2} \{Pr(l|v_i)\} \quad (19)$$

$$\boldsymbol{\xi}_i = [\Delta\theta_i, v_i, h_i, sdl_i] = [\boldsymbol{O}_i, h_i, sdl_i] = [\boldsymbol{D}_i, vdl_i] \quad (20)$$

$$\bigcup_{vdl=1} \boldsymbol{D}_{vdl} = \boldsymbol{D} and \bigcap_{vdl=1} \boldsymbol{D}_{vdl} = \varnothing, vdl = 1, 2...n_2 \quad (21)$$

Second layer: Using the clustered data from Gaussian classification process, each Gaussian regression model is trained individually. And the number of the GMR models equals to the number of GMM models based on velocity. The estimated lookahead distance $\hat{h}(t + \Delta t)$ and the probability $\hat{\sigma}(t + \Delta t)$ of output are same as the general GMR approach based on equation (16-17) where $\boldsymbol{\theta} = \boldsymbol{\theta}_{velocity}$.

3) Number of the GMM Components: Our goal is to obtain the most suitable driving model, although the BIC can investigate optimal GMM components, the GMM components and driving model's performance index are different. More components could result in over-fitting problem while less components could decrease the accuracy of prediction because some characteristics of the underlying sources of data cannot be identify. Further the performance index is introduced as follows.

- In general GMR approach, only one Gaussian Regression Model is established and the GMM components number $N \in \{3, 4, 5, 6, 8, 9, 10, 12, 15, 20\}$ are selected.
- In velocity-based GMR approach,The total GMM components number is $\{3, 4, 5, 6, 8, 9, 10, 12, 15, 20\}$, in the meantime, the number of Gaussian Regression Models $n_2 \in \{3, 4, 5, 6\}$ are chosen.

C. Performance Index

The average standard deviation and average error for estimated lookahead distance are chosen to evaluate performance of two learning-based models, which separately stand for uncertainty of the predicting results and degree of accuracy. Two indexes used to evaluate each approach can be described as follows:

$$\bar{e} = \frac{1}{t_{end}} \int_0^{t_{end}} e(t)dt = \frac{1}{t_{end}} \int_0^{t_{end}} \left|\hat{h}(t) - h(t)\right| dt \quad (22)$$

$$\bar{\sigma} = \frac{1}{t_{end}} \int_0^{t_{end}} \sigma(t)dt \quad (23)$$

A better performance of the learning-based model can be regarded as a smaller value of the average errors and the standard deviation.

D. Model Test Process

The initial value of the learning-based model is randomly generated by the k-means cluster, so the model will be trained repeatedly N times to reduce random error, here we choose N=20. The average errors and standard deviation of 20 runs are selected as the performance index to evaluate the model's prediction performance.

The accuracy of lookahead distance is an significant parameter in pure pursuit path-tracking behaviors. In order to testify the performance of two proposed learning-based driver models, the vehicle was tested in simulation by the experimental track including various curved path segments. The test scene and results with comparison are presented in results analysis. To highlight model's sensitivity with the influence of velocity, we set the longitudinal velocity fluctuate within a certain range $v \in (10, 40)[km/h]$. Therefore, in the test, at each state, with the input of $\boldsymbol{O}_t = [\Delta\theta_t, v_t]$, the model can output the optimized lookahead distance $\hat{h}_t$ for pure pursuit path tracking.

V. RESULTS ANALYSIS

In this section, we report the results of two GMR approaches for modeling pure pursuit path tracking and the comparison the performance between general GMR approach and velocity based GMR approach. The difference of the results in the comparison is discussed. In order to present the results analysis clearly, one of the nine drivers' driving data is selected to evaluate the driver model.

A. Influence of the GMM Components and Methods

In Fig.5 shows with different GMM components the average errors and standard deviation of training and test results using different approaches. It is noted that for the same GMM component, the training data's estimation accuracy is higher than the test data while the training data's standard deviation is lower than the test data. More GMM components would decrease the training errors, but might result in overfitting problems and increase computational costs; Conversely, fewer components could reduce computational efforts but may induce larger errors.

1) Numbers of GMM components: Fig.5(a) shows the average error and standard deviation for testing data and training data using two GMR approaches with different GMM components. As for test data, both approach's average errors and standard deviations decrease with the number of the GMM components increasing from 5 to 15, and then the value almost keep constant. The training data's results have same trend but keep constant when GMM component reach 12 for average error in fig.5(c).

2) The influence of different GMR approaches: From the Fig.5, the predicting performance of velocity based GMR approach is better than general GMR approach's in estimation accuracy, which is described by average error and standard deviation. As for test data, with the GMM components increasing from 3 to 15, the velocity based GMR approach's error predicting error decrease from 1 m to 0.66 m and the standard deviation decrease from 5.1 m to 3.7 m.

As for the training data, the model's predicting results have same tendency for two GMR approaches but both average error and standard deviation are slightly lower than test data.
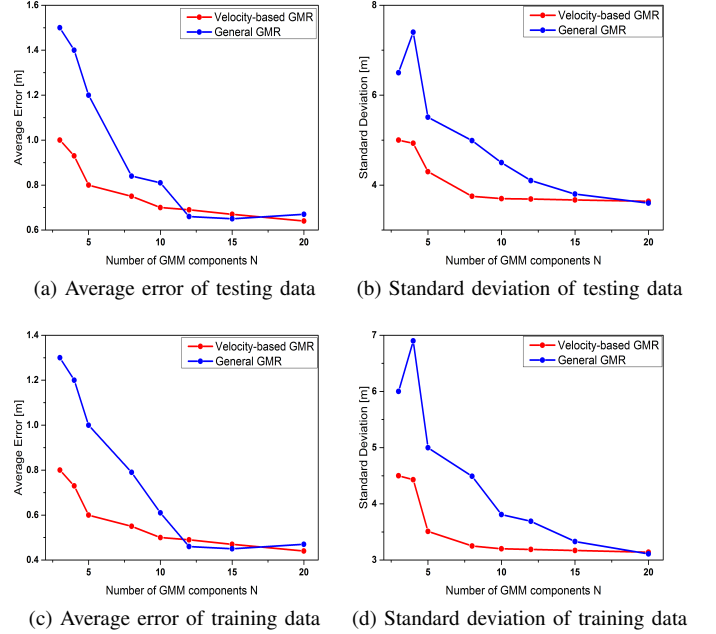

(a) Average error of testing data (b) Standard deviation of testing data

(c) Average error of training data (d) Standard deviation of training data

Fig. 5. The comparison between the two different GMR based approaches with different GMM components

B. Comparison Between Two Methods by Test Track

Test experiments for two methods with same parameters are conducted the following expected path points. For the velocity based GMR approach, the average estimation error can be lower than 0.5 m, but for the general GMR approach, average estimation error is always larger than 0.7 m, even for different numbers of the GMM components and dimensions of training data.

Taking into account of the estimation accuracy of average errors and standard deviations, 12 GMM components for two GMR approaches are selected as the best approaches to make a final comparison. The comparison of the two approaches is shown in the Fig.7. We note that the velocity based GMR approach has a higher ability to increase the model's accuracy and decrease the model's standard deviation. But if with a larger steering angle (more than 15), the uncertainty increases evidently with the reason that the training dataset collected has fewer data for relatively large steering angle (15-80) compared to relatively small steering angle (less than 20).
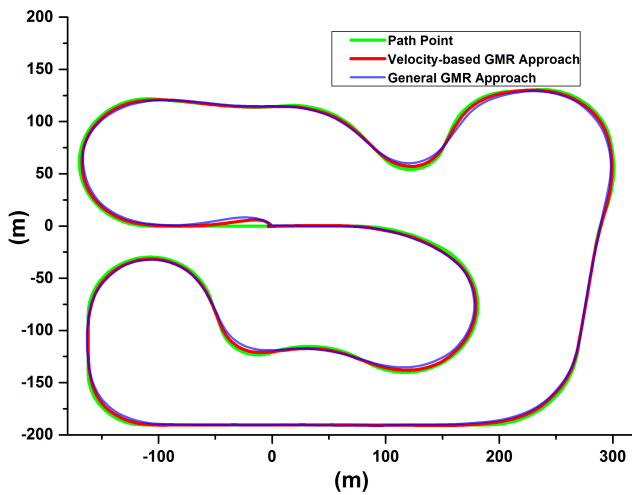
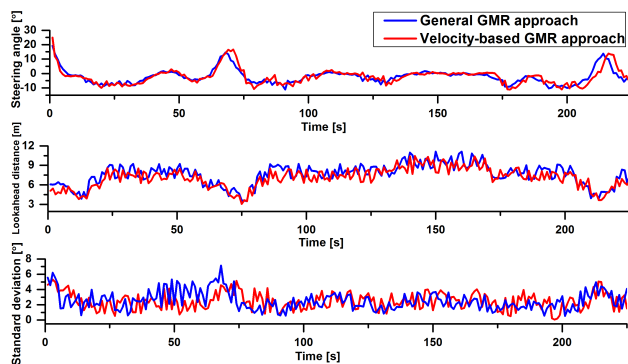Fig. 6. The comparison of pure pursuit between two different GMR based approaches with N=12 GMM components

Fig. 7. The comparison of steering angle, lookahead distance and standard deviation with N=12 GMM components

VI. CONCLUSION AND FUTURE WORKS

In this paper, two learning-based personalized driver models for path-tracking scenario based on pure pursuit algorithm are proposed and compared, the velocity-based GMR approach and the general GMR approach. Each model was trained by the driving data separately to describe the nonlinear and uncertainty of the choose of lookahead distance in the pure pursuit process. The lookahead distance, as the fitting output of the models, was analyzed with different GMM components. The predicting results were tested and compared in a path-tracking scenario.

The results from the training model indicate that the proposed method can generate an accurate lookahead distance in the path-tracking scenario based on pure pursuit algorithm. Because the average error and deviation begin to keep constant from GMM components 12, with the preferred number of GMM components 12-15, the velocity based GMR approach performs better than the general GMR approach in prediction of lookahead distance with 31.1% less average error. The average error and standard deviation of the estimation are 0.66 m and 3.7 m.

In the future work, we will take the influence of the road conditions into consideration and develop a learning-based

bounded driver model.

REFERENCES

[1] Vadim A. Butakov and Petros Ioannou. Personalized driver/vehicle lane change models for adas. *IEEE Transactions on Vehicular Technology*, 64(10):4422–4431, 2015.
[2] Vadim A. Butakov and Petros Ioannou. Personalized driver assistance for signalized intersections using v2i communication. *IEEE Transactions on Intelligent Transportation Systems*, 17(7):1910–1919, 2016.
[3] Vijay Gadepally, Ashok Krishnamurthy, and Umit Ozguner. A framework for estimating driver decisions near intersections. *IEEE Transactions on Intelligent Transportation Systems*, 15(2):637–646, 2014.
[4] Pierre De Beaucorps, Thomas Streubel, Anne Verroust-Blondet, Fawzi Nashashibi, Benazouz Bradai, and Paulo Resende. Decision-making for automated vehicles at intersections adapting human-like behavior. In *Intelligent Vehicles Symposium*, 2017.
[5] Stphanie Lefvre, Ashwin Carvalho, and Francesco Borrelli. A learning-based framework for velocity control in autonomous driving. *IEEE Transactions on Automation Science & Engineering*, 13(1):32–42, 2016.
[6] Wenshuo Wang, Junqiang Xi, and Junmin Wang. Human-centered feed-forward control of a vehicle steering system based on a driver's steering model. In *American Control Conference*, pages 3361–3366, 2015.
[7] Wenshuo Wang, Junqiang Xi, Chang Liu, and Xiaohan Li. Human-centered feed-forward control of a vehicle steering system based on a driver's path-following characteristics. *IEEE Transactions on Intelligent Transportation Systems*, PP(99):1–14, 2017.
[8] Wenshuo Wang, Junqiang Xi, and Huiyan Chen. Modeling and recognizing driver behavior based on driving data: A survey. *Mathematical Problems in Engineering,2014,(2014-2-10)*, 2014(1):1–20, 2014.
[9] Stephanie Lefevre, Ashwin Carvalho, Yiqi Gao, H. Eric Tseng, and Francesco Borrelli. Driver models for personalized driving assistance. *Vehicle System Dynamics*, 53(12), 2015.
[10] Shengbo Eben Li. Economy-oriented vehicle adaptive cruise control with coordinating multiple objectives function. *Vehicle System Dynamics*, 51(1):1–17, 2013.
[11] Chiyomi Miyajima, Yoshihiro Nishiwaki, Koji Ozawa, Toshihiro Wakita, Katsunobu Itou, Kazuya Takeda, and Fumitada Itakura. Driver modeling based on driving behavior and its evaluation in driver identification. *Proceedings of the IEEE*, 95(2):427–437, 2007.
[12] Ding Zhao, Xianan Huang, Huei Peng, Henry Lam, and David J. Leblanc. Accelerated evaluation of automated vehicles in car-following maneuvers. *IEEE Transactions on Intelligent Transportation Systems*, PP(99):1–12, 2017.
[13] S Calinon, F Guenter, and A Billard. On learning, representing, and generalizing a task in a humanoid robot. *IEEE Transactions on Systems Man & Cybernetics Part B*, 37(2):286–298, 2007.
[14] Shogo Sekizawa, Shinkichi Inagaki, Tatsuya Suzuki, Soichiro Hayakawa, Nuio Tsuchida, Taishi Tsuda, and Hiroaki Fujinami. Modeling and recognition of driving behavior based on stochastic switched arx model. *IEEE Transactions on Intelligent Transportation Systems*, 8(4):593–606, 2007.
[15] Ozkan Celik and Seniz Ertugrul. Predictive human operator model to be utilized as a controller using linear, neuro-fuzzy and fuzzy-arx modeling techniques. *Engineering Applications of Artificial Intelligence*, 23(4):595–603, 2010.
[16] Ranjeet Singh Tomar, Shekhar Verma, and G. S Tomar. Prediction of lane change trajectories through neural network. In *International Conference on Computational Intelligence and Communication Networks*, pages 249–253, 2010.
[17] Alireza Khodayari, Ali Ghaffari, Reza Kazemi, and Reinhard Braunstingl. A modified car-following model based on a neural network model of the human driver effects. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 42(6):1440–1449, 2012.
[18] C. J Ostafew, A. P Schoellig, and T. D Barfoot. Learning-based nonlinear model predictive control to improve vision-based mobile robot path-tracking in challenging outdoor environments. *Journal of Field Robotics*, 33(1):133–152, 2016.
[19] R. C. Coulter. Implementation of the pure pursuit path tracking algorithm. *Implementation of the Pure Pursuit Path Tracking Algorithm*, 1992.