

Compensating Delays and Noises in Motion Control of Autonomous Electric Vehicles by Using Deep Learning and Unscented Kalman Predictor

Yanjun Li¹, Guodong Yin¹, *Member, IEEE*, Weichao Zhuang¹, Ning Zhang, Jinxiang Wang, and Keke Geng

Abstract—Accurate knowledge of the vehicle states is the foundation of vehicle motion control. However, in real implementations, sensory signals are always corrupted by delays and noises. Network induced time-varying delays and measurement noises can be a hazard in the active safety of over-actuated electric vehicles (EVs). In this paper, a brain-inspired proprioceptive system based on state-of-the-art deep learning and data fusion technique is proposed to solve this problem in autonomous four-wheel actuated EVs. A deep recurrent neural network (RNN) is trained by the noisy and delayed measurement signals to make accurate predictions of the vehicle motion states. Then unscented Kalman predictor, which is the adaptation of unscented Kalman filter in time-varying-delay situations, combines the predictions of the RNN and corrupted sensory signals to provide better perceptions of the locomotion. Simulations with a high-fidelity, CarSim, full-vehicle model are carried out to show the effectiveness of our RNN framework and the entire proprioceptive system.

Index Terms—Deep learning (DL), four-wheel independently actuated (FWIA) autonomous electric vehicles, network-induced delays, recurrent neural networks (RNNs), unscented Kalman predictor (UKP).

I. INTRODUCTION

AFTER almost 100 years of domination by internal combustion engines, there has been a return of interest in electric vehicles (EVs). Many factors stimulate this resurgence. On the one hand, it is widely believed that EVs can solve energy and environmental problems to some extent. On the other hand, EVs with individually controlled motors,

Manuscript received July 1, 2017; revised December 11, 2017; accepted April 2, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant U1664258 and Grant 51575103, in part by the National Key Research and Development Program in China under Grant 2016YFB0100906 and Grant 2016YFD0700905, in part by the Six Talent Peaks Project in Jiangsu Province under Grant 2014-JXQC001, and in part by the Qing Lan Project and the Fundamental Research Funds for the Central Universities under Grant 2242016K41056. This paper was recommended by Associate Editor X. Xu. (*Corresponding author: Guodong Yin.*)

Y. Li was with the School of Mechanical Engineering, Southeast University, Nanjing 211189, China. He is now with the Department of Mechanical Engineering, City College of New York, New York, NY 10031 USA (e-mail: liyanjunyaya@sina.com).

G. Yin, W. Zhuang, N. Zhang, J. Wang, and K. Geng are with the School of Mechanical Engineering, Southeast University, Nanjing 211189, China (e-mail: ygd@seu.edu.cn; wezhuang@seu.edu.cn; nzhang_cn@seu.edu.cn; wangjx@seu.edu.cn; jsgengke@seu.edu.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSMC.2018.2850367

namely four-wheel independently actuated (FWIA) EVs, allow a significant improvement in motion performance due to their remarkable actuation flexibility [1], [2]. So far, various studies have been conducted on how to fully utilize the potential of the FWIA EVs to enhance the maneuverability and stability of vehicles [3]–[7]. It is also believed that the combination of electrically driven and autonomous driving is the future of vehicle technology [8], [9].

Sophisticated control systems in FWIA EVs or autonomous vehicles are based on the advancement of electrical and electronics technology. In the past, electronic devices in vehicles are connected by point-to-point wiring systems. As the growth of electronic control units (ECUs), continuing to adopt this conventional Electrical/Electronic architecture became impossible because it resulted in an expensive and messy wire system. Nowadays, the data signals between ECUs are transmitted through networks, i.e., controller area network (CAN) or FlexRay, forming a networked control system (NCS). However, as the number of ECUs rising further, which is typical in over-actuated FWIA EVs, the unknown and time-varying communication delays of networks may be large enough to degrade the performances of feedback control systems [10].

Some work has been carried out to deal with the delays induced by in-vehicle networks [11]–[14]. In these studies, two approaches are widely employed to describe the time-varying delays as uncertainties. One is a deterministic method, and the other is a stochastic method. In the deterministic method, delays are assumed as being taken from a fixed probabilistic distribution [11], [12]. Such description is quite simple and keeps the synthesis of the NCS easy. However, it neglects the correlation between current and previous delays, which may increase the conservativeness of the control system. In the stochastic method, the NCS is modeled as a Markov chain [13], [14]. Such random process model takes the dependence between adjacent delays into account and is more precise allowing for the real network phenomena. Once the models of NCS has been established, appropriate robust control methods can be designed to make the performance of the system as high as possible under the uncertainties induced by network delays.

For these studies, an important issue is how to decrease the conservativeness in the algorithms. However, although more accurate NCS models and less conservative control methods are adopted, conservativeness still inevitably exists as long

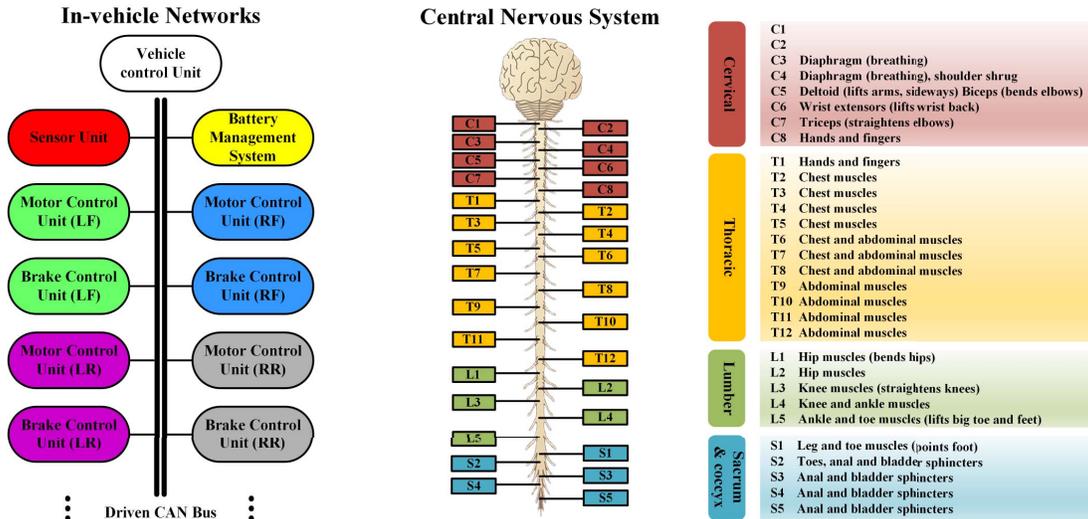


Fig. 1. Comparison diagram of in-vehicle networks and CNS.

as we regard time-varying delays as uncertainties. Moreover, another well-known problem, the noises in sensory signals, is not considered in these literature. These defects motivate us to design a proprioceptive system of autonomous FWIA EVs, which can effectively compensate the time-varying delays in feedback loop (actually, in the autonomous vehicles, motor commands all exist in feedforward loop and generally have high priorities, which means that large network delays concentrate in feedback loop) and suppress the measurement noises simultaneously. The theoretical basis of our method is mainly borrowed from the research on the sensorimotor loop of humans.

A. Inspiration From the Sensorimotor Loop of Humans

Our nervous system is also a typical and intricate NCS. Sensory information from the skin, muscle, and visceral organs are transferred through the peripheral nervous system to the central nervous system (CNS), or, to be more specific, the corresponding spinal cord segment (each segment is responsible for a certain part of the human body). The spinal cord relays signal up to the brain, where the ascending feedback information is integrated and processed. Then, the efferent motor commands are projected forward from cortex to the corresponding spinal cord section [15]. When comparing in-vehicle networks and the nervous system, many similarities can be found. As illustrated in Fig. 1, an actuator/sensor node of in-vehicle networks is like a segment of the spinal cord, and the vehicle control unit (VCU) plays the role of the human brain. Considering the highest speed of neuron signals is only about 120 m/s, the time delay is also an intrinsic nature of sensorimotor control loop of humans. In addition to that, the noise in motor commands and sensory inputs are bigger because of the biomechanical properties of the muscles and neurons. It seems a paradox that we can still control the motion of our limbs and hands so intelligently.

The secret of our outstanding capability to estimate the states of our body and external environment lies in the exploitation of two streams of information. Instead of solely

relying on sensory feedback, which is temporally lagged and tending to be corrupted by considerable amounts of noise, the brain predicts the sensory consequences of our motor commands as well [16], [17]. To predict, the brain requires an internal forward model, which describes the dynamic characteristics of the outside world [18]. The question is: where does the accurate internal model, the base of our prediction, come from? Actually, forward models are not fixed in nature. They should be learned from experience [19]. By comparing the predicted and actual outcomes of a motor command, errors can be generated. Then, human brains take advantage of well-established computational learning rules to translate these errors into changes in synaptic weights to improve future predictions of a forward model.

After the prediction of the internal forward model, measurement signals can be fused to correct the prediction results and form perceptions. The most famous fusion method was proposed by Rudolph Kalman [20], called Kalman filter (KF). Although KF was proposed in engineering, some experiments have examined that it fits the mechanism in our brain well [18], [21], [22].

B. Related Work and Main Contribution of This Paper

In this paper, a proprioceptive system of autonomous FWIA EVs is proposed to double overcome the delays and noises in the feedback loop of in-vehicle networks. First, state-of-the-art deep learning (DL) technology is adopted to represent dynamics models of autonomous FWIA EVs from experience automatically. DL is a branch of machine learning based on a set of algorithms that is especially powerful in extracting useful models [23]. It has turned out to be very good at discovering intricate structures in high-dimensional data and is therefore applicable to many domains of science, business, and government [24]. Since 2006, DL has won numerous official international pattern recognition competitions [25], achieving the first superhuman visual pattern recognition results in limited domains [26], [27]. Second, data fusion technique is exploited to combine the prior information (which is

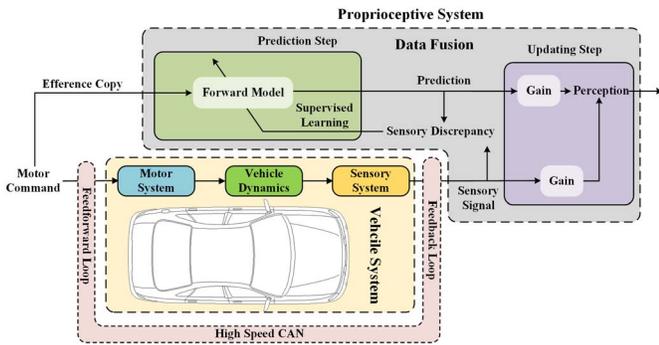


Fig. 2. Overall structure of vehicle proprioceptive system.

learned from experience) and measurement information (which is directly sampled or estimated by the sensory system but corrupted by delays and noises). Thus, this perception system of locomotion is based on the combination of learning and data fusion, just as what is happening in our brains. Using this proprioceptive system, the concerns about conservativeness generated by modeling the time-varying delays in the feedback loop is not required anymore. It should be noted that the attempt to identify the model of intelligent vehicles by machine learning can be dated back to 1990s and still remains popular these days [28]–[31]. However, fusing the learned knowledge with sensory signals to improve the prediction effect has rarely been tried in those works. The overall structure of our vehicle proprioceptive system is shown in Fig. 2.

One quite promising DL solution to tackling the problem of learning sequences of information is the recurrent neural network (RNN) [32]. To train a RNN, it is standard to unroll it through time and convert it into an equivalent very deep feedforward neural network. However, when the error signals backpropagated through time in an unfolded RNN, the gradients typically explode or vanish. As a result, learning with recurrent networks, especially learning long-term dependency, has long been considered difficult [33]. In recent years, thanks to the improved architectures [34], [35], fast implementations and better gradient-following heuristics [36], and deep RNNs are now frequently trained successfully. In this paper, a nine-layer modular deep RNN (MODERNN) is designed to learn the forward model of FWIA EVs.

Nonlinear filter technique is required to utilize the nonlinear feedforward model and sensory signals comprehensively. Extended KF (EKF) is a nonlinear extension of KF, which has been widely adopted in nonlinear state estimators [37]–[39]. However, EKF only provides “first-order” approximations to the probabilities, and it is difficult to calculate the Jacobian in some situations. Another advanced class of nonlinear filters is particle filter (PF). It approximates the filtering process by propagating randomized points through the nonlinear models [40], [41]. PF techniques are especially powerful, but the extraordinary performance is at the cost of additional computation effort, which sacrifices the real-time performance [42]. The unscented KF (UKF) keeps a balance between the low complexity of the KF and the high accuracy of the PF. In addition, both neural networks and UKF

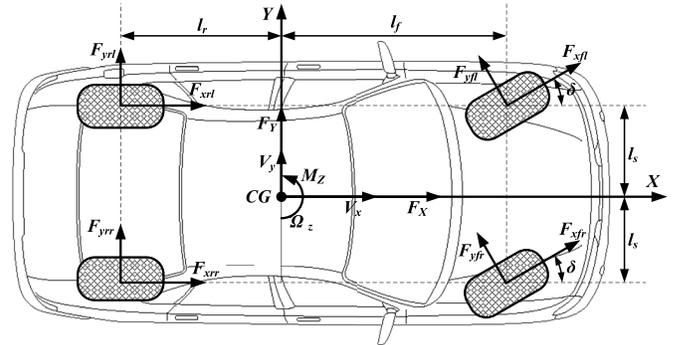


Fig. 3. Schematic of an FWIA EV model.

are well suited for parallel computing and can be massively speeded up by devices like graphics processing unit in actual operation [43]. Therefore, we adapt UKF to the time-varying delay situations at first and then combine the predictions of RNN and sensory measurements through it.

The rest of this paper is organized as follows. A brief dynamic model of FWIA EV is presented in Section II. A nine-layer RNN and its learning algorithm are presented in Section III to extract turning models of autonomous FWIA EVs. Then, in Section IV, an unscented Kalman predictor (UKP) is proposed to cope with time-varying delays in CANs and noises in sensory signals. In Section V, simulation results illustrate that a combination of DL and UKP can perceive vehicle states well with the presence of noise and random delays. Finally, Section VI concludes this paper and discusses the future work.

II. VEHICLE DYNAMICS MODEL

Although we intend to capture the dynamic models of FWIA EVs from sensory data by RNNs, information about these models is still important. Indeed, when the information is available, they should be fused into the design of neural networks as much as possible to alleviate the learning pressure of RNN. Therefore, a model which can describe key characters of the yaw-plane motion of an autonomous FWIA electric ground vehicle is introduced here.

A. Vehicle Planar Dynamic Model

Fig. 3 shows a schematic of the vehicle model. If we ignore the pitch and roll motion, the vehicle body has three degrees of freedom, which can be expressed as [1]

$$\begin{cases} \dot{V}_x = V_y \Omega_z + (F_X - C_a V_x^2)/M \\ \dot{V}_y = -V_x \Omega_z + F_Y/M \\ \dot{\Omega}_z = -V_x \Omega_z + M_Z/I_z. \end{cases} \quad (1)$$

In the above equations, V_x and V_y denote longitudinal and lateral speed, respectively, and Ω_z is the yaw rate. M is the vehicle’s mass, I_z is the vehicle’s yaw inertia, and C_a is the aerodynamic term. F_X , F_Y , and M_Z are the total forces/moment

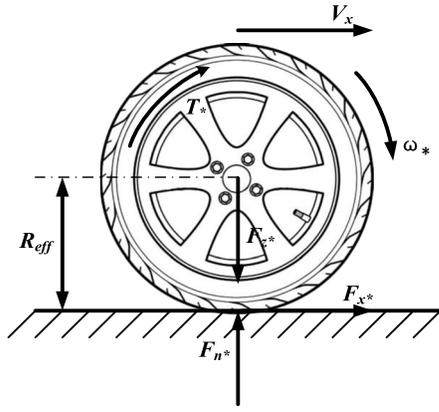


Fig. 4. Schematic of wheel dynamic model.

generated by the four active wheels, which can be described by

$$\begin{cases} F_X = (F_{xfl} + F_{xfr}) \cos \delta - (F_{yfl} + F_{yfr}) \sin \delta + F_{xrl} + F_{xrr} \\ F_Y = (F_{xfl} + F_{xfr}) \sin \delta + (F_{yfl} + F_{yfr}) \cos \delta + F_{yrl} + F_{yrr} \\ M_Z = l_s((F_{xfr} - F_{xfl}) \cos \delta + (F_{yfl} - F_{yfr}) \sin \delta) \\ \quad + l_f((F_{xfl} + F_{xfr}) \sin \delta + (F_{yfl} + F_{yfr}) \cos \delta) \\ \quad - l_r(F_{yrl} + F_{yrr}) + l_s(F_{xrr} - F_{xrl}) \end{cases} \quad (2)$$

where F_{x*} and F_{y*} denote the longitudinal forces of the front and rear wheels, respectively. F_{yf*} and F_{yr*} are the lateral forces of the front and rear wheels, respectively. δ is the steering angle of the front wheel. l_f and l_r are the distances from the center of gravity to the front and rear wheel axle, respectively. l_s is the tread. The suffixes f, r, l , and r mean front, rear, left, and right, respectively.

B. Wheel Dynamic Model

As Fig. 4, if rolling resistance is ignored, the rotational dynamics of each independently driven wheel is represented by

$$\dot{\omega}_* = (T_* - R_{eff}F_{x*})/I \quad (3)$$

where ω_* ($* \in Q := \{fl, fr, rl, rr\}$) indicates the specific tire) is the wheel longitudinal rotational speed and R_{eff} is the tire effective rolling radius, and I is the wheel moment of inertia. T_* is the output torque of a specific in-wheel motor. Typically, the mechanical motions of the independently driven wheels are much slower than the electromagnetic dynamics in motors, which implies that the output torque of a certain in-wheel motor can be simply represented as a product of a factor and the motor command

$$T_* = k_* u_* \quad (4)$$

where u_* is the motor command to a certain motor.

The slip angle of each tire is a key state in tire dynamics and can be defined as the angular difference between the orientation of a wheel and the velocity of the wheel center

$$\begin{cases} \alpha_{fl} = -\delta + \arctan\left(\frac{V_y + \Omega_z l_f}{V_x - \Omega_z l_s}\right) \\ \alpha_{fr} = -\delta + \arctan\left(\frac{V_y + \Omega_z l_f}{V_x + \Omega_z l_s}\right) \\ \alpha_{rl} = \arctan\left(\frac{V_y - \Omega_z l_f}{V_x - \Omega_z l_s}\right) \\ \alpha_{rr} = \arctan\left(\frac{V_y - \Omega_z l_f}{V_x + \Omega_z l_s}\right). \end{cases} \quad (5)$$

The speeds at the wheel centers are

$$\begin{cases} V_{xfl} = (V_x - \Omega_z l_s) \cos \delta + (V_y + \Omega_z l_f) \sin \delta \\ V_{xfr} = (V_x + \Omega_z l_s) \cos \delta + (V_y + \Omega_z l_f) \sin \delta \\ V_{xrl} = V_x - \Omega_z l_s \\ V_{xrr} = V_x + \Omega_z l_s. \end{cases} \quad (6)$$

When driven or braked, tire center speed does not equal to its circumferential. This important phenomenon is called longitudinal tire slip, which is defined as

$$s_* = (\omega_* R_{eff} - V_{x*}) / \max(V_{x*}, \omega_* R_{eff}). \quad (7)$$

C. Tire Dynamic Model

Studies in tire-road friction modeling are abundant, and the magic formula tire model (MFTM) is one of the most well-known of them [44]. The basic equations of the MFTM for the case of pure slip are

$$\begin{cases} y(x) = D \sin\{C \arctan[Bx - E(Bx - \arctan Bx)]\} \\ Y(X) = y(x) + S_v \\ x = X + S_h \end{cases} \quad (8)$$

where $Y(X)$ represents tire pure longitudinal force F_{x0*} or lateral force F_{y0*} of a specific tire, respectively. X is tire slip ratio or slip angle. Coefficient B, C, D, E, S_h , and S_v are the coefficients related to tire normal force.

D. Sensory Signals

In reality, only a part of the states in vehicles can be measured from the sensors, and the measurement signals are usually noisy and time delayed. In this paper, we only consider seven typical sensory (or estimated) signals: four wheel speeds (from wheel speed sensors), lateral acceleration and yaw rate (from gyroscope), and longitudinal speed (from inertial sensors and GPS or estimated). These signals are crucial for vehicle dynamics control. Noises of sensor signals can be easily inferred by observation or looked up in manuals. In general, the noises in wheel speed signals are relatively large, around 4 rpm (1σ). The noise in longitudinal speed signals is moderate, around 0.1 km/h. Those in yaw rate and lateral acceleration are relatively small, around 0.2 deg/s and 0.005 g, respectively [45].

In this paper, the in-vehicle network refers to CAN due to its wide applications in industry. We assume that the sensors and VCU act in a time-driven fashion and the actuators act in an event-driven fashion. A conservative and simple model of the network induced time-varying delays is that they are bounded [46] and uniformly distributed in the interval [11]

$$\max\{0, \tau_{k-1} - T_s\} \leq \tau_k \leq \tau_{large} \quad (9)$$

where $\tau_{large} = 1.7 T_s$, T_s is the sampling period, and k is the discrete-time index. This model is used in the simulation process to verify the effectiveness of our proprioceptive system. Note that, in autonomous vehicles, the motor commands transmitted in a CAN are generally settled with high priorities. Thus, the delays in feedforward loop (from VCU to actuator nodes) are relatively small and deterministic. On the contrary, the priorities of measurement signals are generally low, and the feedback delays are relatively large and random. For these

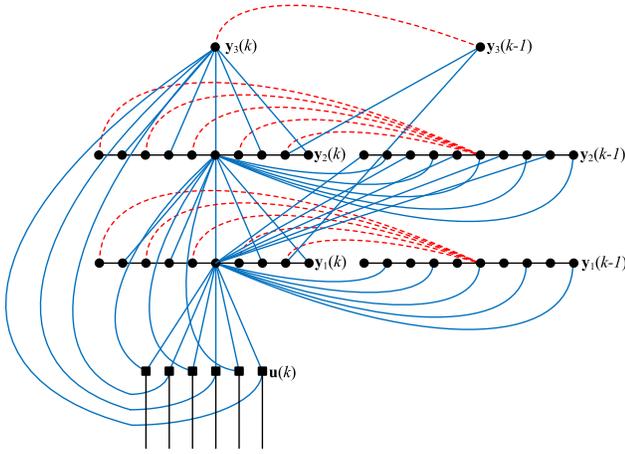


Fig. 5. Three-layer MODERNN, only a part of synaptic connections are illustrated here for brevity. Red lines refer to the connections from hidden or output units to the context units.

reasons, only the feedback delays are considered in this paper for the present.

III. LEARNING THE FORWARD MODEL BY DEEP RNN

Although the model given in the previous section is high order and nonlinear, it is still a simplified version. The real autonomous FWIA EV system couples rigid body dynamics with suspension dynamics, motor dynamics, aerodynamic forces, and internal control loops. Much of this dynamic system involves hidden state variables which cannot be easily measured. Owing to the modern structure and training method of RNN, it is possible to learn an accurate dynamic model of autonomous FWIA EV directly from corrupted sensory data. In this paper, MODERNN architecture and corresponding learning algorithms are employed. We favor MODERNN due to its proven ability to decrease the vanishing/exploding gradient problem in space, and flexible training methods can be applied to this architecture [47]. The detailed network architecture, overall framework and training method are introduced below.

A. RNN Architecture

MODERNN is a generalization over traditional recurrent multilayer perceptron (RMLP). It not only has locally recurrent layers as RMLP but also has connections linking all layers. Fig. 5 illustrates a model of a three-layer MODERNN. The dynamical equations of the MODERNN are described as follows [47]:

$$\begin{cases} \mathbf{x}_i(k) = \mathbf{A}_i \mathbf{y}_i(k-1) + \mathbf{B}_i \mathbf{u}_i(k) + \mathbf{b}_i \\ \mathbf{y}_i(k) = \mathbf{f}_i(\mathbf{x}_i(k)) \end{cases} \quad (10)$$

where i is the layer index, $\mathbf{x}_i(k) \in \mathbb{R}^{n_i}$ is the state of the layer, $\mathbf{y}_i(k) \in \mathbb{R}^{n_i}$ is the output of the layer, $\mathbf{u}_i(k) \in \mathbb{R}^{m_i}$ is the input to the layer, $\mathbf{A}_i(k) \in \mathbb{R}^{n_i \times n_i}$ is the feedback weight matrix, $\mathbf{B}_i \in \mathbb{R}^{n_i \times m_i}$ is the input weight matrix, $\mathbf{b}_i \in \mathbb{R}^{n_i}$ is a bias weight vector, $\mathbf{f}_i(\cdot)$ is the layer activation function, n_i is the number of neurons in the layer, and m_i is the number of input signals to the layer.

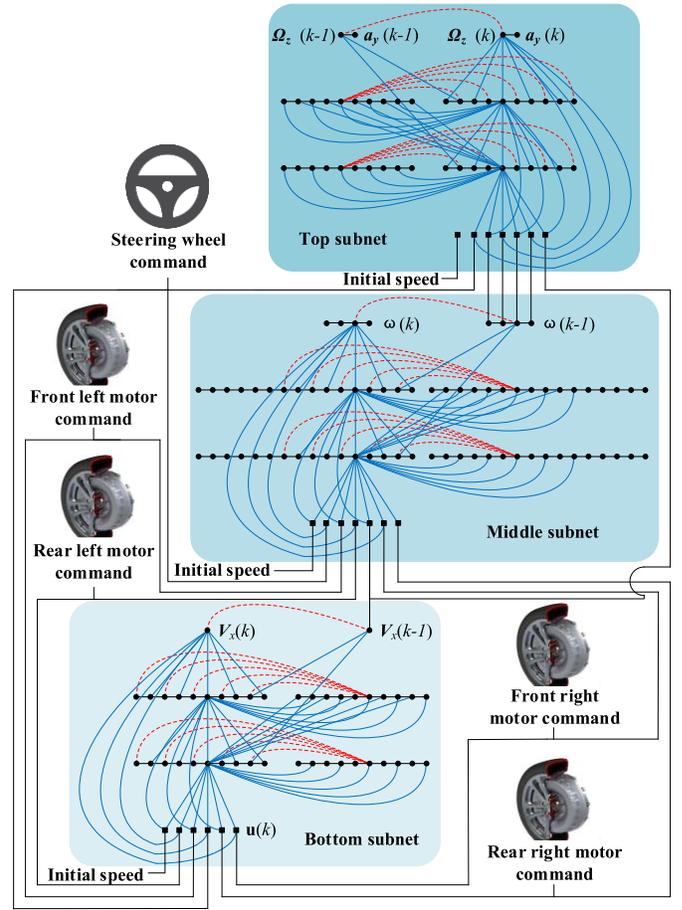


Fig. 6. Overall framework of RNN to predict the vehicle states.

The input to a MODERNN is a time series

$$\mathbf{u}_i(k) = [u^T(k) \ y_1^T(k) \ \cdots \ y_{i-1}^T(k) \ y_{i+1}^T(k-1) \ \cdots \ y_L^T(k-1)]^T \quad (11)$$

where $u(k) \in \mathbb{R}^m$ is the input to the network, L is the recurrent layers of the network, and m is the number of independent inputs to the network.

B. Overall Framework of Network

Our framework of the network to learn the internal forward model of an autonomous FWIA EV is shown in Fig. 6. The network consists of nine layers, six inputs, and seven outputs. It includes three subnetworks, each of which consists of three layers. We conducted several tests to determine the specific structure of each subnet, and achieve a balance between complexity of the neural network and the accuracy of prediction.

In the bottom subnetwork, there are ten neurons in each hidden layer. This subnetwork is designed to learn the dynamics of vehicle longitudinal speed change. It has six inputs, including time sequences of a steering wheel command, four motor commands, and initial speed (initial speed is kept constant through the sequence). The output of this subnet is the

prediction of the longitudinal speed change. The change of longitudinal speed is learned first for three reasons.

- 1) It is the base of the whole vehicle dynamics.
- 2) Its regularity is relatively simple to learn.
- 3) The noise in the estimated longitudinal speed signal is moderate, as described in Section II.

The middle subnetwork has 16 neurons in each hidden layer and four neurons in output layer. We utilize it to learn the laws of wheel speed change of four wheels. According to (7), the longitudinal speed of the vehicle is essential to calculate slip ratio. Thus, it is also important to predict the wheel speed change. For this reason, the output of the bottom subnetwork is defined as an input of the middle subnetwork. Except that, the inputs of middle subnetwork are the same as those of the bottom one.

Tires are the only vehicle components which can generate external forces to affect vehicle motions, and essential to vehicle dynamics and control. As described in (8), slip ratios and slip angles are two main arguments of tire models. Once the longitudinal speed of the vehicle and wheel speeds of four independently driven wheels have been learned, slip ratios can be obtained. Then, we can easily predict the tire forces and lateral motions of the vehicle body. The top subnetwork has seven inputs, including initial vehicle speed, steering wheel command, and the five outputs of the previous two subnetworks. Each hidden layer consists of ten neurons. The target of this subnetwork is learning to predict the lateral motion states. In this paper, lateral acceleration and yaw rate are the final two outputs of the whole network.

The training of this nine-layer MODERNNN proceeds bottom up, similar to that of deep belief networks [48]. The bottom subnet is trained first. Then, the outputs of the bottom subnet are treated as input to train the middle subnet, and so forth. We adopt an offline, batch learning method as in [47] to train each subnet. Considering the noise in sensory signals, cross-validation is suitable for this paper. According to the protocol of CAN, a message transmitted at a certain time can never arrive before a message with the same ID that was transmitted at a previous time. This character of CAN is really useful because, despite time-varying delays, it is easy to pair the motor commands with corresponding sensory outputs in a one-to-one manner. Thus, the training of the RNN can be successfully conducted even in delayed and noisy situations. More training details can be seen in Section V.

IV. FUSING THE DATA BY UNSCENTED KALMAN PREDICTORS

Kant and Smith [49], one of the greatest philosopher, regarded that our perceptions are not a physiological process in which the sense organs simply transmit information to the brain, but a psychological process in which our brain combines the sensory information with what it already predicts. Today, this combination process is often referred to as Bayes filter, which obtains a posterior estimation by prior beliefs and measurements. Following this philosophy, in this paper, the predictions of RNN and the measurement signals are combined to make autonomous FWIA EVs have their proprioception.

A. Unscented Kalman Filter

UKF takes both advantage of the PF and EKF. It is similar to PF, but the difference is that in UKF, particles are deterministically chosen. For this reason, UKF has better real-time performance. The state is again assumed a Gauss distribution as in EKF but is now determined by the carefully selected sigma points. When propagated, they capture the statistical characteristics up to the third order for any nonlinearity, which is better than EKF [50]. Also, there is no need for the Jacobian and Hessian matrices, which can reduce the difficulty in derivation and the possibility of mistakes.

UKF takes two steps, namely the prediction step and the updating step, to fuse the information extracted by DL and sensory information measured from vehicle sensors. The prediction step uses prior information (the model of the system) and previous estimation results to give a prediction of current states. As a precise internal forward model of an autonomous FWIA EV has been learned by MODERNNN in Section IV, this step can make full use of the learned knowledge by propagating particles through the trained deep RNN. In the updating step, the prior prediction is combined with observation information from sensors to increase the accuracy of the estimation further. This improved result is termed as *a posteriori* estimation. Fusion gains of prediction and measurement are dependent on respective uncertainties.

Algorithms 1 and 2 show the pseudocode of the above two steps, and then the whole procedure of UKF can be easily explained by Algorithm 3 [51]. In the algorithms, n is the number of neurons in a subnet, and \mathbf{f} is the corresponding dynamic function. λ is a scaling parameter defined as $\lambda = c - n$, where $c = \alpha^2(n + \kappa)$. The constants $\alpha = 1$, $\beta = 2$, and $\kappa = 0$ are used in this paper. These parameters together generate the parameter matrixes \mathbf{W} and \mathbf{w}_m . \mathbf{m} and \mathbf{m}^- are *a posteriori* estimates and *a priori* predictions of the states, respectively. \mathbf{P} and \mathbf{P}^- are *a posteriori* estimate covariance matrix and *a priori* prediction covariance matrix, respectively. \mathbf{Q} and \mathbf{R} represent process noise covariance matrix and measurement noise covariance matrix, respectively. \mathbf{I} is a $2n \times 2n$ identity matrix. \mathbf{X} is the matrix of sigma points.

B. Unscented Kalman Predictor

UKF is useful in nonlinear state estimation. However, considering the varying delays in communication, in some time steps, no measurement signals arrive at the VCU, and the updating process cannot be carried out. To adapt to this time delay situations, we proposed UKP, which is a simple variation of UKF. When VCU receives no measurement signals in a sampling period, the UKP make predictions based on the prediction results at the previous step. Once the measurement signals come, the updating step is conducted to correct the prediction bias. The detailed description of UKP is given in Algorithm 4. Matrixes \mathbf{mp} and \mathbf{Pp} , respectively, keep a record of the newest prediction of the states and the newest covariance matrix at the current time step k , regardless of the arrival of the measurements. Subscript new means the time index of the newest received measurement signals. $\text{new} = z - 1$ if no

Algorithm 1 PredictFun

Input: \mathbf{m}_{k-1} , k , \mathbf{P}_{k-1} and \mathbf{Q}_{k-1}
Output: \mathbf{m}_k^- , \mathbf{W} , \mathbf{w}_m , and \mathbf{P}_k^-

- 1: $\mathbf{w}_m \leftarrow [W_m^{(0)} \dots W_m^{(2n)}]^T$
- 2: $\mathbf{W} \leftarrow (\mathbf{I} - [\underbrace{\mathbf{w}_m \dots \mathbf{w}_m}_{2n+1}]) \times \text{diag}(W_c^{(0)} \dots W_c^{(2n)})$
 $\times (\mathbf{I} - [\underbrace{\mathbf{w}_m \dots \mathbf{w}_m}_{2n+1}])^T$
- 3: $W_m^{(0)} \leftarrow \lambda/(n+\lambda)$
- 4: $W_c^{(0)} \leftarrow \lambda/(n+\lambda) + (1 - \alpha^2 + \beta)$
- 5: $W_m^{(i)} \leftarrow 1/\{2(n+\lambda)\}$, $i = 1, \dots, 2n$
- 6: $W_c^{(i)} \leftarrow 1/\{2(n+\lambda)\}$, $i = 1, \dots, 2n$
- 7: $\hat{\mathbf{X}}_{k-1} \leftarrow [\mathbf{m}_{k-1} \dots \mathbf{m}_{k-1}] + \sqrt{c}[\mathbf{0} \quad \sqrt{\mathbf{P}_{k-1}} \quad -\sqrt{\mathbf{P}_{k-1}}]$
- 8: $\hat{\mathbf{X}}_k \leftarrow \mathbf{f}(\hat{\mathbf{X}}_{k-1}, k-1)$
- 9: $\mathbf{m}_k^- \leftarrow \hat{\mathbf{X}}_k \mathbf{w}_m$
- 10: $\mathbf{P}_k^- \leftarrow \hat{\mathbf{X}}_k \mathbf{W} [\hat{\mathbf{X}}_k]^T + \mathbf{Q}_{k-1}$
- 11: **return** \mathbf{m}_k^- , \mathbf{W} , \mathbf{w}_m , and \mathbf{P}_k^-

Algorithm 2 UpdateFun

Input: \mathbf{m}_k^- , k , \mathbf{P}_k^- , \mathbf{W} , \mathbf{w}_m , \mathbf{R}_k , and y_k
Output: \mathbf{m}_k and \mathbf{P}_k

- 1: $\hat{\mathbf{X}}_k^- \leftarrow [\mathbf{m}_k^- \dots \mathbf{m}_k^-] + \sqrt{c}[\mathbf{0} \quad \sqrt{\mathbf{P}_k^-} \quad -\sqrt{\mathbf{P}_k^-}]$
- 2: $\mathbf{Y}_k^- \leftarrow \mathbf{h}(\hat{\mathbf{X}}_k^-, k)$
- 3: $\boldsymbol{\mu}_k \leftarrow \mathbf{Y}_k^- \mathbf{w}_m$
- 4: $\mathbf{S}_k \leftarrow \mathbf{Y}_k^- \mathbf{W} [\mathbf{Y}_k^-]^T + \mathbf{R}_k$
- 5: $\mathbf{C}_k \leftarrow \hat{\mathbf{X}}_k^- \mathbf{W} [\mathbf{Y}_k^-]^T$
- 6: $\mathbf{K}_k \leftarrow \mathbf{C}_k \mathbf{S}_k^{-1}$
- 7: $\mathbf{m}_k \leftarrow \mathbf{m}_k^- + \mathbf{K}_k [y_k - \boldsymbol{\mu}_k]$
- 8: $\mathbf{P}_k \leftarrow \mathbf{P}_k^- - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T$
- 9: **return** \mathbf{m}_k and \mathbf{P}_k

Algorithm 3 UKF

- 1: $\mathbf{m}_0 \leftarrow E[\mathbf{x}_0]$
- 2: $\mathbf{P}_0 \leftarrow E[(\mathbf{x}_0 - \mathbf{m}_0)(\mathbf{x}_0 - \mathbf{m}_0)^T]$
- 3: **while** $k < k_{\max}$
- 4: Read \mathbf{y}_k
- 5: $[\mathbf{m}_k^-, \mathbf{W}, \mathbf{w}_m, \mathbf{P}_k^-] \leftarrow \text{PredictFun}(\mathbf{m}_{k-1}, k, \mathbf{P}_{k-1}, \mathbf{Q}_{k-1})$
- 6: $[\mathbf{m}_k, \mathbf{P}_k] \leftarrow \text{UpdateFun}(\mathbf{m}_k^-, k, \mathbf{P}_k^-, \mathbf{W}, \mathbf{w}_m, \mathbf{R}_k, \mathbf{y}_k)$
- 7: $k \leftarrow k + 1$
- 8: **end while**

measurement signals arrive during time step $k-1$ and time step k . In line 19, sign \sim indicates that the corresponding variable is not important and can be ignored.

In this paper, an UKP is designed for each subnet described in the previous section. So there are a total of three predictors. It is also assumed that the process noise \mathbf{Q}_k and measurement noise \mathbf{R}_k in UKPs are constant, namely, $\mathbf{Q}_k = \mathbf{Q}$, $\mathbf{R}_k = \mathbf{R}$. Measurement noises are easy to know, as mentioned in Section II, but process noises are not considered in this paper. Nevertheless, process noise covariance matrixes are still important in this paper. Obviously, it is impossible that the forward model learned by RNN is completely accurate. Keeping this in mind, the process noise covariance matrixes can be treated as a measure of the confidence level, or the uncertainty of the RNN prediction. For the simplicity, it is

Algorithm 4 UKP

- 1: $\mathbf{m}_0 \leftarrow E[\mathbf{x}_0]$
- 2: $\mathbf{m}_p \leftarrow \mathbf{m}_0$
- 3: $\mathbf{P}_0 \leftarrow E[(\mathbf{x}_0 - \mathbf{m}_0)(\mathbf{x}_0 - \mathbf{m}_0)^T]$
- 4: $\mathbf{P}_p \leftarrow \mathbf{P}_0$
- 5: $[\mathbf{m}_1^-, \mathbf{W}, \mathbf{w}_m, \mathbf{P}_1^-] = \text{PredictFun}(\mathbf{m}_0, 1, \mathbf{P}_0, \mathbf{Q})$
- 6: $k \leftarrow 1$, $z \leftarrow 0$, $s \leftarrow 0$
- 7: **while** $k < k_{\max}$
- 8: read new measurement signals $\mathbf{y}_z \sim \mathbf{y}_{new}$ in the buff
- 9: **while** $z < new + 1$
- 10: $[\mathbf{m}_z^-, \mathbf{P}_z^-] \leftarrow \text{UpdateFun}(\mathbf{m}_z^-, z, \mathbf{P}_z^-, \mathbf{W}, \mathbf{w}_m, \mathbf{R}_z, \mathbf{y}_z)$
- 11: $z \leftarrow z + 1$
- 12: $s \leftarrow z$
- 13: $[\mathbf{m}_z^-, \mathbf{W}, \mathbf{w}_m, \mathbf{P}_z^-] \leftarrow \text{PredictFun}(\mathbf{m}_{z-1}^-, z, \mathbf{P}_{z-1}^-, \mathbf{Q}_{z-1})$
- 14: $\mathbf{m}_{p_s} \leftarrow \mathbf{m}_s^-$
- 15: $\mathbf{P}_{p_s} \leftarrow \mathbf{P}_s^-$
- 16: **end while**
- 17: **while** $s < k$
- 18: $s \leftarrow s + 1$
- 19: $[\mathbf{m}_{p_s}, \sim, \sim, \mathbf{P}_{p_s}] \leftarrow \text{PredictFun}(\mathbf{m}_{p_{s-1}}, s, \mathbf{P}_{p_{s-1}}, \mathbf{Q}_{s-1})$
- 20: **end while**
- 21: $k \leftarrow k + 1$
- 22: **end while**

TABLE I
SOME IMPORTANT PARAMETERS IN CARSIM SIMULATIONS

Quantity	Value	Unit
Sprung mass	1022	kg
Unsprung mass	160	kg
Yaw Inertia	1471.7	kg·m ²
Wheel moment of inertia	0.8	kg·m ²
Height of the center of gravity of sprung mass	600	mm
Distance between center of gravity and front axle	1197	mm
Distance between center of gravity and rear axle	1203	mm
Wheel tread	1364	mm
Effective rolling radius	270	mm
Tire-road friction coefficient	0.85	-

assumed that \mathbf{Q} is the product of an identity matrix and a factor as

$$\mathbf{Q} = q\mathbf{I}, \quad q > 0. \quad (12)$$

Then, q is regarded as the optimizing parameter and can be adjusted carefully to optimize the fusion results.

V. SIMULATION RESULTS

A. Training Details

Training and test data of the MODERNNN were collected from simulation results of Carsim (a specialized software to analyze vehicle dynamics, calculate a car's performance characteristics, and develop active controllers). The training dataset consists of 100 input-output pairs (collected from 100 times Carsim simulations), while the test data set consists 5000 ones (collected from 5000 times Carsim simulations). In each simulation, the steering wheel command was randomly picked from fishhook style or sine wave style, turning left or turning

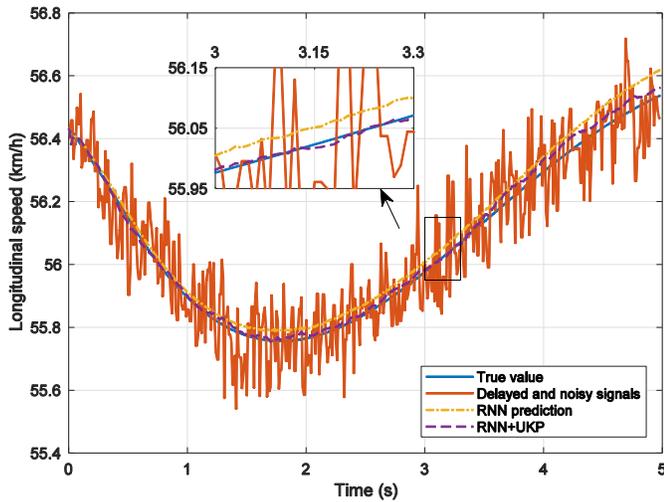


Fig. 7. Performance of longitudinal speed perception.

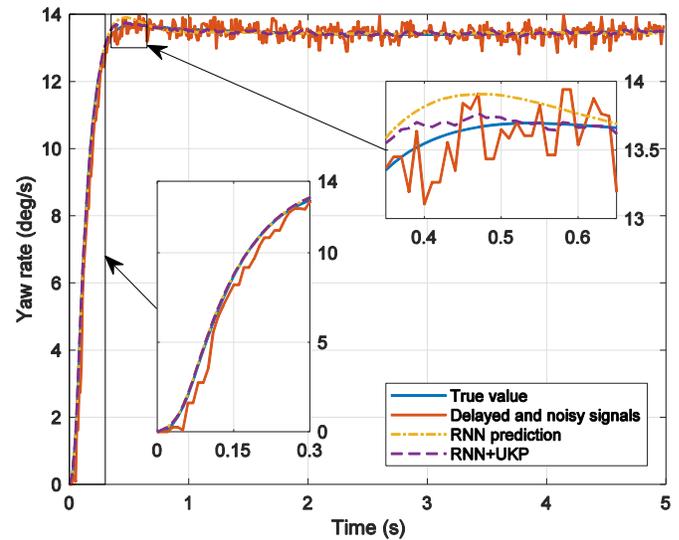


Fig. 9. Performance of yaw rate perception.

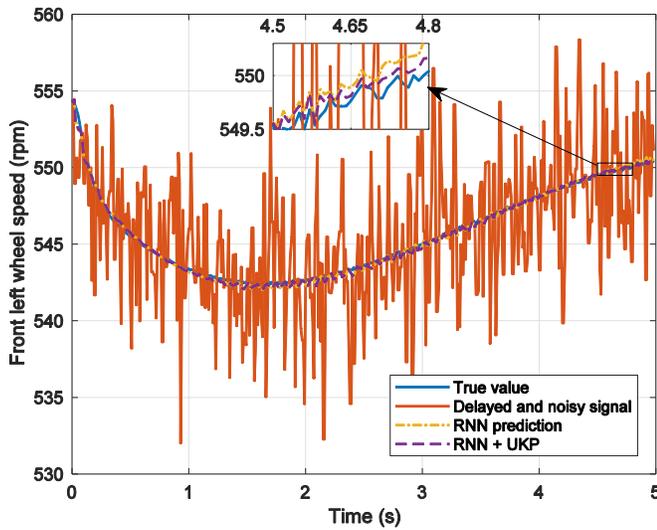


Fig. 8. Performance of front left wheel speed perception.

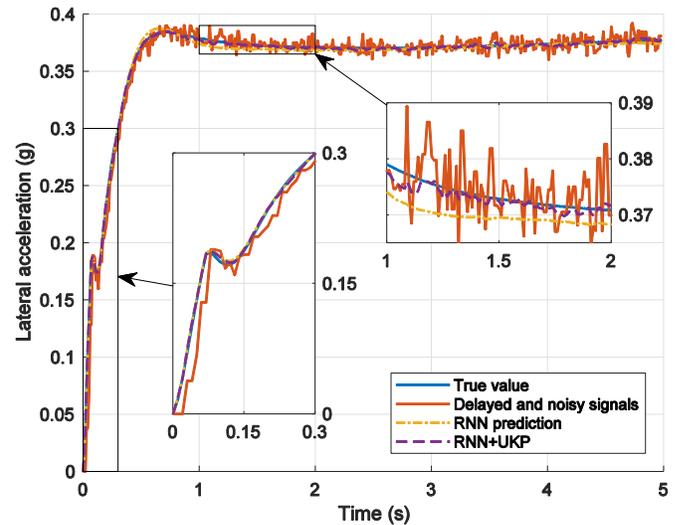


Fig. 10. Performance of lateral acceleration perception.

right. In fishhook style, the steer ramp was picked randomly in 600–700 deg/s. In sine wave style, the frequency was picked randomly from 0.1875–0.25 Hz. In both styles, the maximum steering wheel angle was picked randomly from 10°–60°. Motor commands were generated by a controller which designed to keep the longitudinal speed constant and add a direct yaw moment to the vehicle body (proportional to steering angle, the proportional coefficient can be random numbers). The initial longitudinal speed of the vehicle was set randomly in middle-speed range (40–70 km/h).

Some important parameters in Carsim simulations are listed in Table I. The inputs (steering wheel command and four electric motor commands) and outputs (vehicle longitudinal speed, four-wheel speeds, lateral acceleration, and yaw rate) for each pair are trajectories over time with the same length. The length of each generated trajectory is 5 s in time with a 100-Hz sampling frequency. Corresponding noises and delays were added to each element of the outputs to simulate the real conditions. Equation (9) is adopted as the delay model in the simulation,

and the noise level of longitudinal speed, wheel speed, yaw rate, and lateral acceleration are 4 rpm (1σ), 0.1 km/h (1σ), 0.2 deg/s (1σ), and 0.005 g (1σ), respectively, as described in Section II. We aim to predict the motions of vehicles at each time step, so the motor commands were paired with the output of next time step before training. On a portable computer with a Core I7 processor and an 8-GB RAM, it takes about 20 min to complete the training of each subnetwork.

B. Performance of the Proprioceptive System

We compare the state prediction performance of the trained MODERNN and the overall proprioceptive system (MODERNN + UKP) for a sample in the test dataset. The results are given in Figs. 7–10. In this sample, the autonomous FWIA EV does a fishhook style maneuver at the speed of 56 km/h (for brevity, only one of the four wheels' speed is given). As can be seen from the figures, prediction results of RNN are all close to the true values, which indicates that the

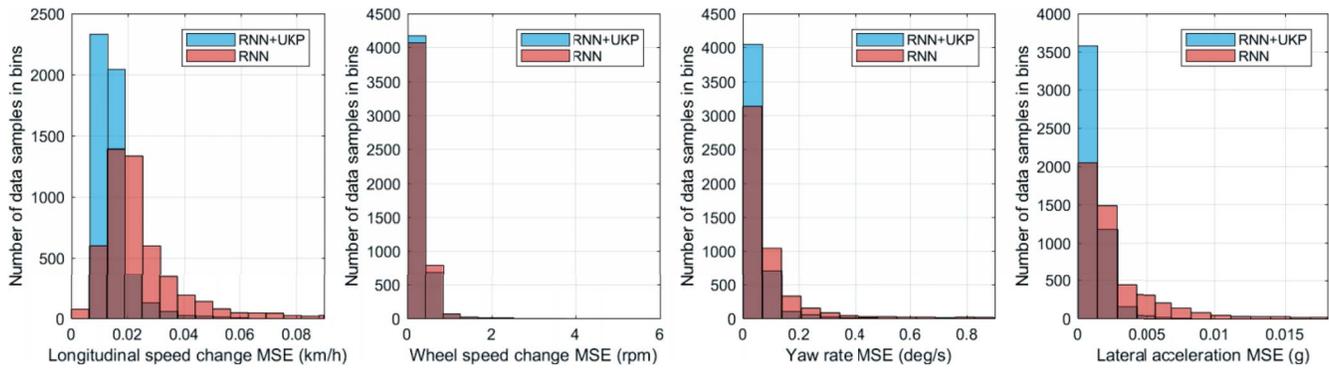


Fig. 11. MSE histograms of RNN prediction and RNN + UKF results.

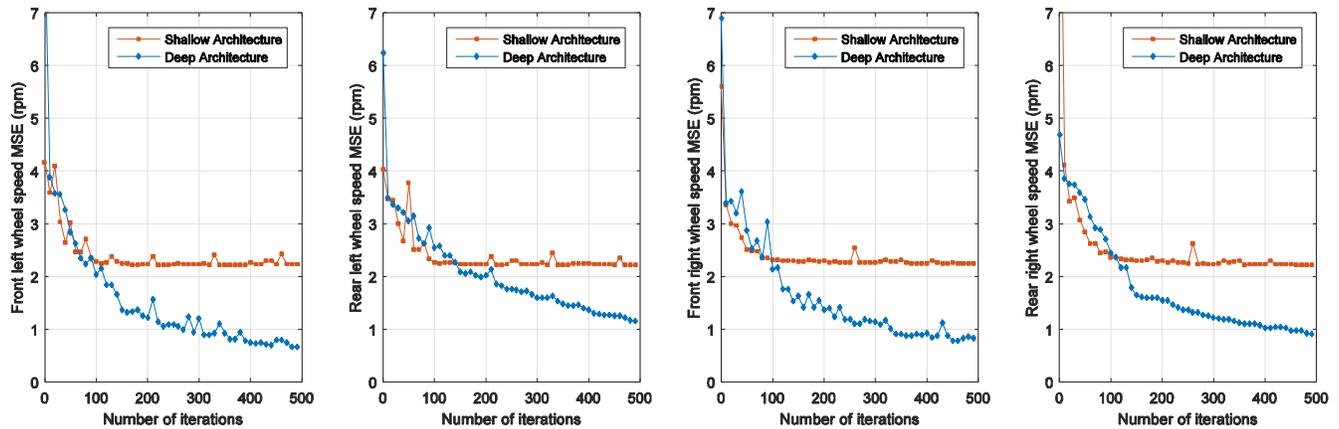


Fig. 12. Learning curves of deep and shallow RNN.

MODERNN we designed has successfully learned the vehicle dynamics from experience. The existence of time delay factors is particularly obvious when there are significant changes in vehicle states, as shown in zoomed-in areas of Figs. 9 and 10. Although the measurement signals are lagged randomly, the time-varying delays of in-vehicle networks have no negative effect on the predictions of RNN. This is the key to solving the problem of feedback delays.

The results of the proprioceptive system are based on the combination of RNN prediction and measurement signals. As shown in Figs. 7–10, the fusion of two streams of information can work better than each one working individually. When noises and delays corrupt the sensory signals, RNN predictions provide relatively precise information. When the predictions of RNN are biased, measurement signals can correct them to a certain degree.

The perception performance over the whole test dataset (5000 input–output pairs) is illustrated in Fig. 11. Mean squared errors (MSEs) of RNN and RNN+UKF are calculated to evaluate their performance for each data sample. Then the histogram shows the distribution of MSE values over the entire dataset. The blue bar and orange bar represent the prediction error distribution of RNN+UKF and single RNN, respectively. The two colors fuse together and form a third color where two bars overlap with each other. It can be observed that in all the histograms, the majority of MSE values fall close to zero, which indicates a strong prediction performance.

Moreover, the distributions of RNN+UKF results have higher peaks than that of pure RNN prediction, which indicates that the combination of two streams of information performs better than one.

To demonstrate the effectiveness of our overall RNN framework, an experiment is conducted to compare the learning results of our deep RNN and a shallow one. In the experiment, a deep RNN (the middle subnet described in Section III supported by the trained bottom subnet) and a shallow RNN (also a three-layer MODERNN with 16 neurons in each hidden layer, but without the support of bottom subnet) are trained to learn the speed changes of four active wheels. Mini-batch gradient descent (batch size is ten) with cross-validation was adopted as the learning method. The learning curves are shown in Fig. 12, with the number of iterations as the horizontal axis, and the MSE of prediction of all training data as the vertical axis. It is clear that after an epoch of training, the deep framework has better learning result. Moreover, the shallow RNN has reached a local minimum, while the learning curves of the deep RNN still show downward trends, which demonstrates the potential for better performance.

C. Proprioceptive System-Based Yaw Rate Control

The sliding mode control (SMC), especially the second order SMC (SOSMC) has long been considered an ideal robust algorithm for vehicle control systems. However, it has

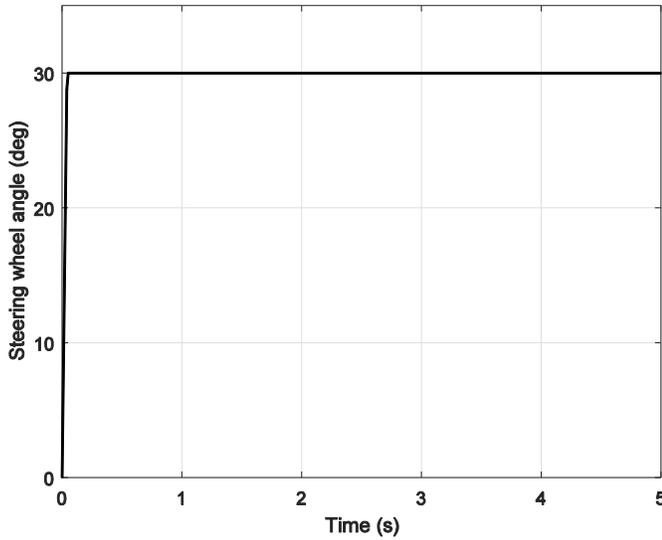


Fig. 13. Steering wheel angle in the fishhook style maneuver.

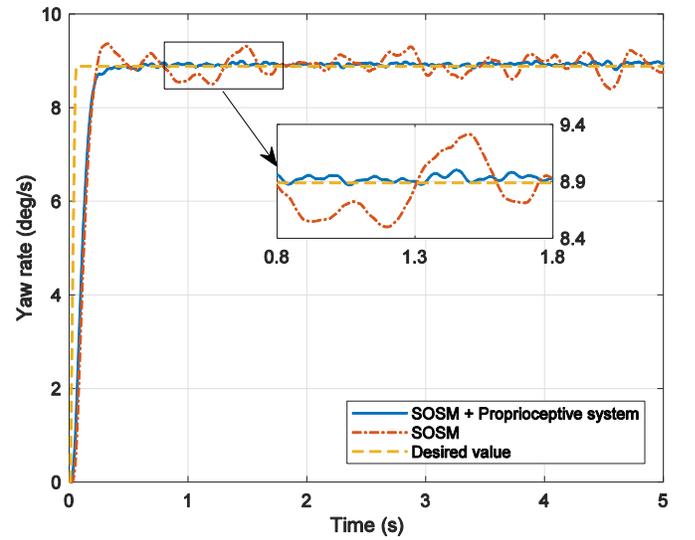


Fig. 15. Yaw rate response in the fishhook style maneuver.

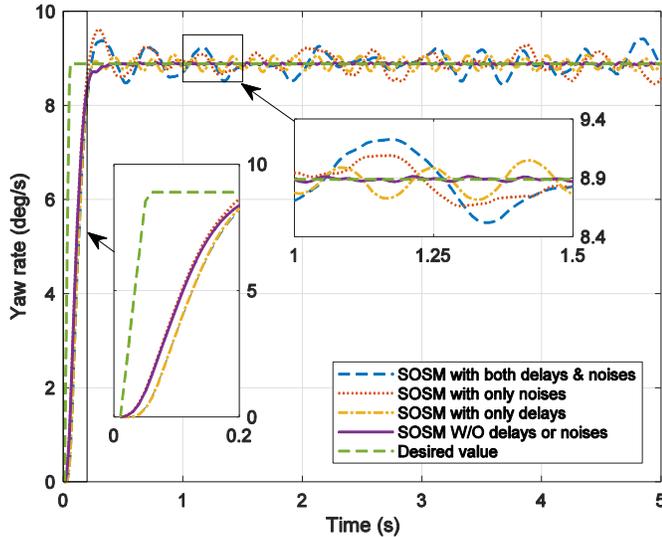


Fig. 14. Influence of delays and noises on the SOSMC yaw rate tracking controller.

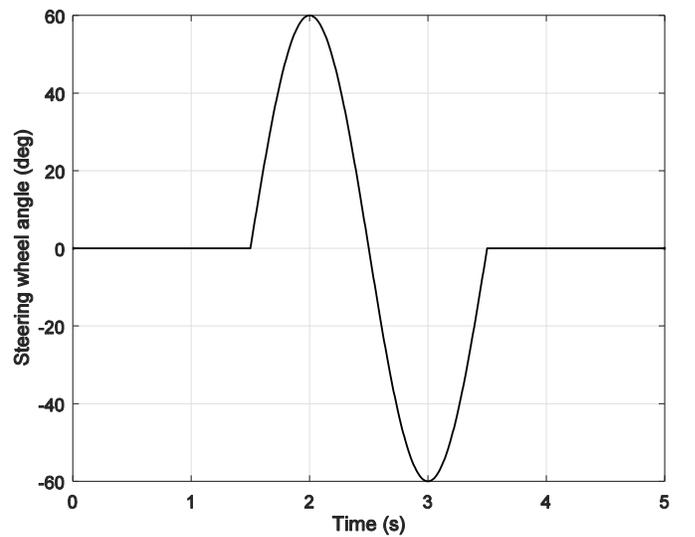


Fig. 16. Steering wheel angle in the single-lane change maneuver.

recently been found that some undesired factors in communication buses and sensors can cause the chattering problem of SOSMC [6]. Fig. 14 shows that when SOSMC is applied to the yaw rate tracking of FWIA EVs in Carsim fishhook turning style, the performance of the system can be significantly degraded by communication delays and measurement noises. The desired yaw rate value of our tracking controller is similar to the reference value in [52], and the SOSMC control law is designed as [53]

$$\begin{cases} y_1 = \dot{\Omega}_z - \dot{\Omega}_z^* \\ \Delta \dot{M}_{Zd} = -K_{SL} \text{sign}\left\{y_1 - \frac{1}{2}y_{1M}\right\} \end{cases} \quad (13)$$

where $\dot{\Omega}_z^*$ is the desired yaw rate value, y_1 is the sliding mode variable, $\Delta \dot{M}_{Zd}$ is the direct yaw moment command of the controller, K_{SL} is the control gain, and y_{1M} is the value of the last singular point of y_1 . In the simulation, the vehicle speed is set as 50 km/h, and the steering wheel angle is like

a step response, as shown in Fig. 13. The level of delays and noises adopted in the simulation are the same as those we selected for the training process. As can be seen, SOSMC can work well if we remove the undesired factors. Only small fluctuation exists due to the discrete-time sampling (sampling period is 10 ms). However, when delays and noises are added to the sensory signals separately, obvious oscillation occurs. The chattering becomes even larger when the two factors are considered at the same time. It is also easy to observe that in the first 200 ms, the yaw rate responses with delays lag behind those without delays. This lag may be less important if we compare it with the time to reduce the yaw rate error. Actually, it takes more than 200 ms for the error to approach zero. However, the time for the sliding variable to enter the sliding mode surface can be largely reduced if we further optimize our algorithm. Combining the conventional SMC with SOSMC is a popular way in terms of accelerating the convergence. SMC can push the yaw rate to the reference value in less time, and

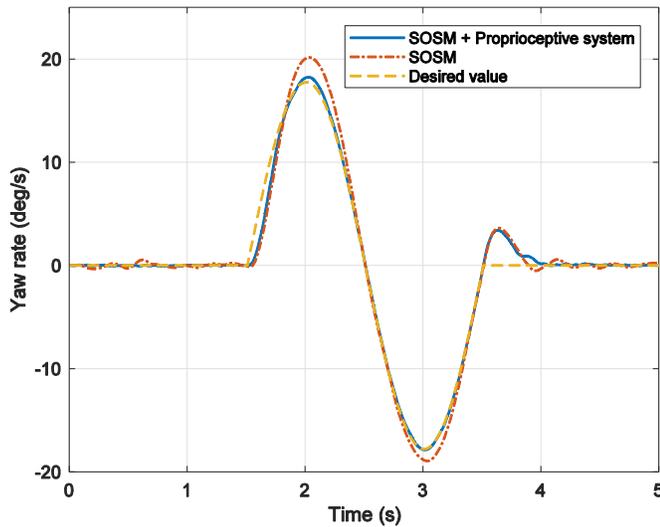


Fig. 17. Yaw rate response in the single-lane change maneuver.

TABLE II
MSE OF YAW RATE

Controller	Fishhook MSE (deg/s) ²	Single-lane change MSE (deg/s) ²
SOSMC+ proprioceptive system	0.003	0.051
SOSMC	0.041	1.883

then, the switch to SOSMC can eliminate the disadvantages of SMC. Nevertheless, the response lag resulting from delays cannot be overcome so easily.

Combining the proprioceptive system with the SOSMC has the potential to alleviate such performance degradation caused by delays and noises. In Fig. 15, we can see how the RNN and UKP work together to mitigate the chattering problem with the same simulation settings of Fig. 14. With the employment of the proprioceptive system, the maximum tracking error declines to around one-fourth of its original value. The maximum lateral acceleration of the FWIA EV is around 0.25 g in this simulation, so the vehicle still mainly demonstrates linear characteristics. To evaluate the effectiveness of our method under nonlinear conditions, we also simulated in single-lane change maneuver with larger steering wheel angle. The steering wheel angle input is given in Fig. 16. Setting the simulation speed as 50 km/h, the maximum lateral acceleration reaches 0.5 g. A performance comparison between the proprioceptive system + SOSMC and single SOSMC under this condition can be seen in Fig. 17. With the proprioceptive system, the yaw rate of the FWIA EV reaches a good tracking of the desired value after around 300 ms of the turning, whereas large overshoot occurs around the peak value of the steering wheel angle with single SOSMC. Table II gives the MSE of the yaw rate in the 0.5–5 s of the fishhook maneuver and 1.8–3.5 s of the single-lane change maneuver. With the introduction of the proprioceptive system, the MSE of yaw rate (we only consider the period after the first time the yaw rate error crosses the sliding mode surface) can decrease by an order of magnitude.

VI. CONCLUSION

A brain-inspired method, which combines DL and data fusion was proposed to overcome the feedback delays and noises in autonomous FWIA s EVs in this paper. The simulation results demonstrate that modern RNN architecture and learning method can successfully extract nonlinear dynamic models of the vehicle from sensory data, and UKP can fuse the two streams of information well. The performance of yaw rate tracking of FWIA EVs can be improved with the introduction of our proprioceptive system.

In this paper, the training was conducted offline. However, autonomous vehicles face an ever-changing environment in operation. Some parameters of vehicles, like mass, friction coefficient are not constant, and the dynamic models of vehicle motion are time-varying. As such, the online learning method which can adapt to the changes in the environment along with the learning, inference, and selection of multimodels are our goals for the future study. Besides, the delays in the feedforward loop are ignored in this paper. Although these delays are not as obvious as those in the feedback loop, they still exist and can affect the closed loop control performance. The real-time performance considers feedback delay, feedforward delay, and VCU computational delay should be carefully studied in the future.

REFERENCES

- [1] R. Wang and J. Wang, "Fault-tolerant control with active fault diagnosis for four-wheel independently driven electric ground vehicles," *IEEE Trans. Veh. Technol.*, vol. 60, no. 9, pp. 4276–4287, Nov. 2011, doi: [10.1109/TVT.2011.2172822](https://doi.org/10.1109/TVT.2011.2172822).
- [2] D. Kasinathan *et al.*, "An optimal torque vectoring control for vehicle applications via real-time constraints," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 4368–4378, Jun. 2016, doi: [10.1109/TVT.2015.2467374](https://doi.org/10.1109/TVT.2015.2467374).
- [3] K. Nam, Y. Hori, and C. Lee, "Wheel slip control for improving traction ability and energy efficiency of a personal electric vehicle," *Energies*, vol. 8, no. 7, pp. 6820–6840, 2015, doi: [10.3390/en8076820](https://doi.org/10.3390/en8076820).
- [4] R. De Castro, R. E. Araujo, and D. Freitas, "Wheel slip control of EVs based on sliding mode technique with conditional integrators," *IEEE Trans. Ind. Electron.*, vol. 60, no. 8, pp. 3256–3271, Aug. 2013, doi: [10.1109/TIE.2012.2202357](https://doi.org/10.1109/TIE.2012.2202357).
- [5] G. Yin, R. Wang, and J. Wang, "Robust control for four wheel independently-actuated electric ground vehicles by external yaw-moment generation," *Int. J. Autom. Technol.*, vol. 16, no. 5, pp. 839–847, 2015, doi: [10.1007/s12239-015-0086-2](https://doi.org/10.1007/s12239-015-0086-2).
- [6] T. Goggia *et al.*, "Integral sliding mode for the torque-vectoring control of fully electric vehicles: Theoretical design and experimental assessment," *IEEE Trans. Veh. Technol.*, vol. 64, no. 5, pp. 1701–1715, May 2015, doi: [10.1109/TVT.2014.2339401](https://doi.org/10.1109/TVT.2014.2339401).
- [7] F.-K. Wu, T.-J. Yeh, and C.-F. Huang, "Motor control and torque coordination of an electric vehicle actuated by two in-wheel motors," *Mechatronics*, vol. 23, no. 1, pp. 46–60, 2013, doi: [10.1016/j.mechatronics.2012.10.008](https://doi.org/10.1016/j.mechatronics.2012.10.008).
- [8] T. D. Chen, K. M. Kockelman, and J. P. Hanna, "Operations of a shared, autonomous, electric vehicle fleet: Implications of vehicle & charging infrastructure decisions," *Transp. Res. A Policy Pract.*, vol. 94, pp. 243–254, Dec. 2016, doi: [10.1016/j.tra.2016.08.020](https://doi.org/10.1016/j.tra.2016.08.020).
- [9] H. Kanchwala and H. Ogai, "Development of an intelligent transport system for EV," *SAE Int. J. Passenger Cars Electron. Elect. Syst.*, vol. 9, no. 1, pp. 9–21, 2016, doi: [10.4271/2015-01-9132](https://doi.org/10.4271/2015-01-9132).
- [10] N. van de Wouw, D. Nešić, and W. P. M. H. Heemels, "A discrete-time framework for stability analysis of nonlinear networked control systems," *Automatica*, vol. 48, no. 6, pp. 1144–1153, 2012, doi: [10.1016/j.automatica.2012.03.005](https://doi.org/10.1016/j.automatica.2012.03.005).

- [11] Z. Shuai, H. Zhang, J. Wang, J. Li, and M. Ouyang, "Combined AFS and DYC control of four-wheel-independent-drive electric vehicles over CAN network with time-varying delays," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 591–602, Feb. 2014, doi: [10.1109/TVT.2013.2279843](https://doi.org/10.1109/TVT.2013.2279843).
- [12] C. F. Caruntu, M. Lazar, R. H. Gielen, P. P. J. van den Bosch, and S. Di Cairano, "Lyapunov based predictive control of vehicle drivetrains over CAN," *Control Eng. Pract.*, vol. 21, no. 12, pp. 1884–1898, 2013, doi: [10.1016/j.conengprac.2012.05.012](https://doi.org/10.1016/j.conengprac.2012.05.012).
- [13] H. Zhang and J. Wang, "Robust two-mode-dependent controller design for networked control systems with random delays modelled by Markov chains," *Int. J. Control.*, vol. 88, no. 12, pp. 2499–2509, 2015, doi: [10.1080/00207179.2015.1048293](https://doi.org/10.1080/00207179.2015.1048293).
- [14] X. Zhu, H. Zhang, J. Wang, and Z. Fang, "Robust lateral motion control of electric ground vehicles with random network-induced delays," *IEEE Trans. Veh. Technol.*, vol. 64, no. 11, pp. 4985–4995, Nov. 2015, doi: [10.1109/TVT.2014.2383402](https://doi.org/10.1109/TVT.2014.2383402).
- [15] R. A. Adams, S. Shipp, and K. J. Friston, "Predictions not commands: Active inference in the motor system," *Brain Struct. Funct.*, vol. 218, no. 3, pp. 611–643, 2013, doi: [10.1007/s00429-012-0475-5](https://doi.org/10.1007/s00429-012-0475-5).
- [16] D. M. Wolpert and J. R. Flanagan, "Motor prediction," *Current Biol.*, vol. 11, no. 18, pp. R729–R732, 2001, doi: [10.1016/S0960-9822\(01\)00432-8](https://doi.org/10.1016/S0960-9822(01)00432-8).
- [17] R. Shadmehr, M. A. Smith, and J. W. Krakauer, "Error correction, sensory prediction, and adaptation in motor control," *Annu. Rev. Neurosci.*, vol. 33, no. 1, pp. 89–108, 2010, doi: [10.1146/annurev-neuro-060909-153135](https://doi.org/10.1146/annurev-neuro-060909-153135).
- [18] D. M. Wolpert and Z. Ghahramani, "Computational principles of movement neuroscience," *Nat. Neurosci.*, vol. 3, no. 11s, pp. 1212–1217, 2000, doi: [10.1038/81497](https://doi.org/10.1038/81497).
- [19] D. M. Wolpert, J. Diedrichsen, and J. R. Flanagan, "Principles of sensorimotor learning," *Nat. Rev. Neurosci.*, vol. 12, no. 12, pp. 739–751, 2011, doi: [10.1038/nrn3112](https://doi.org/10.1038/nrn3112).
- [20] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Fluids Eng.*, vol. 82, no. 1, pp. 35–45, 1960.
- [21] R. P. Rao and D. H. Ballard, "Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects," *Nat. Neurosci.*, vol. 2, no. 1, pp. 79–87, 1999, doi: [10.1038/4580](https://doi.org/10.1038/4580).
- [22] S. Vaziri, J. Diedrichsen, and R. Shadmehr, "Why does the brain predict sensory consequences of oculomotor commands? Optimal integration of the predicted and the actual sensory feedback," *J. Neurosci.*, vol. 26, no. 16, pp. 4188–4197, 2006, doi: [10.1523/JNEUROSCI.4747-05.2006](https://doi.org/10.1523/JNEUROSCI.4747-05.2006).
- [23] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: [10.1038/nature14539](https://doi.org/10.1038/nature14539).
- [24] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015, doi: [10.1016/j.neunet.2014.09.003](https://doi.org/10.1016/j.neunet.2014.09.003).
- [25] D. Cireşan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *Proc. CVPR*, Providence, RI, USA, 2012, pp. 3642–3649.
- [26] Y. Bengio, "Deep learning of representations: Looking forward," in *Proc. SLSP*, Tarragona, Spain, 2013, pp. 1–37.
- [27] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. ICCV*, Santiago, Chile, 2015, pp. 1026–1034.
- [28] I. Rivals, D. Canas, L. Personnaz, and G. Dreyfus, "Modeling and control of mobile robots and intelligent vehicles by neural networks," in *Proc. IEEE Symp. Intell. Veh.*, Paris, France, 1994, pp. 137–142.
- [29] Y. U. Yim and S. Y. Oh, "Modeling of vehicle dynamics from real vehicle measurements using a neural network with two-stage hybrid learning for accurate long-term prediction," *IEEE Trans. Veh. Technol.*, vol. 53, no. 4, pp. 1076–1084, Jul. 2004, doi: [10.1109/TVT.2004.830145](https://doi.org/10.1109/TVT.2004.830145).
- [30] A. Numakura *et al.*, "FAD learning: Separate learning for three accelerations-learning for dynamics of boat through motor babbling," in *Proc. ICRA*, Stockholm, Sweden, 2016, pp. 5609–5614.
- [31] S. Bansal, A. K. Akametalu, F. J. Jiang, F. Laine, and C. J. Tomlin, "Learning quadrotor dynamics using neural network for flight control," in *Proc. IEEE 55th CDC*, Las Vegas, NV, USA, 2016, pp. 4653–4660.
- [32] J. Sjöberg *et al.*, "Nonlinear black-box modeling in system identification: A unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995, doi: [10.1016/0005-1098\(95\)00120-8](https://doi.org/10.1016/0005-1098(95)00120-8).
- [33] 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, doi: [10.1109/72.279181](https://doi.org/10.1109/72.279181).
- [34] S. Hochreiter and J. J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1–32, 1997, doi: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735).
- [35] N. Mohajerin and S. L. Waslander, "Modular deep recurrent neural network: Application to quadrotors," in *Proc. SMC*, San Diego, CA, USA, 2014, pp. 1374–1379.
- [36] J. Martens and I. Sutskever, "Learning recurrent neural networks with Hessian-free optimization," in *Proc. ICML*, Bellevue, WA, USA, 2011, pp. 1033–1040.
- [37] X. Huang and J. Wang, "Real-time estimation of center of gravity position for lightweight vehicles using combined AKF-EKF method," *IEEE Trans. Veh. Technol.*, vol. 63, no. 9, pp. 4221–4231, Nov. 2014, doi: [10.1109/TVT.2014.2312195](https://doi.org/10.1109/TVT.2014.2312195).
- [38] C. Cheng and D. Cebon, "Parameter and state estimation for articulated heavy vehicles," *Veh. Syst. Dyn.*, vol. 49, nos. 1–2, pp. 399–418, 2011, doi: [10.1080/00423110903406656](https://doi.org/10.1080/00423110903406656).
- [39] G. L. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs—Part 3. State and parameter estimation," *J. Power Sources*, vol. 134, no. 2, pp. 277–292, 2004, doi: [10.1016/j.jpowsour.2004.02.031](https://doi.org/10.1016/j.jpowsour.2004.02.031).
- [40] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proc. F Radar Signal Process.*, vol. 140, no. 2, pp. 107–113, Apr. 1993, doi: [10.1049/ip-f-2.1993.0015](https://doi.org/10.1049/ip-f-2.1993.0015).
- [41] S. Thrun, "Particle filters in robotics," Presented at the Uncertainty AI, 2002. [Online]. Available: <http://robots.stanford.edu/papers/thrun.pf-in-robotics-uai02.pdf>
- [42] H. Shen, V. Vittaldev, C. D. Karlgaard, R. P. Russell, and E. Pellegrini, "Parallelized sigma point and particle filters for navigation problems," in *Proc. AAS*, Breckenridge, CO, USA, 2013, pp. 2013–2034.
- [43] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004, doi: [10.1109/JPROC.2003.823141](https://doi.org/10.1109/JPROC.2003.823141).
- [44] H. B. Pacejka and E. Bakker, "The magic formula Tyre model," *Veh. Syst. Dyn.*, vol. 21, pp. 1–18, Aug. 2007, doi: [10.1080/00423119208969994](https://doi.org/10.1080/00423119208969994).
- [45] J. Ryu and J. C. Gerdes, "Integrating inertial sensors with global positioning system (GPS) for vehicle dynamics control," *J. Dyn. Syst. Meas. Control*, vol. 126, no. 2, pp. 243–254, 2004, doi: [10.1115/1.1766026](https://doi.org/10.1115/1.1766026).
- [46] T. Herpel, K.-S. Hielscher, U. Klehmet, and R. German, "Stochastic and deterministic performance evaluation of automotive CAN communication," *Comput. Netw.*, vol. 53, no. 8, pp. 1171–1185, 2009, doi: [10.1016/j.comnet.2009.02.008](https://doi.org/10.1016/j.comnet.2009.02.008).
- [47] N. Mohajerin and S. L. Waslander, "Modelling a quadrotor vehicle using a modular deep recurrent neural network," in *Proc. IEEE SMC*, Kowloon, China, 2015, pp. 376–381.
- [48] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006, doi: [10.1162/neco.2006.18.7.1527](https://doi.org/10.1162/neco.2006.18.7.1527).
- [49] I. Kant and N. K. Smith, *Critique of Pure Reason*, New York, NY, USA: St. Martin's Press, 1965, p. 93.
- [50] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proc. ASSPCC*, Lake Louise, AB, Canada, 2000, pp. 153–158.
- [51] S. T. Sarkka, "Recursive Bayesian inference on stochastic differential equations," Ph.D. dissertation, Dept. Elect. Eng., Helsinki Univ Tech., Espoo, Finland, 2006.
- [52] P. He and Y. Hori, "Optimum traction force distribution for stability improvement of 4WD EV in critical driving condition," in *Proc. AMC*, Istanbul, Turkey, 2006, pp. 596–601.
- [53] M. Canale, L. Fagiano, A. Ferrara, and C. Vecchio, "Vehicle yaw control via second-order sliding-mode technique," *IEEE Trans. Ind. Electron.*, vol. 55, no. 11, pp. 3908–3916, Nov. 2008, doi: [10.1109/TIE.2008.2003200](https://doi.org/10.1109/TIE.2008.2003200).



Yanjun Li was born in Suzhou, China, in 1991. He received the B.S. degree in mechanical engineering and automation and M.S. degree in vehicle engineering from the School of Mechanical Engineering, Southeast University, Nanjing, China, in 2014 and 2017, respectively.

He is currently a Visiting Scholar with the Biomechanics and Intelligent Robotics Laboratory, City College of New York, New York, NY, USA, researching on the lower-limb exoskeleton project. His current research

interests include dynamics and control of mobile robots and vehicles, design of actuation systems, and human–robot interaction control.



Guodong Yin (M'15) received the Ph.D. degree in vehicle engineering from Southeast University, Nanjing, China, in 2007.

From 2011 to 2012, he was a Visiting Research Scholar with the Department of Mechanical and Aerospace Engineering, Ohio State University, Columbus, OH, USA. He is currently a Professor with the School of Mechanical Engineering, Southeast University. His current research interests include vehicle dynamics and control, connected vehicles, and multiagent control.



Jinxiang Wang received the B.S. degree in mechanical engineering and automation and the Ph.D. degree in vehicle engineering from Southeast University, Nanjing, China, in 2002 and 2010, respectively.

From 2014 to 2015, he was a Visiting Research Scholar with the Department of Mechanical and Aerospace Engineering, Ohio State University, Columbus, OH, USA. He is currently an Associate Professor with the School of Mechanical Engineering, Southeast University. His current

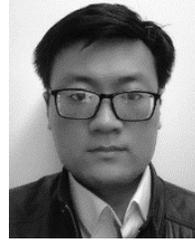
research interests include vehicle dynamics and control, and control on autonomous vehicles.



Weichao Zhuang received the B.S. and Ph.D. degrees in mechanical engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2012 and 2017, respectively.

From 2014 to 2015, he was a Visiting Student with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI, USA. He is currently an Assistant Professor with the School of Mechanical Engineering, Southeast University, Nanjing. His current research interests include vehicle dynamics and control, optimal control, clean

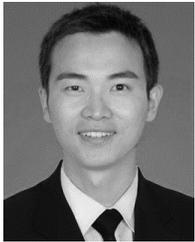
energy vehicles, connected vehicles, and multiagent control.



Keke Geng received the B.S. degree in mechanical engineering from the Nanjing University of Technology and Engineering, Nanjing, China, in 2010, and the M.S. and Ph.D. degrees in system control and information processing from the Department of Information and Control Systems, Bauman Moscow State Technical University, Moscow, Russia, in 2013 and 2017, respectively.

He is currently an Assistant Professor with the School of Mechanical Engineering, Southeast University, Nanjing. His current research interests

include autonomous motion control of vehicles, intelligent environmental awareness, integrated navigation system, and the stability of complex control systems.



Ning Zhang received the doctoral degree in mechanical engineering from the Darmstadt University of Technology, Darmstadt, Germany, in 2015.

He is currently a Research Staff with the School of Mechanical Engineering, Southeast University, Nanjing, China. He has published over 30 peer-reviewed journal and conference papers. His current research interests include vehicle system dynamics and control, ADAS and intelligent vehicles, intelligent transportation system, and sound and vibration.

Dr. Zhang is an active member of the International

Association for Vehicle System Dynamics.