IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Dual Deep Neural Network based Adaptive Filter for Estimating Absolute Longitudinal Speed of Vehicles

JONG HAN KIM AND SANG WON YOON, (Senior Member, IEEE)

Department of Automotive Engineering, Hanyang University, Seoul 04763, South Korea

Corresponding author: Sang Won Yoon (swyoon@hanyang.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) under Grant 2020R1F1A1069925 and 2020R1A4A4079701.

ABSTRACT This study employs a dual deep neural network (D-DNN) to accurately estimate the absolute longitudinal speed of a vehicle. Accuracy in speed estimation is crucial for vehicle safety, because longitudinal speed is a common parameter employed as a state variable in active safety systems such as anti-lock braking system and traction control system. In this study, DNNs are applied to determine the gain of an adaptive filter to estimate vehicle speed. The used data consists of longitudinal acceleration, wheel speed, filter gain, and estimated vehicle speed. The data generated from Carsim software are collected and preprocessed using a Simulink model. To acquire data with numerous wheel slip patterns, various acceleration and deceleration conditions are applied to four different road conditions. Though, it is challenging to achieve a single DNN model that is optimally cope with the various driving situations. Thus, we adopt two DNN models that were individually trained in low and high acceleration regions. The dual DNN model results in error reductions of 74% and 65%, compared with a single DNN and classical adaptive Kalman filter approaches, respectively.

INDEX TERMS Adaptive filter, deep neural network, slip ratio, vehicle speed estimation

I. INTRODUCTION

Active safety technologies are used in most vehicles. Typical examples are adaptive cruise control, autonomous emergency braking system, an anti-lock braking system (ABS), and a traction control system (TCS). Precise information on vehicle state is crucial for these systems [1]. For example, as the slip ratio is a controlled variable in ABS and TCS [2], an accurate longitudinal speed is required to determine the slip ratio. Maintaining a slip ratio in the desired region is essential for vehicle safety and performance because it allows the wheel to sustain a friction coefficient with the road surface above a certain level. In other words, the performance and safety of the vehicle can be improved by enhancing the accuracy of its longitudinal speed data.

However, the longitudinal speed of a vehicle is complex or expensive to measure [3–4], additional sensors such as radar may be required [5]. Its estimation rather than measurement is therefore sometimes preferred. Researches related to vehicle speed estimation have primarily been conducted using either indirect or direct methods.

Indirect methods based on observers or Kalman filters uses precise vehicle models and dynamics to determine a filtering method that estimates longitudinal speed. In the observer-based method, vehicle dynamics and a precise vehicle model were used to design the state observer. In a previous study, yaw-and-roll models of a vehicle were employed for switching observer schemes, and a full-state observer was used in speed estimation [6]. Another study utilized a six-degrees-of-freedom vehicle model with a second-order sliding-mode observer [7]. The Kalman-filter-based model also requires accurate vehicle parameters and a complex vehicle model. A nine-degrees-of-freedom bicycle model and extended Kalman filter (EKF) have been presented [1]. In addition, an unknown input Kalman filter was applied with a nonlinear tire model to estimate vehicle speed [8]. A mixed EKF algorithm was applied to a six-degrees-of-freedom vehicle model with longitudinal and lateral velocities, yaw rate, and four-wheel rotational speeds as state vector [9]. Although indirect methods can estimate the speed of a vehicle with high accuracy, they are time-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040733, IEEE Access

consuming and complex because they require a complex vehicle model. This requirement refers that all vehicle parameters must be accurately determined to develop a reliable speed-estimation algorithm in various vehicle models and road conditions.

The direct method requires a relatively simple vehicle model, and thus, does not need to identify all vehicle parameters. This method is generally classified by the diversity of used sensor data. A typical combination is wheel speed data and/or longitudinal acceleration data. The vehicle speed was estimated using only wheel speed data applied to an adaptive nonlinear filter [10]. In this work, an experimental algorithm was used to estimate the true speed of the vehicle during braking. This approach has the advantage that it only uses wheel speed sensor data. However, when the data are noisy, the estimation result is not reliable. Additively, data on both wheel speed and longitudinal acceleration were used [11–13] to estimate vehicle speed. These studies focused on designing the best filter algorithms for vehicle speed estimation, by controlling the weights of the wheel speed and longitudinal acceleration, in given vehicle driving conditions. A gain-tuning algorithm based on the Kalman filter [11] sets the weight of the wheel speed data to zero under certain conditions, so that only the longitudinal acceleration is used when a slip occurs. Fuzzy logic can also be applied to vehicle speed estimation [12, 13]. Fuzzy logic was employed to determine the P, Q, and R matrices of the Kalman filter according to wheel speed and longitudinal acceleration data [12]. Another method using fuzzy logic has been proposed to directly determine the weights of the wheel speed, longitudinal acceleration, and previous estimates [13]. All these studies presented accurate estimation results using weight-tuning algorithms. However, these algorithms are heavily dependent on experiments and experiences, implying that it is difficult to define an algorithm that will reliably enable optimized performance in all driving situations.

Recently, neural network techniques have been vigorously applied to estimating various vehicle states [14–21]. Neural network structures can overcome the limitations of empirical methods because they can identify data characteristics that a human observer cannot detect [22]. Simple deep neural network was applied for predict the vehicle sideslip angle [14]. Another study employed a feedforward neural network with a fully connected model to estimate road grade and vehicle mass [15]. Depending on the purpose, neural network models have been adopted in various forms [16-18]. Time variant data were processed by a time-delayed neural network in a sideslip angle estimation [16]. An integrated time-series model based on a multivariate deep recurrent neural network with long short-term memory was demonstrated to estimate vehicle brake pressure [17]. Another effort attempted to solve the lateral state estimation problem of a preceding target vehicle using multiple neural networks, consisting of a nonlinear autoregressive exogenous model net, feedforward net, and Elman net [18]. In addition, hybrid approaches, combining a neural network with other methods, have been applied to estimate vehicle states [19-21]. Vehicle roll dynamics based unscented Kalman filter coupled with an artificial neural network provided a good estimation of vehicle roll angles [19]. A recurrent neural network combined with a vehicle kinematic model was trained using simulation data to estimate vehicle sideslip angles [20]. A principal component analysis was adopted for the preprocessing of input data of a neural network to estimate vehicle sideslip angles [21]. These methods reportedly lowered computational loads with increased accuracy.

A neural network has also been applied to a longitudinal speed estimation problem [23], using a simple regression method with a feedforward net. The study presented promising estimation performance using wheel speed and longitudinal acceleration data. However, the results were only evaluated using data derived from limited road-surface conditions. Moreover, the simple regression technique is disadvantageous in that the range to be estimated is too wide.

In this study, a neural network is not used to estimate vehicle speed directly. Instead, two deep neural networks (DNNs) are employed to accurately estimate the gain of an adaptive filter. The dual DNN structure was evaluated in four road conditions (dry, wet, snowy, and icy) and two vehicle situations (ABS on and off). The results were compared with the accuracy achieved with either a single DNN structure or a conventional adaptive Kalman filter (AKF).

Our proposed approach is the first attempt to estimate vehicle longitudinal speed using two DNNs with an adaptive filter, which has generally not been used for this purpose. In our dual DNN (D-DNN) approach, the training data are sorted by their accelerations and used to train each of the two DNN models. Owe to this, one of the two DNN models is optimized to the data having low accelerations while the other is optimized to high-acceleration data. Thus, it is anticipated to achieving superior optimized models compared with using only one DNN model, named as a single DNN.

The overall structure of this study is depicted in Fig. 1. Wheel speed (V_w^k) and longitudinal acceleration (A_x^k) data are acquired using Carsim software and preprocessed using a low-pass filter. The preprocessed data is stored in data storage with the estimated speed ($\hat{V}_x^k$) and filter gain (K^{k-1}) of the previous step. They are stacked in a time order to be the input feature ($\boldsymbol{u}^k$) for the DNN. The features are sorted by the amplitude of the vehicle acceleration ($|A_x^k|$) and individually used to generate the two DNN models optimized at high and low accelerations. The dual DNN derives the adaptive filter gain (K^k) as its output. The adaptive filter structure is designed to function as a state observer. The adaptive filter receives an estimated longitudinal speed of the previous step ($\hat{V}_x^k$), current longitudinal acceleration (A_x^k), current wheel speed (V_w^k) and the gain (K^k). The filter outputs the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040733, IEEE Access

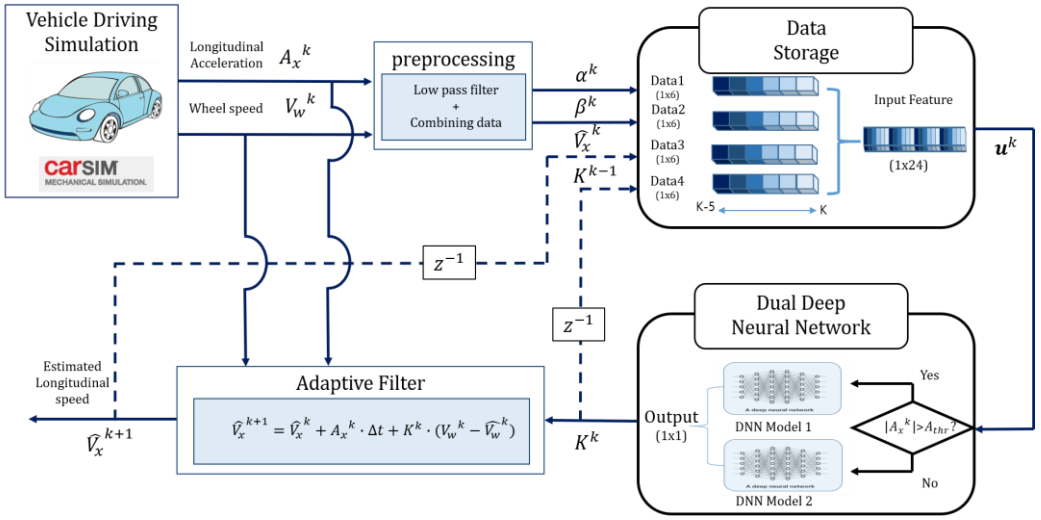


FIGURE 1. Overall structure of this work

longitudinal speed of a vehicle ($\hat{V}_x^{k+1}$) as adaptively determined by the filter gain. The proposed DNN model is trained using 640,000 samples and evaluated using 320,000 samples.

II. DESIGN OF ADAPTIVE FILTER AND DEEP NEURAL NETWORK

This section describes the overall system design procedure. First, the structure of the adaptive filter is introduced. Then, the characteristics of the desired gain and the method for determining the target gain are explained. The DNN models used in this study are introduced, including the DNN training process. The DNN output corresponds to the current filter gain K^k, which is compared with the target gain of the current step $\tilde{K}^k$. The mean square error between the K^k and $\tilde{K}^k$ is used as the loss function to update the weights and biases of each DNN model. The structure of the DNN model is explained in the last subsection.

A. ADAPTIVE FILTER: FILTER STRUCTURE

The adaptive filter derives the estimated speed by receiving the wheel speed and longitudinal acceleration as input. It also uses the estimated speed and filter gain of the previous step. The filter equations employed in this study are:

$$\hat{V}_x^{k+1} = \hat{V}_x^k + A_x^k \cdot \Delta t + K^k \cdot (V_w^k - \hat{V}_w^k), \quad (1)$$

$$\hat{V}_w^k = H \cdot \hat{V}_x^k \quad (H = 1), \quad (2)$$

$$V_w^k = r_{eff} * \omega_w^k \quad (3)$$

where $\hat{V}_x^{k+1}$ and $\hat{V}_x^k$ are the estimated speeds of the current and previous steps, respectively. Δt represents a sampling time of 0.01s and A_x^k is the measured longitudinal acceleration at time step k. K^k indicates the current filter gain, and V_w^k is the measured wheel speed, which is a product of the measured wheel angular speed ω_w^k and efficient wheel radius r_{eff}. The $\hat{V}_w^k$ is the estimated speed of the front right wheel.

B. ADAPTIVE FILTER: TARGET GAIN

As shown in Eq. (1), A_x^k and V_w^k are used to estimate vehicle speed. The magnitude of the current filter gain determines whether the measurement of A_x^k or V_w^k is reliable. If $|K^k|$ is low, A_x^k is highly weighted, which means the integration value of the acceleration is primarily used to obtain the longitudinal speed. Conversely, when the $|K^k|$ is high, V_w^k is more reliable than A_x^k, and thus, the wheel speed is principally used.

The reliability of A_x^k or V_w^k is determined by the wheel slip condition. The wheel slip ratio (λ) is defined in Eq. (4) to denote the difference between the vehicle and wheel speeds.

$$\lambda = (V_x^k - V_w^k)/max\,(V_x^k, V_w^k), \quad (4)$$

When a wheel slip occurs, the gain should be lowered, because the wheel speed is inaccurate, and the longitudinal acceleration is more reliable. Alternatively, under non-slip conditions, the wheel speed is almost the same as the vehicle speed and the integration of acceleration has a cumulative error, and thus, the gain should be elevated.

Equation (5) shows the estimated speed updated by the integration value of A_x^k. Note that the cumulative error can be explained by measurement noise.

$$\hat{V}_a^{k+1} = \hat{V}_a^k + A_x^k \cdot \Delta t \quad (5)$$

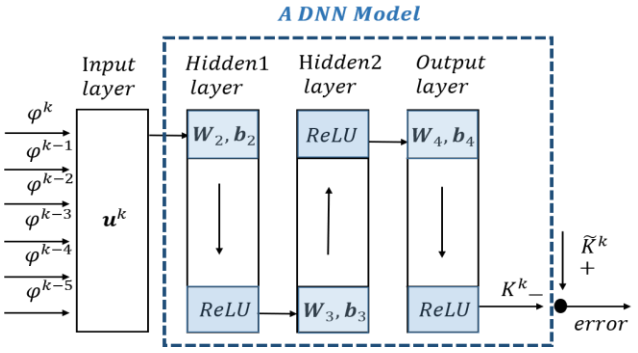
Here, the true speed value of the next step, V_x^{k+1}, can be acquired from offline data. Therefore, the target gain value of the current step $\tilde{K}^k$ can be derived by Eq. (6), following Eq. (1). $\tilde{K}^k$ is used as the desired value of the DNN output, which estimates the true speed.

$$\tilde{K}^k = (V_x^{k+1} - \hat{V}_x^k + A_x^k \cdot \Delta t)/(V_w^k - \hat{V}_w^k) \quad (6)$$

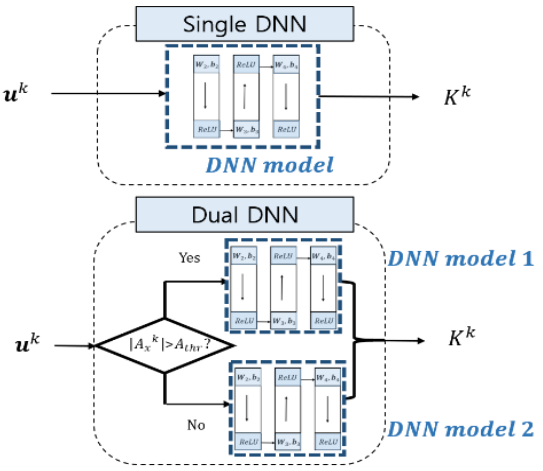
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040733, IEEE Access

C. DEEP NEURAL NETWORK: TRAINING PROCEDURES

A DNN is an artificial neural network comprising an input layer, hidden layers, and an output layer, and is equipped with weights, biases, and activation functions such as a rectified linear unit (ReLU) [24]. Figure 2(a) depicts the simplified DNN structure used in this study. As shown in the figure, two hidden layers are used. The numbers of input and output data items are 24 and 1, respectively, and the number of nodes in the hidden layers is discussed in the next subsection. The DNN input is denoted as $\boldsymbol{u}^k$ and comprises six consecutive steps of $\boldsymbol{\varphi}^k$, which consists of α^k, β^k, K^k, and $\hat{V}_x^{\,k}$. The total number of DNN input data items for each sample is therefore 24. One DNN structure in Fig. 2(a) is employed in the single DNN shown in Fig. 2(b), while two DNN structures, labeled by DNN model 1 and 2, are adopted in the dual DNN.



(a)



(b)

FIGURE 2. A simplified structure of the used DNN structure and diagram of single and dual DNN model

$$\boldsymbol{u}^k = [\boldsymbol{\varphi}^{k-5}, \boldsymbol{\varphi}^{k-4}, \boldsymbol{\varphi}^{k-3}, \boldsymbol{\varphi}^{k-2}, \boldsymbol{\varphi}^{k-1}, \boldsymbol{\varphi}^k] \quad (7)$$

$$\boldsymbol{\varphi}^k = [\alpha^k, \beta^k, K^{k-1}, \hat{V}_x^{\,k}] \quad (8)$$

$$\alpha^k = \hat{V}_w^{\,k} - V_w^{\,k} \quad (9)$$

$$\beta^k = A_x^{\,k} - A_w^{\,k} \quad (10)$$

In these equations, α^k represents the error between the estimated wheel speed $\hat{V}_w^{\,k}$ and the measured wheel speed $V_w^{\,k}$. β^k is the error between the derivative of the wheel speed $A_w^{\,k}$ and $A_x^{\,k}$. As wheel speed and vehicle longitudinal speed differ when wheel slip occurs, both α^k and β^k values become higher compared with non-slip conditions. As a consequence, α^k and β^k are useful measures representing slip situations, and they are selected as input variables to the DNN.

The DNN training procedure can be divided into two steps: feedforward and backward. In feedforward, the output of DNN K^k is computed using the activation function, weights and biases given in Eq. (11).

$$K^k = f(\boldsymbol{W}_4 \cdot f(\boldsymbol{W}_3 \cdot f(\boldsymbol{W}_2 \cdot \boldsymbol{u}^k + \boldsymbol{b}_2) + \boldsymbol{b}_3) + \boldsymbol{b}_4) \quad (11)$$

$\boldsymbol{W}_i$ and $\boldsymbol{b}_i$ (I = 2, 3, 4) indicate the weights and biases of the hidden and output layers. In this study, the ReLU activation function is applied to all layers, and f in Eq. (11) represents the ReLU function. In backward process, loss function is selected as the mean squared error of gain (MSEG) between the output of DNN K^k and the target gain $\tilde{K}^k$.

$$MSEG = \frac{1}{n}\sum_{k=1}^{n}(\tilde{K}^k - K^k)^2. \quad (12)$$

The loss function is minimized by updating the weights of layers. The *RMSprop* optimizer is applied to the update. After the backward process, the adaptive filter derives the estimated speed $\hat{V}_x^{\,k+1}$ using the output of the DNN, K^k. Additionally, the mean squared error of speed (MSES) is derived to define the speed estimation performance.

$$MSES = \frac{1}{n}\sum_{k=1}^{n}(V_x^{\,k} - \hat{V}_x^{\,k})^2. \quad (13)$$

Finally, for each training step, the estimated speed $\hat{V}_x^{\,k+1}$ and gain of filter K^k are stored for the next training procedure. Once an over-fitting is observed, the training procedure is reset and re-initialized using the He initialization method.

D. DEEP NEURAL NETWORK: STRUCTURE SELECTION

Several DNN structures are examined to find the optimal structure for the data used in this study. Table I compares the MSES of eight DNN structures after training. The quantity of nodes in the input and output layers is the same for all structures. Structure 5 is selected because the MSES value is lowest.

TABLE I
COMPARISON OF TRAINING MSES'S FOR EIGHT DNN STRUCTURES

	Shape of the structure (input, hidden1, hidden2, output)	MSES
Structure 1	(24,12,12,1)	1.001
Structure 2	(24,24,24,1)	1.730
Structure 3	(24,24,12,1)	2.873
Structure 4	(24,36,24,1)	0.403
Structure 5	(24,36,12,1)	0.106
Structure 6	(24,48,36,1)	0.393
Structure 7	(24,48,24,1)	0.113
Structure 8	(24,28,12,1)	0.895

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040733, IEEE Access

III. DATA ACQUISTION CONDITIONS

Wheel speed and longitudinal acceleration data are acquired considering the generalized performance of the selected DNN structure. The generalized performance indicates a proper operation even in unlearned circumstances and is crucial when applied to vehicle control and estimation systems. This requires training the DNN with the various slip data features that may occur when driving. For this purpose, data collection was conducted in three simulation conditions that can affect vehicular wheel slip: acceleration, road, and ABS conditions.

A. ACCELERATION CONDITIONS

Vehicular wheel slip is affected by the magnitude of acceleration and deceleration even on the same road condition. Thus, the simulated vehicle is exposed to various acceleration and deceleration conditions by setting its desired speed as shown in Fig. 3. Note that the slope in the figure changes with every cycle, indicating the acceleration and deceleration also changes. Figure 4 shows the wheel speed V_w^k and true speed V_x on a wet surface following the target speed scenario in Fig. 3. If the gradient of V_x (which is proportional to A_x^k) exceeds a certain value, the difference between the true and wheel speeds increases, indicating the occurrence of wheel slip.

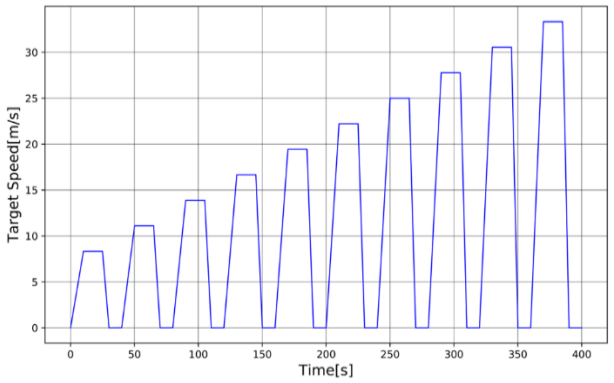


FIGURE 3. Desired longitudinal speed of the test vehicle to be exposed to various acceleration and deceleration conditions

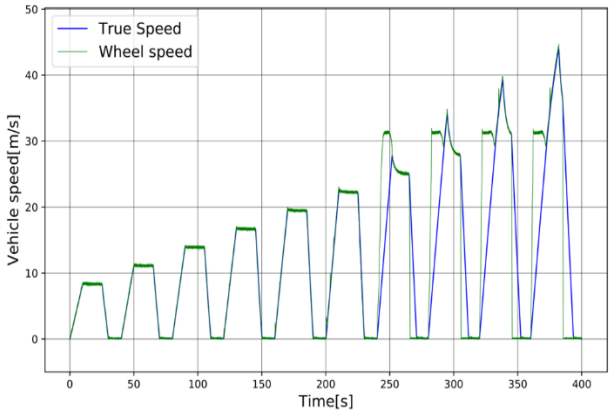


FIGURE 4. Comparison of true speed and wheel speed. The difference between the blue and green data represents the wheel slip.

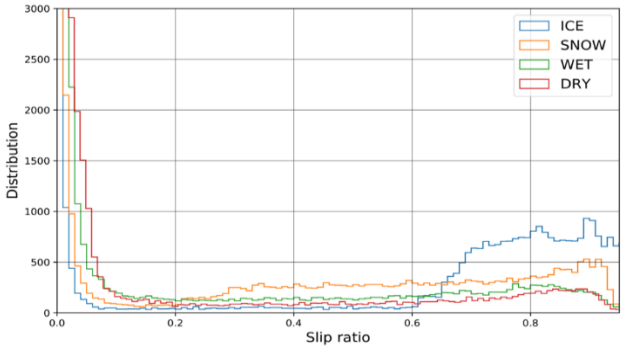


FIGURE 5. Slip ratio distributions of four road surface conditions

B. ROAD CONDITIONS

Road conditions also affect the vehicular wheel slip. To achieve a generalized performance, the DNN should be able to function in various road conditions. We selected four normal road conditions [25]: ice-covered, snow-covered, wet, and dry roads. The road friction coefficient of each road condition is set to 0.2 (ice), 0.3 (snow), 0.5 (wet), or 0.85 (dry). Figure 5 depicts the distribution of the slip ratio in the different road conditions following the target speed scenario in Fig. 3. Each road condition has 40,000 data points. The blue, yellow, green, and red lines represent the slip ratio distribution on icy, snowy, wet, and dry roads, respectively. For icy roads, which has the lowest road friction coefficient, the data are heavily distributed in a high slip-ratio region. This means that slip occurs more easily on icy roads than on other road conditions. Conversely, on the dry road, the data are distributed in the low slip-ratio region, meaning that the slip occurrence is relatively rare.

C. ANTI-LOCK BRAKING SYSTEM CONDITIONS

The features of A_x^k and V_w^k differ depending on whether or not ABS is activated, because the ABS controller activates the brake actuator to follow the desired slip [26]. In addition, as A_x^k and V_w^k are entered as DNN input, the DNN must be able to correctly respond to given situations. Therefore, we collected data when the ABS was on or off.

The dataset for training and testing (Table II) is divided into eight cases for the four road surfaces and two ABS conditions. Each case consists of 120,000 data samples, of which 80,000 are used for training and the left 40,000 samples are used for testing.

As an example, the "D-F" datasets in the table are acquired when a vehicle drives on a dry road condition (initialed by "D") with ABS-off (abbreviated by "F") following the speed scenario in Fig. 3. Likewise, the data obtained on a snowy road condition (initialed by "S") with ABS-on (abbreviated by "N") are marked as "S-N" in the table.

TABLE II
DATASET FOR TRAINING AND TESTING

	Dry	Wet	Snow	Ice
ABS off	D-F	W-F	S-F	I-F
ABS on	D-N	W-N	S-N	I-N

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040733, IEEE Access

IV. SPEED ESTIMATION RESULTS

After the training, the developed DNN model is validated using test data. For comparison purposes, the dual DNN is compared with a single DNN. In addition, the estimation results using a conventional AKF are also compared as a reference, because it is a typical case of an empirically determined indirect speed estimation method.

A. SPEED ESTIMATION USING A SINGLE DNN

Figure 6 shows the process of speed estimation using a single DNN. The procedure is the same as explained in Fig. 1, except the single DNN is employed instead of a dual DNN. The DNN input $\boldsymbol{u}^k$ is derived from the data storage, and the DNN computes the filter gain K^k using Eq. (11). The adaptive filter determines the estimated speed using Eq. (1). K^k and estimated speed $\widehat{V}_x^{k+1}$ are stored in the data storage for the next step. In the figure, $\boldsymbol{x_1}^k$ is the vector of data that acts as the input to the data storage and $\boldsymbol{x_2}^k$ is the longitudinal acceleration and wheel speed data that are used in the adaptive filter for speed estimation.

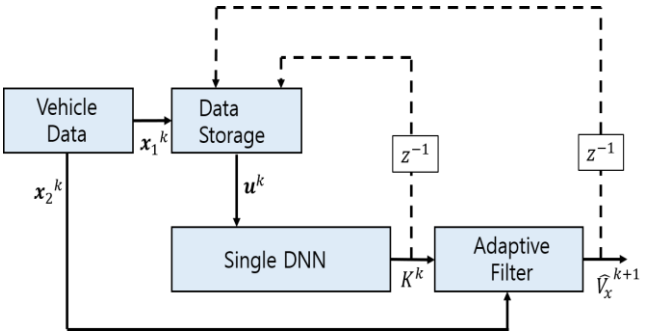


FIGURE 6. Speed estimation procedure using a single DNN

$$\boldsymbol{x_1}^k = [A_x{}^k, V_w{}^k, A_w{}^k],\ \boldsymbol{x_2}^k = [A_x{}^k, V_w{}^k]. \qquad (14)$$

The estimated speed and filter gain are illustrated in Fig. 7. This is the estimation results of structure 5 (in Table I) using the test data generated from the wet and ABS-off condition (W-F in Table II). The blue, red, green, and yellow lines represent the vehicle's true speed, estimated speed using the single DNN, wheel speed, and integration of longitudinal acceleration obtained by Eq. (5), respectively. As shown in Fig. 7, a slip occurs during acceleration and deceleration situations, where the true speed (blue) and wheel speed (green) significantly differ. This judgment is based on Eq. (4). The gain of filter K^k (black) is shown to be almost zero in the slip regions, meaning that no weighting is applied to the wheel speed data. Conversely, in the non-slip region, the K^k has a value between zero and one, which implies that the DNN is now applying a higher weight to $V_w{}^k$ than to $A_x{}^k$.

Specific regions of Fig. 7, which are specified as A and B, are enlarged and depicted in Fig. 8. It is clear that the wheel slip occurs at the acceleration and deceleration conditions. In the non-slip regions, the yellow graph, calculated by Eq. (5), drifts over time and is obviously

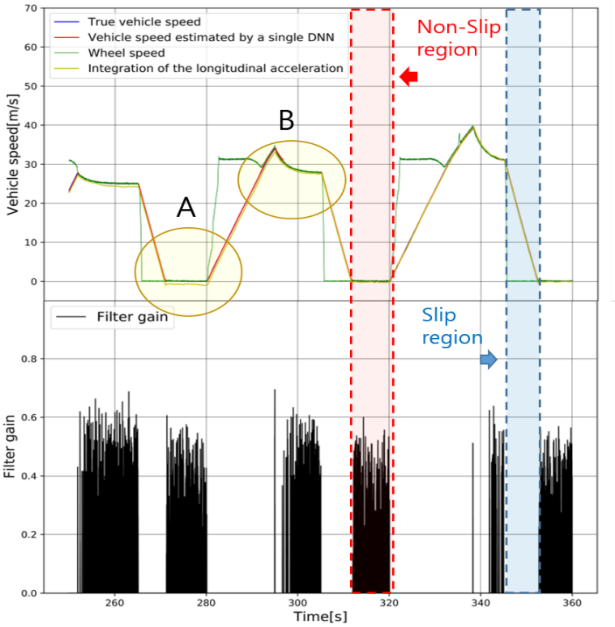


FIGURE 7. Speed estimation results and corresponding filter gains from a single DNN. The red- and blue-boxed regions indicate examples of the slip and non-slip regions, respectively. The focused A and B regions are further explained in Fig. 8.

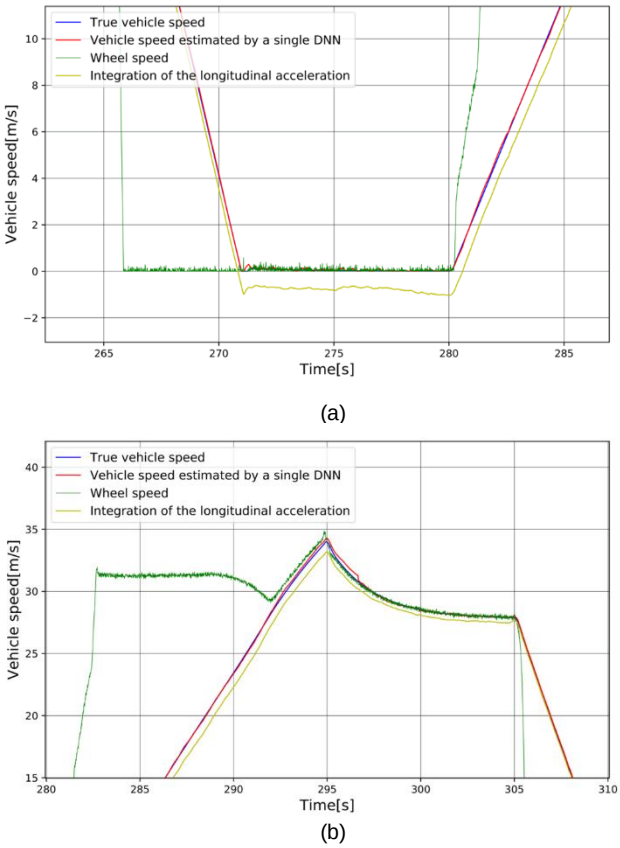


FIGURE 8. Enlarged images of speed estimation results from Fig. 7, noted as (a) region A and (b) region B

different from other three speed values. It is because of the accumulated error stemming from the integration of the acceleration $A_x{}^k$. However, the red line, showing the $\widehat{V}_x{}^k$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040733, IEEE Access

estimated by the single DNN, follows the true vehicle speed closely across entire regions. In addition, the estimated speed has less noise than the wheel speed V_w^k. These observations result in a low average test MSES of 0.293.

B. SPEED ESTIMATION USING A DUAL DNN

Although the single DNN provides a relatively high degree of estimate accuracy, it would be challenging for one DNN model to optimally handle diverse vehicle driving conditions. This motivated the introduction of the D-DNN approach employing two different DNN models. The key strategy is the threshold value of the current longitudinal acceleration, A_{thr}, classifying current vehicle states.

Figure 9 shows the estimation procedure considering the threshold A_{thr}. If the absolute value of A_x^k is higher than the threshold value, DNN1 is employed as a gain-tuning model. Otherwise, DNN2 is used. This rule is applied to training as well as test processes. The structure of each DNN is the same as that of a single DNN described in the previous section.

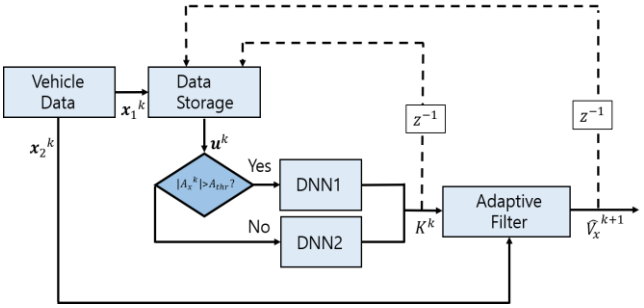


FIGURE 9. **Speed estimation procedure using a dual DNN**

The longitudinal acceleration threshold value is selected empirically. Table III shows the test MSES of the eight data conditions, according to several A_{thr} values. Based on the observations, the threshold value is set at 0.2, because the average MSES value is lowest.

TABLE III
TEST MSES'S OF DIFFERENT ACCELERATION THRESHOLDS

A_{thr} / Test data	None	0.2	0.5	0.7	1.0	2.0
D-F	0.388	0.016	0.014	0.014	0.015	0.050
W-F	0.291	0.020	0.022	0.024	0.022	0.130
S-F	0.262	0.203	3.463	0.346	2.176	0.331
I-F	0.094	0.091	0.094	0.091	1.278	0.367
D-O	0.389	0.015	0.014	0.014	0.014	0.055
W-O	0.394	0.021	0.021	0.034	0.020	0.133
S-O	0.434	0.154	0.158	0.078	0.410	0.198
I-O	0.095	0.089	0.143	0.117	0.384	1.556
Average	0.293	0.076	0.491	0.089	0.539	0.352

C. COMPARISION OF SPEED ESTIMATION ERRORS

Figure 10 compares the speed estimation errors of the three methods. The green line indicates the AKF method. An adaptive filter using a single DNN is visualized using a red line. The blue line represents the proposed model: an adaptive filter using a D-DNN. The figure includes errors of all eight considered conditions (four road surfaces in two ABS conditions). The speed estimation error is derived using the following equation:

$$error^k = \left| V_x^k - \hat{V}_x^k \right| \tag{15}$$

All three methods are found to make accurate estimations. Most error values are less than 0.3 m/s. More specifically, 81.81%, 75.02%, and 73.13% of the error points are less than 0.3 m/s in the dual DNN, single DNN, and AKF methods, respectively. As Fig. 10 shows, in all eight conditions, the error of the AKF method often rises rapidly, compared with the other two methods. The AKF and single DNN methods produce larger errors when the ABS is on, compared with conditions in which the ABS is off. These results imply that the two methods may be more dependent on vehicle driving conditions.

Conversely, when using the D-DNN method, whether the ABS is on or off does not change the errors notably. Another advantage is that the error value is noticeably small and rarely rises suddenly. Consequently, our proposed model is expected to offer superior accuracy.

Table IV compares the test MSES values of the three methods used to estimate vehicle speed. Our proposed model has the lowest MSES for all test conditions. The D-DNN greatly reduces the average MSES value by 74% and 65% compared with a single DNN and AKF, respectively, confirming that the proposed dual DNN model provides a generalized performance with a high degree of accuracy for vehicle speed estimation.

TABLE IV
TEST MSES VALUES FOR THE THREE COMPARED METHODS

	Single DNN	Dual DNN	AKF
D-F	0.388	0.016	0.035
W-F	0.291	0.020	0.233
S-F	0.262	0.203	0.3464
I-F	0.094	0.091	0.094
D-O	0.389	0.015	0.042
W-O	0.394	0.021	0.2625
S-O	0.434	0.154	0.3624
I-O	0.095	0.089	0.3698
Average	0.293	0.076	0.218

VI. CONCLUSION

In this study, a filter-gain tuning method using a dual DNN to estimate the absolute speed of a vehicle was proposed. The generalized performance and high degree of accuracy of the resulting speed estimation were achieved with a trained D-DNN model. Although our proposed method produced accurate estimation of longitudinal speeds, the trained model was optimized for the datasets used in this study. This implies that the trained model presented here could be trained further using additional datasets, generated from different road or driving conditions, and become a more generalized model.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040733, IEEE Access

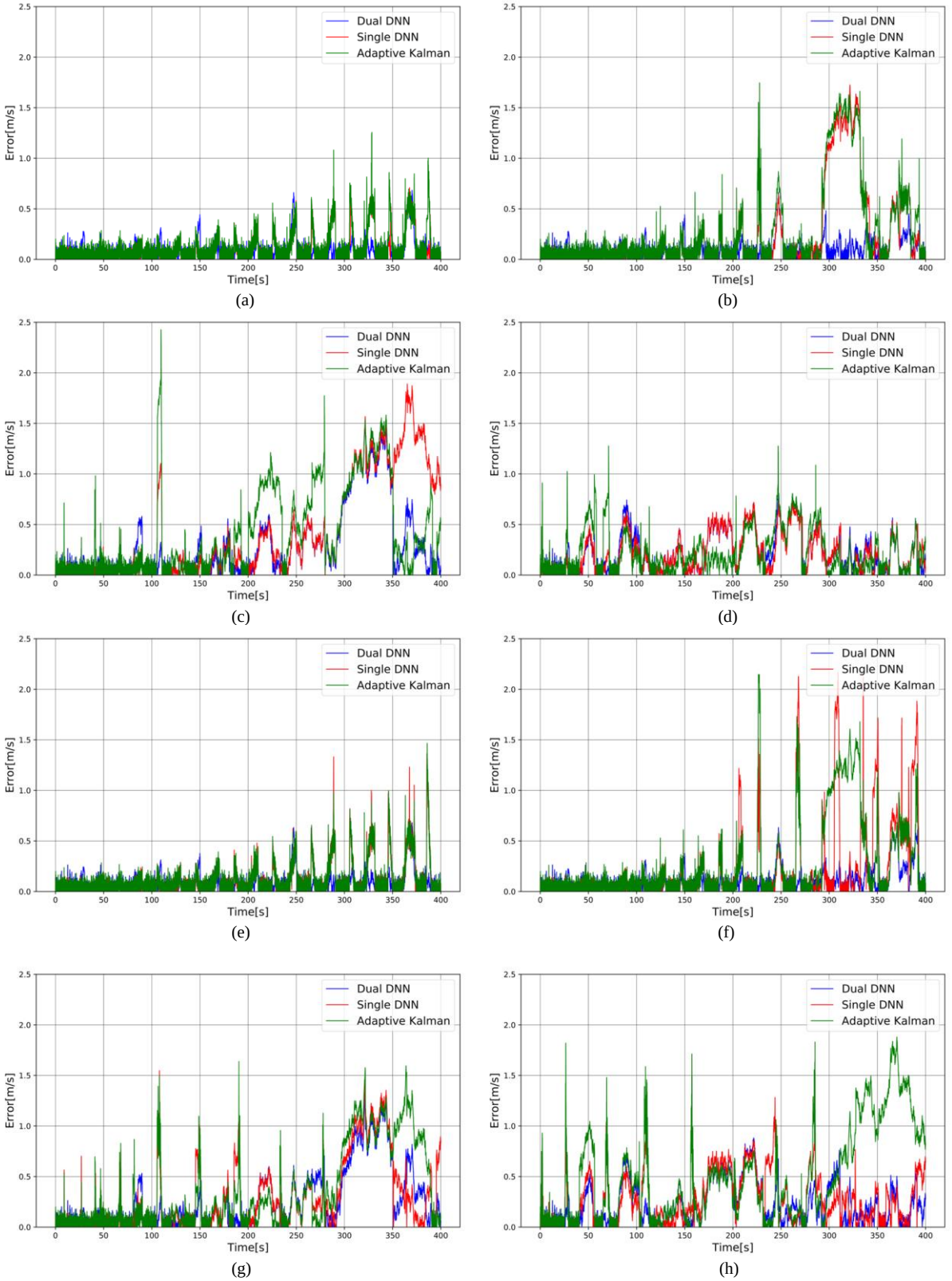


FIGURE 10. **Speed estimation errors of the three compared methods for the eight cases in Table II. (a) error on D-F, (b) error on W-F, (c) error on S-F, (d) error on I-F, (e) error on D-N, (f) error on W-N, (g) error on S-N and (h) error on I-N cases**

REFERENCES

[1] L. R. Ray, "Nonlinear estimation of vehicle state and tire forces," in *Am. Control Conf., IEEE*, 1992, pp. 526–530.

[2] F. Pretagostini, L. Ferranti, G. Berardo, V. Ivanov, and B. Shyrokau, "Survey on wheel slip control design strategies, evaluation and application to antilock braking systems," *IEEE Access*, vol. 8, pp. 10951–10970, 2020.

[3] A. Satya Kalyan, T. Divakar, K. N. Rao, and A. Mani Chakravarthi, "Vehicle velocity prediction & estimation in 2d video for night condition," *Int. J. Process. Image Process. Pattern Recognit.*, vol. 4, 2011.

[4] M. Noumura, M. Hori, J. Shimomura, and M. Terashima, "Velocity measurement using phase orthogonal spatial filters," *IEEE Trans. Ind. Appl.*, vol. 32, no. 4, pp. 796–801, July 1996.

[5] E. Klinefelter and J. A. Nanzer, "Interferometric microwave radar with a feedforward neural network for vehicle speed-over-ground estimation," *IEEE Microw. Wirel. Compon. Lett.*, vol. 30, no. 3, pp. 304–307, March 2020.

[6] L. Y. Hsu and T. L. Chen, "Vehicle full-state estimation and prediction system using state observers," *IEEE Trans. Veh. Technol.*, vol. 58, no. 6, pp. 2651–2662, July 2009.

[7] N.K. M'sirdi, A. Rabhi, L. Fridman, J. Davila, and Y. Delanne, "Second order sliding mode observer for estimation of velocities, wheel sleep, radius and stiffness," *Am. Control Conf.*, 2006, pp. 6.

[8] B. Moaveni, M. K. R. Abad, and S. Nasiri, "Vehicle longitudinal velocity estimation during the braking process using unknown input Kalman filter," *Veh. Syst. Dyn.*, vol. 53, no. 10, pp. 1373–1392, Jul. 2015

[9] H. Guo, H. Chen, F. Xu, F. Wang, and G. Lu, "Implementation of EKF for vehicle velocities estimation on FPGA," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 3823–3835, Sept. 2013.

[10] F. Jiang and Z. Gao, "An adaptive nonlinear filter approach to the vehicle velocity estimation for ABS," *IEEE Int. Conf. Control Appl.*, Sept. 2000, pp. 490–495.

[11] Y. Zhang, B. Leng, L. Xiong, Z. Yu, and D. Zeng, "Distributed drive electric vehicle longitudinal velocity estimation with adaptive kalman filter: Theory and Experiment," *SAE Technical Paper*, Apr. 2019.

[12] K. Kobayashi, K. Cheok, and K. Watanabe, "Estimation of absolute vehicle speed using fuzzy logic rule-based Kalman filter," *Proc. 1995 Am. Control Conf. ACC'95, IEEE*, vol. 5, June. 1995, pp. 3086–3090.

[13] A. Daiss and U. Kiencke, "Estimation of vehicle speed fuzzy-estimation in comparison with Kalman-filtering," *Proc. Int. Conf. Control Appl., IEEE*, 1995, pp. 281–284.

[14] X. Du, H. Sun, Y. Li, and L. Lu, "A prediction model for vehicle sideslip angle based on neural network," *IEEE Int. Conf. Inform. Financ. Eng.*, Sept. 2010, pp. 451–455.

[15] S. Torabi, M. Wahde, and P. Hartono, "Road grade and vehicle mass estimation for heavy-duty vehicles using feedforward neural networks," *Int. Conf. Intell. Transport. Eng.*, 2019, pp. 316–321.

[16] J. Liu, Z. Wang and L. Zhang, "A time-delay neural network of sideslip angle estimation for in-wheel motor drive electric vehicles," *IEEE Vehicular Technology Conference*., July 2020, pp. 1-5.

[17] X. Yang and L. Chen, "Dynamic state estimation for the advanced brake system of electric vehicles by using deep recurrent neural networks," *IEEE Transactions on industrial electronics*., vol. 67, no. 11, Nov 2019.

[18] C. Li, Y. Wang, W. Z. Zhou, J. Wu, W. Jin and C. Yin, "Lateral state estimation of preceding target vehicle based on multiply neural network ensemble," *IEEE Intelligent Vehicles Symposium*, June 2019, pp. 640-646.

[19] P. M. Sieberg, S. Blume, N. Harnack, N. Mass, and D. Schramm, "Hybrid state estimation combining artificial neural network and physical model," *IEEE ITSC*, Oct. 2019, pp. 894–899.

[20] T. Gräber, S. Lupberger, M. Unterreiner, and D. Schramm, "A hybrid approach to side-slip angle estimation with recurrent neural networks and kinematic vehicle models," *IEEE Trans. Intell. Veh.*, vol. 4, no. 1, pp. 39–47, March. 2019.

[21] M. De Martino, F. Farroni, N. Pasquino, A. Sakhnevych, and F. Timpone, "Real time estimation of the vehicle sideslip angle through regression based on principal component analysis and neural networks," *IEEE Int. Sys. Eng. Symp.*, 2017, pp. 1–6.

[22] A. L'heureux, K. Grolinger, H. F. Elyamany, and M. A. Capretz, "Machine learning with big data: Challenges and approaches," *IEEE Access*, vol. 5, pp. 7776–7797, April. 2017.

[23] K. H. Oh and C. K. Song, "Absolute vehicle speed estimation using neural network model," *J. Korean Soc. Precis. Eng.*, vol. 19, no. 9, pp. 51–58, Sept. 2002.

[24] C. C. Aggarwal, "Training deep neural networks," in *Neural networks and deep learning*, Springer, 2018, pp. 105–167.

[25] R. Rajamani, "Tire-road friction measurement on highway vehicles," in *Veh. Dyn. Control*, Springer Science & Business Media, 2011, pp. 397–424.

[26] H. Mirzaeinejad and M. Mirzaei, "A novel method for non-linear control of wheel slip in anti-lock braking systems," *Control. Eng. Pract.*, vol. 18, pp. 918–926, Aug. 2010.

JONG HAN KIM received the B.S. degree in automotive engineering from Hanyang University, Seoul, South Korea, in 2019. He is currently pursuing the master's degree in automotive engineering with Hanyang University, Seoul, South Korea. His current research interest includes control and state estimation of vehicles and artificial intelligence.

SANG WON YOON (Senior Member, IEEE) received the B.S degree in electrical engineering from Seoul National University, Seoul, South Korea, in 2000, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the University of Michigan, Ann Arbor, MI, USA, in 2003 and 2009, respectively. From 2009 to 2013, he was a Senior Scientist and a Staff Researcher with the Toyota Research Institute of North America, Ann Arbor, where he conducted research in the fields of power electronics and sensor systems for automobiles. Since 2013, he has been with the Department of Automotive Engineering, Hanyang University, Seoul, where he is currently an Associate Professor. His current research interests include power electronics, sensors and sensor systems, electronic reliability, and their applications in conventional and future vehicles.