applied sciences

MDPI

Article

Driving Behavior Classification and Sharing System Using CNN-LSTM Approaches and V2X Communication

Seong Kyung Kwon, Ji Hwan Seo, Jun Young Yun and Kyoung-Dae Kim *

Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu 42988, Korea;
sk_kwon@dgist.ac.kr (S.K.K.); seojh410@dgist.ac.kr (J.H.S.); jyyun1209@dgist.ac.kr (J.Y.Y.)
* Correspondence: kkim@dgist.ac.kr

Abstract: Despite advances in autonomous driving technology, traffic accidents remain a problem to be solved in the transportation system. More than half of traffic accidents are due to unsafe driving. In addition, aggressive driving behavior can lead to traffic jams. To reduce this, we propose a 4-layer CNN-2 stack LSTM-based driving behavior classification and V2X sharing system that uses time-series data as an input to reflect temporal changes. The proposed system classifies driving behavior into defensive, normal, and aggressive driving using only the 3-axis acceleration of the driving vehicle and shares it with the surroundings. We collect a training dataset by composing a road that reflects various environmental factors using a driving simulator that mimics a real vehicle and IPG CarMaker, an autonomous driving simulation. Additionally, driving behavior datasets are collected by driving real-world DGIST campus to augment training data. The proposed network has the best performance compared to the state-of-the-art CNN, LSTM, and CNN-LSTM. Finally, our system shares the driving behavior classified by 4-layer CNN-2 stacked LSTM with surrounding vehicles through V2X communication. The proposed system has been validated in ACC simulations and real environments. For real world testing, we configure NVIDIA Jetson TX2, IMU, GPS, and V2X devices as one module. We performed the experiments of the driving behavior classification and V2X transmission and reception in a real world by using the prototype module. As a result of the experiment, the driving behavior classification performance was confirmed to be ~98% or more in the simulation test and 97% or more in the real-world test. In addition, the V2X communication delay through the prototype was confirmed to be an average of 4.8 ms. The proposed system can contribute to improving the safety of the transportation system by sharing the driving behaviors of each vehicle.

Keywords: CNN-LSTM; driving behavior classification; driving safety

check for updates

Citation: Kwon, S.K.; Seo, J.H.; Yun, J.Y.; Kim, K.-D. Driving Behavior Classification and Sharing System Using CNN-LSTM Approaches and V2X Communication. *Appl. Sci.* **2021**, *11*, 10420. https://doi.org/10.3390/app112110420

Academic Editors: Tae-Koo Kang, Dong-Sung Pae and Myo-Taeg Lim

Received: 10 September 2021
Accepted: 3 November 2021
Published: 5 November 2021

1. Introduction

In the past decade, remarkable advances have been realized in various fields such as object detection [1,2], tracking [3,4], control [5], and Vehicle-to-Everything (V2X) communication [6,7] to achieve the goal of autonomous driving. Object detection uses sensors such as cameras, lidar, and radar to detect objects that affect driving. Intelligent driving assistance systems are being proposed through object tracking and control algorithms based on detected objects. Driving applications including adaptive cruise control (ACC) that assists driving is a representative example [8,9]. V2X provides not only vehicle-to-vehicle, but also vehicle-to-infrastructure connectivity, providing safety applications such as construction section warnings, stop sign violations and curve speed warnings [10]. These driving applications and safety applications have helped reduce the stress on many drivers [11,12].

Despite advances in autonomous driving technology, traffic accidents remain a problem to be solved in the transportation system [13,14]. According to the World Health Organization, statistics are reported that ~1.3 million people die in traffic accidents each year [15]. In the US alone, an average of 6 million traffic accidents occurs each year [16], and a study has reported that aggressive driving is more than 56% of the causes of traffic accidents [17]. Aggressive driving behavior can cause traffic accidents or adversely affect the driving of adjacent vehicles, resulting in traffic congestion [18,19]. Advances in communication, intelligent transportation systems and computer systems provide opportunities to provide traffic safety, convenience, and efficiency. Along with deep learning, the V2X system is an effective approach that can collect information from a variety of sources, broaden driver awareness and prevent potential accidents [20]. V2X communication is one of the key elements to increase both the safety and efficiency of driving vehicles by expanding the vehicle network, which aims to increase the safety and efficiency of the transportation system by sharing information between vehicles, pedestrians, and infrastructure.

We aim to improve traffic safety by predicting aggressive driving behavior using only low-cost sensor data and sharing it with the surrounding vehicles to reduce the cycle of adverse impacts. Therefore, we propose a 4-layer CNN-2 stacked LSTM-based approach for aggressive behavior classification and a V2X-based driving behavior sharing system. The proposed system reduces cost by using time series data of only 3-axis accelerations to classify aggressive behavior. In addition, we use the traffic simulator IPG CarMaker [21] to build a road that mimics the real-world environments, and generate a training dataset by repeatedly human driving vehicles with various driving behaviors. The training dataset is supplemented by adding real road driving data as well as simulators. The vehicle's driving behavior is shared with nearby vehicles [22], pedestrians or traffic light systems through V2X communication. We built a prototype module including an NVIDIA Jetson TX2 board and a V2X device. Vehicles equipped with the prototype module transmit driving behavior in real time to the surrounding vehicles via V2X communication. The proposed system has been verified in the simulation and a real vehicle, and can be configured at low cost with only an IMU, V2X device, and a mini PC. In addition, traffic safety can be improved by notifying the surrounding vehicles of aggressive driving behavior, and there is a possibility of applying various applications through V2X communication.

The structure of this paper is as follows. In Section 2, the conventional researches to detect driving behaviors are discussed. A deep learning-based driving behavior detection and V2X system are described in Section 3. The proposed system has been validated in Section 4, and the remaining challenges and conclusions are discussed in Section 5.

2. Related Works

Effective and reliable driving behavior analysis and early warning systems are required for the purpose of reducing traffic accidents and improving road safety [23]. For the classification of driving behavior, techniques using an in-vehicle camera [24,25], and sensors that measure the movement of the vehicle [26–28] have been proposed. A technique using an in-vehicle camera detects unsafe driving by observing the movement of the driver's eyes. Unsafe driving includes driver distraction, driver fatigue, and drowsy driving. The Inertial Measurement Units (IMU) sensor has been used in a number of studies to greatly improve the detection accuracy of driving events [10]. In conventional approaches, data from the IMU are used to detect the vehicle's driving behaviors through algorithms such as pattern matching, principal component analysis (PCA), and dynamic time warping (DTW). A MIROAD system that detects and recognizes driving events by fusion of the accelerometer, gyroscope, magnetometer, GPS, and the video sensor of a smartphone has been proposed [27]. The MIROAD system uses a rear camera, accelerometer, gyroscope, and GPS to classify only aggressive and non-aggressive behaviors based on multi-sensor data fusion and DTW algorithms. Standard driving events were collected from normal city driving, and aggressive driving events were collected in a controlled environment for safety. Aggressive driving was set with a hard left and right turns, swerves, and sudden braking and acceleration patterns. MIROAD showed an aggressive event detection performance of ~97% by combining the smartphone's gyroscope value, accelerometer value, and Euler angle rotation value. Conventional approaches using camera sensors may have problems with recognition performance due to environmental changes such as weather and illuminance [29]. In addition, some of the previous studies do not reflect various road environments or have not been verified in actual vehicles [30]. Specifically, aggressive driving on a curve is a factor that causes a crash accident, so it is necessary to predict driving behavior on a curve [10]. Additionally, the conventional method of detecting abnormal driver behavior by capturing and analyzing the driver's face and vehicle dynamics through image and video processing cannot capture the complex temporal features of driving behavior [31].

Recently, a technique for predicting driving maneuver through input of vehicle speed, acceleration, and steering wheel angle using deep learning has been proposed [32]. As the driving behavior of a vehicle can change with time, it is advantageous to use time series information rather than instantaneous information. Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and the Gated Recurrent Units (GRU), which are deep learning networks that utilize time series data for driver behavior profiling, have been used. For the comparison of the aggressive behavior classification performance of RNN, LSTM, and GRU, an evaluation based on data collected from a smartphone accelerometer has been proposed [33]. The proposed evaluation method constructed a dataset with 69 samples by driving a car on four different routes, and then classified seven types of driving events through RNN, LSTM and GRU. GRU and LSTM showed a classification accuracy of ~95% or better, whereas RNN was evaluated with a classification accuracy of ~70%. A driver behavior classification approach through CNN alone has also been proposed [34]. Information from the smartphone's accelerometer, gyroscope, and Engine Control Unit (ECU) was collected to distinguish driver behavior. The ECU information used is the vehicle's RPM, speed, and throttle, and is collected from an On-Board Diagnostic (OBD II) adapter. After that, a time-window applies to the nine signals collected to transform the temporal dependence of the signals into spatial dependence, and an image is generated through the recurrence plot technique. The generated image is trained by CNN, and has its own performance evaluation result of up to 99.99%. Despite the high classification performance of the introduced method, it requires a prior work to convert the signal into an image. The deep learning method using the accelerometer of a smartphone is being applied to Human Activity Recognition (HAR) in addition to driver behavior classification [35,36]. In the state-of-the-art approaches, a combined network of CNN-LSTM is applied rather than one kind of network of CNN or LSTM. The CNN-LSTM network extracts the features of the input signal through the CNN and uses it as the input of the LSTM. The combination of CNN and LSTM improves recognition performance by utilizing both spatial and temporal information. A 4-layer CNN-LSTM network has been proposed as the latest approach to improve the recognition performance in CNN-LSTM networks [35]. The 4-layer CNN-LSTM is a CNN-LSTM network with four one-dimensional convolutional layers and a maxpooling layer to summarize the feature map and reduce the computational cost. The 4-layer CNN-LSTM network showed about 99.39% of classification performance in HAR.

To classify aggressive driving behaviors using deep learning, a dataset for training a neural network is required. However, the actual road environment is very wide and the behavior of drivers is very diverse, so it is difficult to construct the dataset. For the generation of datasets for neural network training, simulators are an efficient alternative. IPG's TruckMaker-based driver model was used to train LSTM to classify the truck driver model [37]. The proposed technique categorizes the truck driver model into six types based on longitudinal and lateral acceleration, engine speed, vehicle speed, throttle, and pitch angle. TruckMaker was used to create a vehicle model with five different trailer loads of 0, 5, 10, 15, and 20 tones for the LSTM's training data generation. As a result, it classifies drivers by approximately 82% and 74% classification accuracy on the training and test set. However, the training model of this method has not been validated in a real environment. In addition, the proposed method only performs the classification of the driver model and does not detect aggressive driving behaviors or driving events.

The proposed system combines a classifier that classifies the driving behavior of individual vehicles as $defensive$, $normal$, or $aggressive$, and a V2X system that shares driving behavior. The driving behavior classifier is extended based on a 4-layer CNN-LSTM network with high behavior classification performance without requiring pre-processing of input data such as signal to image converting. As the input of the proposed network uses only 3-axis acceleration representing vehicle movement, the system cost is low and it is robust in the environment changes. In addition, we present the comparison results of the classification performance of the proposed network and the networks presented in other literature. We include not only driving behavior classification, but also behavior classification results in Basic Safety Message (BSM) to form a prototype of a system that shares with surrounding vehicles through V2X communication. Finally, we contribute to improving driving safety by providing an example of an advanced ACC system through sharing driving behavior.

3. Driving Behavior Classification and Sharing System

Figure 1 shows the structural diagram of the proposed system. The proposed system can measure the 3-axis acceleration of each vehicle in 10 ms cycle through IMU. Four-layer CNN-2 stacked LSTM is performed to predict the behavior of the vehicle using measured information from time series data as input. The vehicle's behavior is predicted to be defensive, normal or aggressive behavior. After that, the predicted behavior of the driving vehicle is inserted into the BSM. The BSM containing the driving vehicle's behavior information is transmitted to nearby vehicles through the V2X module. Nearby vehicles can drive more safely by sharing the driving behavior of other vehicles.

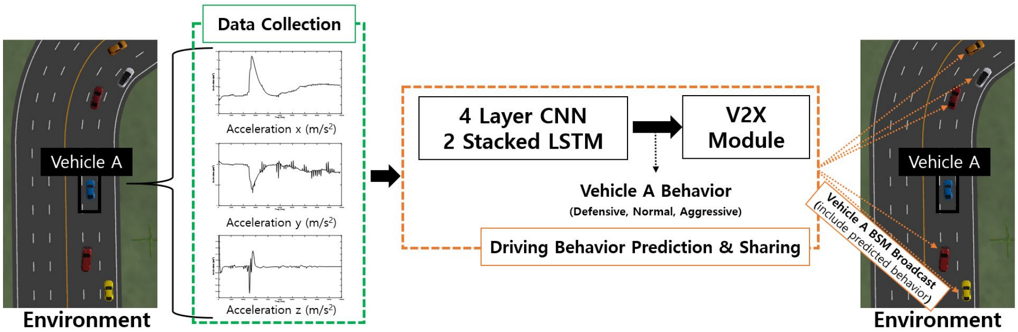


Figure 1. The proposed 4-layer CNN-2 stacked LSTM-V2X system for driving behavior classification and transmission.

3.1. LSTM Cell Structure

LSTM is composed of an input gate, an output gate, and a forget gate that combines short-term memory and long-term memory, solving the vanishing gradient problem of general RNNs [38]. Figure 2 shows the general structure of LSTM. LSTM has four interactive modules, and learning is performed through the following equations:

$$f_t = \sigma(W_f \cdot\ x_t + U_f \cdot\ h_{t-1} + b_f) \tag{1}$$

$$i_t = \sigma(W_i \cdot\ x_t + U_i \cdot\ h_{t-1} + b_i) \tag{2}$$

$$\bar{C}_t = \tanh(W_c \cdot\ x_t + U_c \cdot\ h_{t-1} + b_c) \tag{3}$$

$$C_t = f_t \otimes\ C_{t-1} + i_t \otimes\ \bar{C}_t \tag{4}$$

$$o_t = \sigma(W_o \cdot\ x_t + U_o \cdot\ h_{t-1} + b_o) \tag{5}$$

$$h_t = o_t \otimes\ \tanh\left(C_t\right) \tag{6}$$

where $\otimes$ means Hadamard product; $W_f, W_i, W_o, W_c, U_f, U_i, U_o, U_c$ are the weight parameters; and b_f, b_i, b_o, b_c are the bias vectors of corresponding operations. Here, the values of the weight and bias vectors are determined through the network training process. C_t is called cell state or long-term memory. h_t is called hidden state or short-term memory. The LSTM input x_t is concatenated with h_{t-1}, the hidden state of the previous time step. The forget gate f_t of (1) is a gate for forgetting past information. The value sent by the forget gate is the value that takes h_{t-1} and x_t and takes the sigmoid σ. Here, the output range of the sigmoid function is between 0 and 1, and as it gets closer to 0, the value of the previous state is not used. On the other hand, if the output of the sigmoid function is close to 1, it maintains the value of the previous state, affecting the prediction of the current state. The input gate $i_t \otimes \bar{C}_t$ is a gate for memorizing the current information. i_t and $\bar{C}_t$ are expressed in (2) and (3), respectively. i_t and $\bar{C}_t$ affect the update of cell state and hidden state through the Hadamard product. If the output value of the Forget gate f_t is 0, the Hadamard product result with the previous cell state C_{t-1} is also 0. In this case, only the result of the input gate is used to determine the current cell state C_t in (4). o_t in (5) is the value of the input x_t at the current time and the hidden state at the previous time h_{t-1} passed the sigmoid function. In the output gate, C_t passes through the hyperbolic tangent and is determined as a value between -1 and 1, and the Hadamard product with o_t is performed. Finally, h_t in (6) is determined the hidden state at the current time and is used the output of the LSTM cell at the current time.

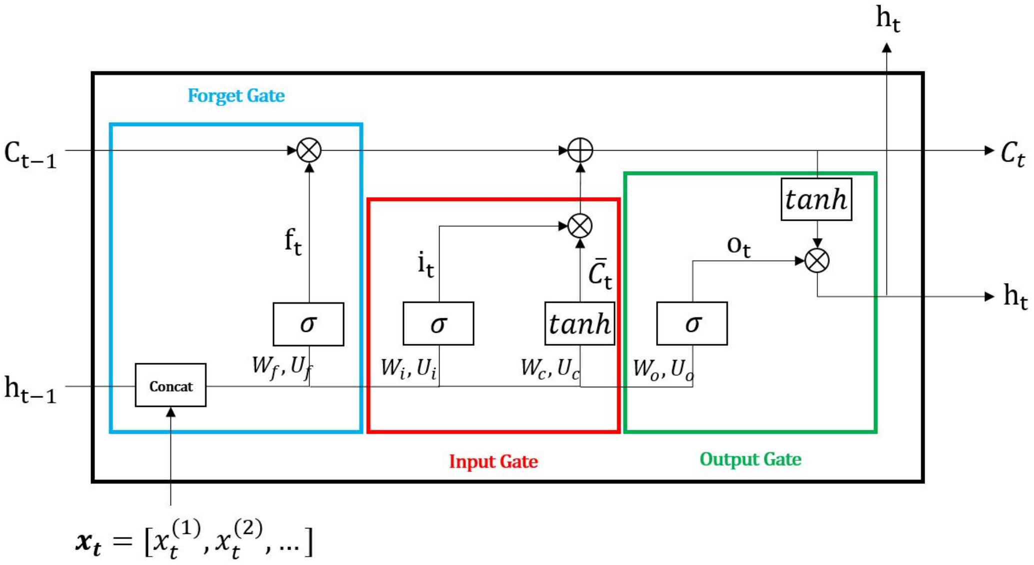


Figure 2. LSTM structure.

3.2. Base Network for Driving Behavior Classification

Deep learning-based techniques are being studied to address various classification problems. Here, we describe state-of-the-art behavioral classification networks including CNN, LSTM, and CNN-LSTM models. Through each network analysis, the base network of our proposed system is selected and extended to improve performance. Figure 3 is a 1D-CNN network architecture. 1D-CNN is mainly used for classification using timeseries data with LSTM. Representative applications using 1D-CNN include real-time heart monitoring [39] and abnormality detection for EEG data [40], or classification of movements such as running and walking from an accelerometer data in HAR researches [41]. 1D-CNN and 2D-CNN consist of a convolutional layer, a pooling layer, an activation function, and a fully connected layer. 1D-CNN consists of two 1D convolutional layers and a fully connected layer. In addition, overfitting was prevented and computation was reduced through two maxpooling layers and one dropout. The final result is output as one of defensive, normal, and aggressive behaviors through softmax. 2D-CNN is generally applied to various classification techniques based on image input. Recently, a 2D-CNN technique for detecting driver behavior has been proposed [34]. The 2D-CNN network [34] uses a total of 9 signal data: 3-axis acceleration, 3-axis gyroscope, RPM, speed, and throttle information.

The 9 signal data are converted into 3 images with grayscale size of 150 × 50 using the recurrence plot function of python. Each of the three images corresponds to an RGB image channel, and the final network input is an RGB image of size 3 × 150 × 50. However, as we only use 3-axis acceleration information, we create 50 × 50 images with x-, y-, and z-axis accelerations as red, green, and blue channels of RGB image, respectively. The 2D-CNN network consists of a total of three convolutional layers and two fully connected layers, as shown in the Figure 4. In addition, the computation was reduced and overfitting was prevented using two dropouts and maxpooling, and one global maxpooling.

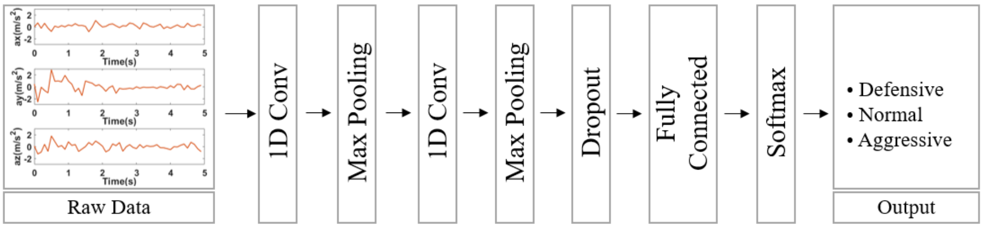


Figure 3. 1-Dimensional Convolution Neural Network (1D-CNN) structure [39].

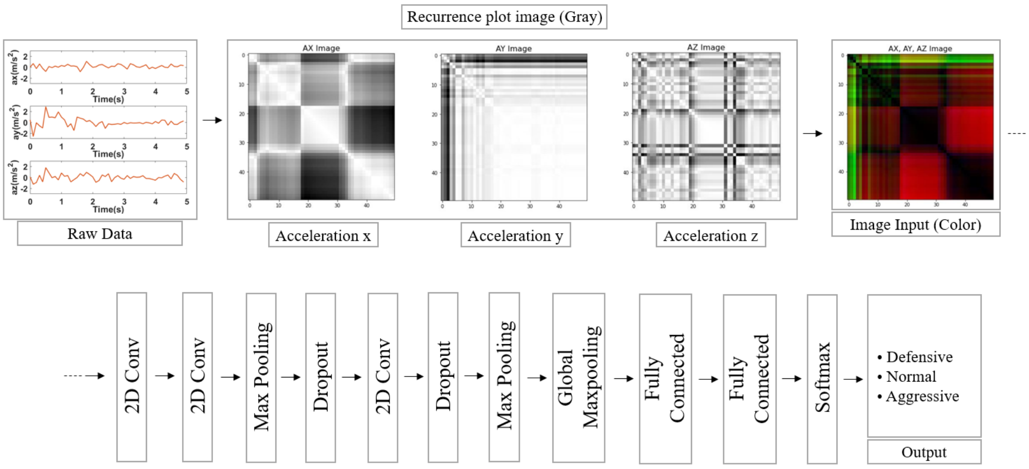


Figure 4. 2-Dimensional Convolution Neural Network (2D-CNN) structure [34].

The configuration of the LSTM network is as shown in Figure 5. The LSTM network consists of one LSTM layer and one fully connected layer. Overfitting is prevented through a dropout layer, and driving behavior is finally classified through a softmax.

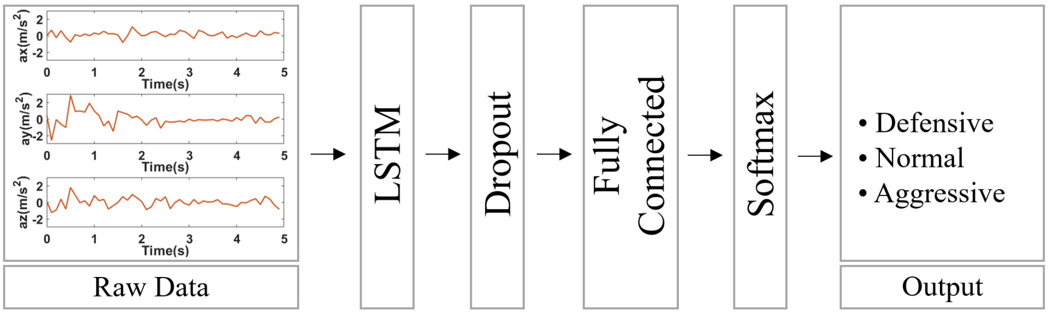


Figure 5. LSTM structure [37].

The CNN-LSTMs have been applied as an approach to predict text descriptions using image sequences. The CNN-LSTM architecture uses a CNN layer in the feature extraction process of the input data integrated with LSTM to support sequence prediction, as shown in Figure 6. The structure of the network consists of two convolutional layers: one LSTM layer and one fully connected layer. In addition, overfitting is prevented with one maxpooling and two dropouts.

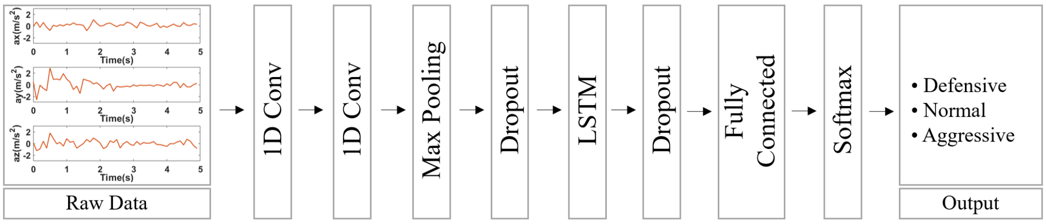


Figure 6. CNN-LSTM structure [35].

A 4-layer CNN-LSTM network, a state-of-the-art method for classifying human activity, is presented in Figure 7 [35]. The 4-layer CNN-LSTM was applied to the HAR research area by inputting the smartphone's 3-axis acceleration and 3-axis gyroscope information. The basic structure of the 4-layer CNN-LSTM is similar to the CNN-LSTM architecture, and there are four convolutional layers before the LSTM layer. Then, a max-pooling layer is added to summarize the feature map and reduce computational cost. Furthermore, two dropouts are inserted to prevent overfitting. Unlike 2D-CNN, which converts signal data into an image and uses it as an input, 4-layer CNN-LSTM uses signal data directly as an input to the network. In addition, the structure combining 1D-CNN and LSTM has strengths in feature extraction of input data and time series prediction performance, so it is selected as a base model for behavior classification.

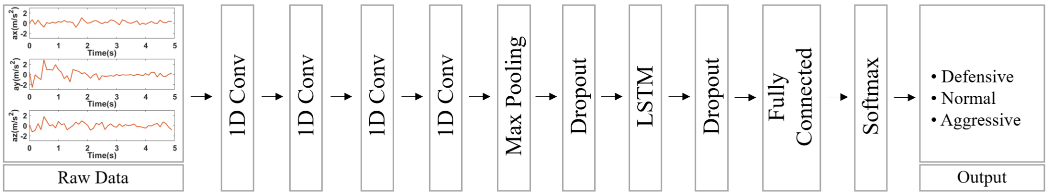


Figure 7. 4-Layer CNN-LSTM structure [35].

3.3. Four-Layer CNN-2 Stacked LSTM for Driving Behavior Classification

We propose a 4-layer CNN-2 stacked LSTM with a stacked LSTM applied with CNN to improve driving behavior classification performance. The proposed network is extended based on a 4-layer CNN-stacked LSTM. We created a deeper network by adding LSTMs to improve driving behavior classification performance. A stacked LSTM structure can be defined as an LSTM model composed of multiple LSTM layers to utilize the temporal feature extraction obtained from each LSTM layer [35]. In addition, stacked LSTM has better prediction performance than Vanilla-LSTM [42,43]. Figure 8 shows the structure of the proposed 4 layer CNN-2 stacked LSTM network. For the input data, feature extraction and the summary are performed by repeating two convolutional layers and one maxpooling layer 2 times. A Maxpooling layer is added in the CNN layer to mitigate the increased risk of overfitting and the amount of computation by the configuration of stacked LSTM. As with other networks, two dropouts are inserted to prevent overfitting. The features output from the stacked LSTM are returned as a classification result for driving behavior through a fully connected layer and a softmax layer.

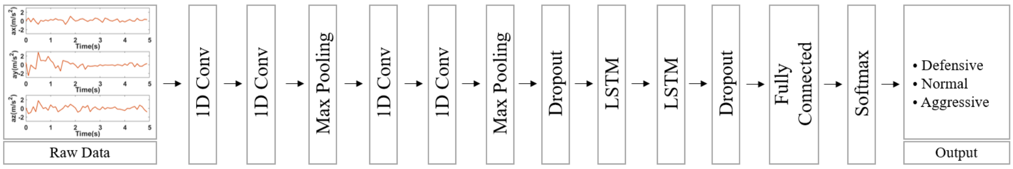
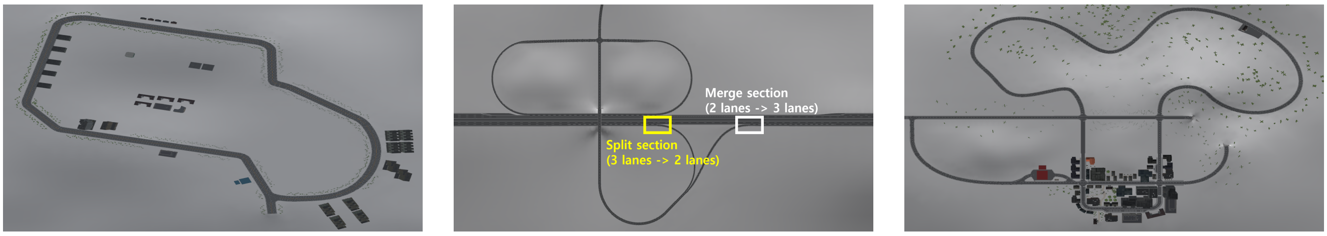


Figure 8. Four-layer CNN-2 stacked LSTM structure.

3.4. Dataset Generation

A training dataset is needed to train a network that classifies driving behavior by inputting only 3-axis acceleration. Time series data are selected to explore temporal relationships in the input data. Furthermore, the training dataset needs labels that correspond to the time series inputs. Unfortunately, collecting data from only real vehicles with various driving behaviors to construct a dataset is difficult and risky. Therefore, we safely collect various driving data using the autonomous driving simulator IPG CarMaker together with driver data in real road environments to build a training dataset in various environments.

Figure 9 is an example of CarMaker simulation for data set creation. Figure 9a is a map generated for data collection. This map was created to mimic the real road, DGIST campus. However, the actual driving lanes on the DGIST campus are one-lane or two-lane. To collect a lot of traffic and various behavioral data, the lanes were extended to three lanes in the simulation map. Furthermore, roundabouts were not created smoothly in the simulation, so we replaced the roundabouts with straight lanes. The driving distance of the vehicle in the campus driving simulation is 2.5 km, and it consists of straight lines and curves. Because roundabout is not taken into account, the driver's behavior may differ from the actual driving behavior on the DGIST campus. To supplement this, highway and urban scenarios were added to collect various driver behaviors in straight and curve. Figure 9b is the highway driving environment. The highway driving section is about 5.5 km straight and includes merging and splitting sections. The highway consists of a three-lane road, reducing to two lanes at 2.79 km. After this, at 3.02 km, it expands back to three-lane. The composition of the urban road environment is shown in Figure 9c. All roads in the urban environment consist of one lane, and there are straight lines and curves including uphill and downhill. The total distance of urban road using dataset is approximately 4.15 km. Table 1 shows the configuration and speed limit of the simulated road for dataset generation. The speed limit in the campus road is set at 30 km/h. The speed limit in highway conditions is 100 km/h. Urban roads have various speed limits set according to the driving distance.



(**a**) (**b**) (**c**)

Figure 9. CarMaker simulation roads for dataset generation: (**a**) DGIST Campus environment. (**b**) Highway environment. (**c**) Urban environment.

To collect the driving behavior dataset through simulation, we built a simulation driving environment as shown in Figure 10a. To mimic the real driving environment, we utilized a driving simulator. The compact driving simulator is composed of real vehicle parts, and the steering reaction force that feels the same as the real one is reproduced. We connected the driving simulator and CarMaker to perform simulation driving as shown in the Figure 10b. Each simulated road environment has one driver-driven vehicle and more than 200 traffic vehicles that mimic real drivers. In this environment, the simulated driving was repeated 3 times for a total of 3 driving behaviors by 3 drivers along each road environment to generate 27 driving profiles, which were used as datasets. At each iteration, the location of the lane change event of all traffic vehicles except the control vehicle, the number of lane changes, and the initial target speed is randomly reassigned. Each traffic vehicle uses Adaptive Cruise Control (ACC) to avoid collisions with other vehicles. Thus, interaction with surrounding vehicles occurs. For example, in Figure 10c, *Vehicle B* decelerates rapidly under the influence of *Vehicle A* making a lane change. The target speed of the traffic vehicle changes to a random value in the range of 10% of the speed limit every 1.5 s. In addition, each driving vehicle performed a lane change event in a random section, creating a complex environment similar to real roads.

Table 1. Predefined driving environment variables in CarMaker.

	Environment Variable			
Types	**Total Road Length**	**Road Section**	**Limit Speed**	**# of Lanes**
DGIST Campus	2256.3 m	0∼2256.3 m	30.0 km/h	1
Highway	5509.0 m	0∼2790 m	100.0 km/h	3
		2790∼3020 m		2
		3020∼5509 m		3
Urban	4154.5m	0∼280 m	30.0 km/h	1
		280∼315 m	50.0 km/h	
		315∼1470 m	70.0 km/h	
		1470∼1530 m	50.0 km/h	
		1530∼1855 m	80.0 km/h	
		1855∼1930 m	70.0 km/h	
		1930∼2175 m	50.0 km/h	
		2175∼4154.5 m	70.0 km/h	

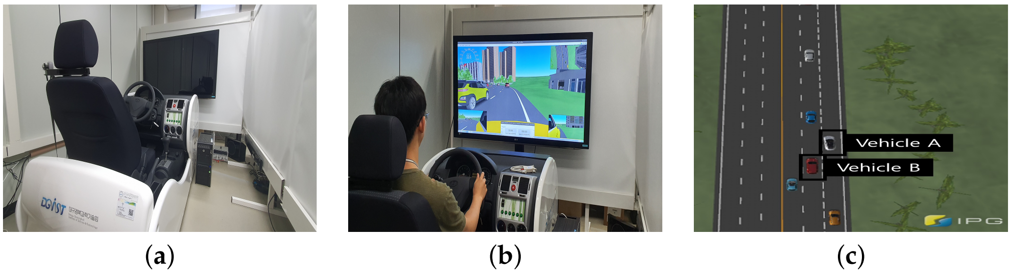


Figure 10. CarMaker simulation environments for dataset generation: (**a**) Driving environment. (**b**) Simulation driving. (**c**) Simulation scenario.

Driving datasets in real road environments were collected from real DGIST campus. Figure 11 is a vehicle used for real road dataset collection and algorithm testing. Real road driving data were collected from one driver repeating three driving with three driving behaviors under a speed limit of 30 km/h. Through the simulation and real-world driving, we obtained 3-axis acceleration information when passing a speed bump as well as going straight, curve, left turn, right turn, uphill, and downhill according to driving behaviors. In addition, real-world and simulation drivers themselves labeled driving aggression. As a driver's self-definition can be subjective, we followed the behavior of aggressive driving as defined by the AAA Foundation [44]. Among the aggressive driving behaviors, speeding in heavy traffic, cutting in front of another driver and then slowing down, and blocking cars attempting to pass or change lanes were performed. Defensive behavior was performed following the guidelines of the New York Department of Motor Vehicles (DMV) [45], such as maintain the correct speed, allow space. Normal behavior was determined to be between aggressive and defensive behavior. The driver performed sufficient practice driving for about 10 h or more according to the guidelines, and then data for each driving behavior were collected. All processes for dataset generation and labeling were performed under the supervision of a supervisor. The supervisor has a clear understanding of the guidelines for driving behavior and is a veteran driver with over 10 years of driving experience. If the driver selects the defensive behavior label but does not match the guidelines, such as exceeding the speed limit, data and labels are excluded. In addition, unrealistic driving data and intentional collision data with vehicles or pedestrians are also excluded. Table 2 represents the standard deviation and median for each collected driving behavior. The standard deviation of the x-axis acceleration represents an indicator for the acceleration and deceleration of the vehicle, and the larger the standard deviation, the larger the amount of change in acceleration and deceleration. The standard deviation of the y-axis acceleration represents an indicator of the vehicle's rotation, and the larger the standard deviation, the greater the amount of change in steering. As the z-axis acceleration is changed by the rolling and pitching of the vehicle, the standard deviation increases as the x-axis and y-axis accelerations change larger. Therefore, driving with aggressive behavior increases the standard deviation of the 3-axis acceleration. The median of the 3-axis acceleration for each driving behavior collected is close to zero, indicating that the collected data are not biased.

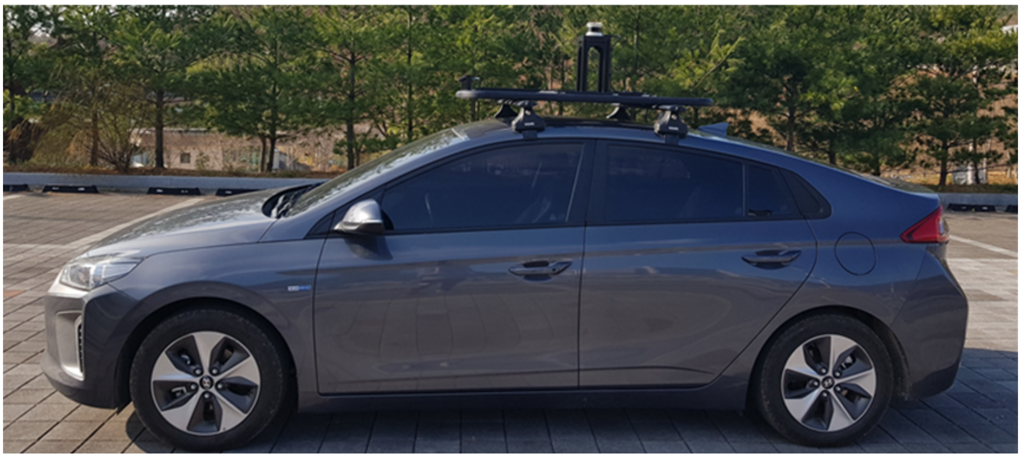


Figure 11. Real-world vehicle for data collection and testing.

Table 2. Three-axis acceleration standard deviation and median according to driving behavior.

		Defensive Behavior (m/s^2)	
	Acceleration X	Acceleration Y	Acceleration Z
Median	0.0209606	0.0041105	−0.0003024
Standard Deviation	0.439897633	0.517115998	0.263576081
		Normal Behavior (m/s^2)	
	Acceleration X	Acceleration Y	Acceleration Z
Median	0.0176545	0.0027567	−0.0005219
Standard Deviation	0.519467419	0.876498761	0.373084299
		Aggressive Behavior (m/s^2)	
	Acceleration X	Acceleration Y	Acceleration Z
Median	0.07252035	0.02170125	−0.00392755
Standard Deviation	0.915436498	1.462502723	0.491320123

We built a dataset from the collected 3-axis acceleration information and labels. The collected data are a total of 93,286 3-axis acceleration data and labels, consisting of 79,052 simulations and 14,234 real vehicle data. Defensive behavior is 32,650, normal behavior is 30,784, and aggressive behavior is 29,852, which are balanced in proportions of 35%, 33%, and 32% in the total dataset, respectively. We randomly extract 15% of the data from the entire dataset and use it as a validation set, and use the remaining 85% as a training dataset. The validation set is used to measure the performance of the model built on the training set. In general, various parameters and models are used to find out which model best fits the data, and among them, the model with the best performance is selected as the validation set. For the construction of the test set, we additionally collected data on three driving behaviors of one driver each on the simulated and real roads. The test set is used to measure the performance of the model after the model to be used as the validation set is decided. The total number of data used in the test is presented in Table 3, divided by scenario.

Table 3. Number of data assigned to labels in the test dataset.

Environments		Behavior (Ground Truth)			
		Defensive	Normal	Aggressive	Total Data
Simulation	DGIST	4440	3412	3292	11144
	Urban	3777	3002	1666	8445
	Highway	4257	3271	3124	10652
Real world	DGIST	2812	2248	1886	6946

3.5. Prediction Result Sharing via V2X

In order to improve the traffic safety, the driving behaviors classified through 4-layer CNN-2 stacked LSTM must be communicated to other traffic participants. As a method for sharing information between vehicles and transportation system, V2X communication is a useful means. Messages delivered via V2X include BSM, Emergency Vehicle Alert (EVA), Common Safety Request (CSR), and traveler information as stipulated by Society of Automotive Engineers (SAE). Among them, BSM is a message that provides the vehicle information for safety, and is composed of Part I and Part II. Part I contains the vehicle's GNSS position, speed, steering wheel angle, acceleration, brake system status and vehicle specifications, and is an essential component of BSM. On the other hand, Part II is divided into vehicle safety extension and vehicle status. The vehicle safety extension includes information on path history and path prediction, and vehicle status is optional. We include the predicted driving behavior via 4-layer CNN-2 stacked LSTM in vehicle status of BSM Part II, following the configuration of the BSM. The BSM with predicted driving behavior through the proposed system is broadcast to the around vehicles, which can assist nearby traffic participants in making safe driving decisions.

Figure 12 is a prototype of the 4-layer CNN-2 stacked LSTM-V2X system for predicting and sharing driving behavior. NVIDIA Jetson TX2 is used as a small computing board to drive the proposed 4-layer CNN-2 stacked LSTM-V2X system. The V2X device for sending and receiving BSM is Ettifos ETF-DV-02. In addition, the MTi-670 is mounted as an IMU for measurement of longitudinal and lateral acceleration. The prototype was equipped with an MRP-2000 GPS to collect the vehicle's driving trajectory and a battery to power each device. Jetson TX2 was selected because of its 4-layer CNN-2 stacked LSTM network operation and compatibility with the used V2X device. The total size of the prototype is 180 × 180 × 95mm, and consists of three layers. The first layer is the battery and V2X device, the second layer is the computing board, and the third layer is the IMU and GPS.

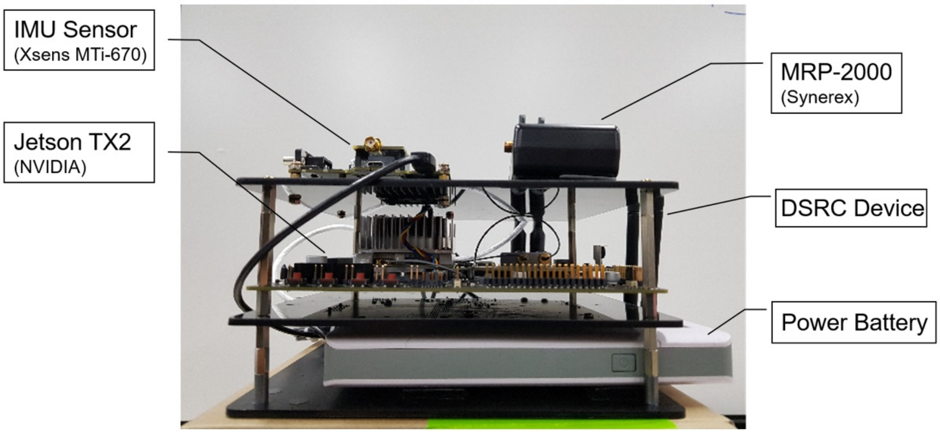


Figure 12. Four-layer CNN-2 stacked LSTM-V2X module.

4. Performance Evaluation

In this section, we evaluate the proposed driving behavior prediction and sharing system. The performance of the proposed system is evaluated by 4 layer CNN-2 stacked

LSTM's training results, simulations, and real-world test. In addition, the effectiveness of predicted driving behavior sharing on nearby vehicles is also evaluated in the simulation. In a real-world test, experimental results of V2X communication for sharing driving behavior between vehicles are presented.

4.1. Network Training and Comparison

To evaluate the performance of the proposed 4 layer CNN-2 stacked LSTM, we compare it with other conventional networks. In addition, we present performance indicators of networks according to the length of the input timeseries. For fair comparisons, all networks were trained for 50 epochs with a batch size of 32. Relu was used as the activation function and Adam optimizer was applied. As the loss function, categorical cross entropy was used. Network training was performed on NVIDIA Titan V graphics card, 64 GB RAM, and Intel i7-7700k CPU environment.

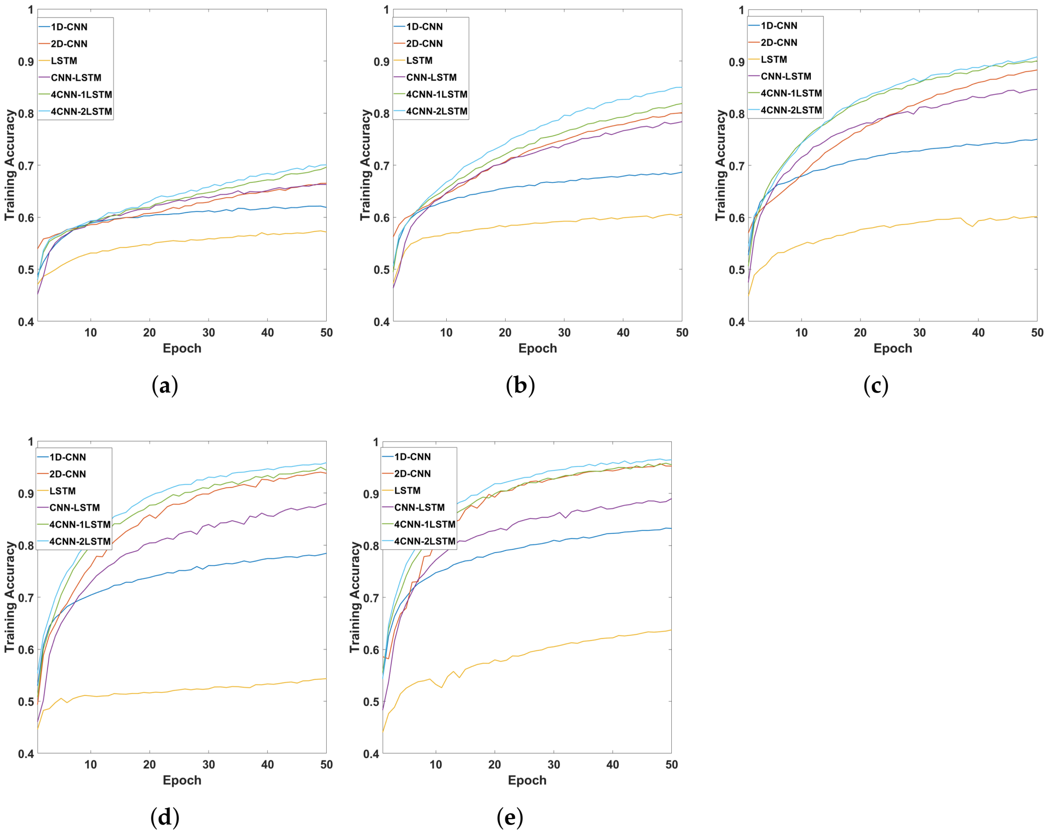
Figure 13 represents the training accuracy when data with different timeseries lengths are used as input. The input timeseries length was evaluated as 5 types ranging from 1 s to 5 s. As a result of the comparison according to the length of the input timeseries, all networks showed the best performance when given a timeseries of 5 s. The training accuracy of LSTM was ~63%,which was the worst among the compared networks. 1D-CNN had an average of 24% higher performance than LSTM with a training accuracy of ~83%. 2D-CNN and CNN-LSTM had similar performance at a time series length of 1 s. However, as the timeseries length increased, the performance of 2D-CNN was superior to that of CNN-LSTM. In all cases, the 4-layer CNN-LSTM showed the second highest training accuracy among networks. Our proposed 4-layer CNN-2 stacked LSTM has superior results compared to other networks in all cases. At 3 s of timeseries input, the performance was similar to that of the 4-layer CNN-LSTM, but in other cases, it showed an average performance improvement of 1.75%. In addition, it showed the highest training accuracy with about 96.5% at a timeseries length of 5 s. Table 4 shows the training time of each network model. In terms of training time, 1D-CNN with a simple structure is the fastest, and 2D-CNN is the slowest because it trains images. CNN-LSTM is trained about 4.3 times faster than LSTM through the process of extracting and summarizing features through CNN. The 4-layer CNN-2 stacked LSTM with the highest training accuracy trained the third fastest, and was trained ~18.3% faster than the 4-layer CNN-LSTM with the second highest training accuracy. Therefore, it has been verified that the proposed 4-layer CNN-2 stacked LSTM has high training accuracy while training faster compared to other networks.

Table 4. The networks training time at 50 epochs.

Networks	1D-CNN	2D-CNN	LSTM
Training Time (s)	17.82742023	476.4470894	163.4962823
Networks	CNN-LSTM	4CNN-LSTM	4CNN-2LSTM
Training Time (s)	37.72614694	92.75716472	75.77209926

4.2. Simulation Result of Driving Behavior Prediction

The performance of the trained networks is evaluated through the validation data and the simulation test data. We present the validation results at the input of 5 s, because the case of 1 to 4 s of input is significantly lower accuracy than the case of 5 s of input. Table 5 is the network performance indicator when the input timeseries is 5 s. The classification accuracy for the defensive behavior is about 92% or more in all networks, especially the CNN-LSTM network, the accuracy is improved by about 11% compared to when the input timeseries is 4 s. For normal behavior, the CNN network has a classification error of about 29% as defensive behavior. As a result of the aggressive behavior, 2D-CNN had a classification accuracy of about 78% when the input time series was 4 s, but showed a steep performance improvement to about 99% when the input timeseries was 5 s. The 4-layer CNN-2 stacked LSTM has an average accuracy of 98.52%, and the performance improvement is about 0.39% compared to the 98.13% of the 4-layer CNN-LSTM.



(**a**) (**b**) (**c**)

(**d**) (**e**)

Figure 13. Training accuracy input length: (**a**) Timeseries 1 s. (**b**) Timeseries 2 s. (**c**) Timeseries 3 s. (**d**) Timeseries 4 s. (**e**) Timeseries 5 s.

Table 5. Validation confusion matrix at 5 s timeseries input.

Validation		**Output Label**		
Input: 5 s	**Networks**	**Defensive**	**Normal**	**Aggressive**
Defensive	1D-CNN	92.87%	6.49%	0.63%
	2D-CNN	95.04%	0.29%	4.66%
	LSTM	93.02%	4.13%	2.86%
	CNN-LSTM	94.24%	5.76%	0.00%
	4CNN-LSTM	97.02%	2.91%	0.07%
	4CNN-2LSTM	99.52%	0.41%	0.07%
Normal	1D-CNN	29.4%	67.23%	3.37%
	2D-CNN	0.32%	91.56%	8.11%
	LSTM	62.23%	18.53%	19.24%
	CNN-LSTM	19.92%	79.92%	0.16%
	4CNN-LSTM	1.78%	97.92%	0.29%
	4CNN-2LSTM	3.29%	96.39%	0.32%
Aggressive	1D-CNN	3.39%	6.89%	89.72%
	2D-CNN	0.21%	0.07%	99.73%
	LSTM	15.83%	9.22%	74.95%
	CNN-LSTM	0.65%	4.08%	95.27%
	4CNN-LSTM	0.14%	0.41%	99.45%
	4CNN-2LSTM	0.16%	0.19%	99.65%

From the validation process, we select the input of timeseries 5 s that gives the best performance for classifying driving behavior. The trained networks are evaluated for performance in each environment through test data collected from DGIST campus, urban roads and highways in the simulation. Tables 6–8 are the results of network evaluation based on the test data generated in each simulation environment. The classification accuracy of proposed 4-layer CNN-2 stacked LSTM is more about 99% on highways that are simply composed of straight roads. In addition, on DGIST campuses and urban roads with various road types, the classification accuracy of the proposed network is relatively reduced compared to highways, but the classification accuracy is ~97% or higher on average. The network with the second highest classification performance is a 4-layer CNN-LSTM, which has a reduced performance of 1.3% in the DGIST campus scenario, 0.31% in the urban scenario, and 3.08% in the highway compared to our proposed network. The proposed 4-layer CNN-2 stacked LSTM takes a classification time of ~19.7 ms in the real-time evaluation performed on the TX2 board, and classifies driving behaviors about 1.6 ms faster than the classification time of 4-layer CNN-LSTM, about 21.3 ms. The next high performance networks are 2D-CNN networks and CNN-LSTM networks. 1D-CNN confuses normal and defensive behavior in all scenarios, and LSTM has the worst classification performance of normal behavior. As a result of performance verification based on validation data and test data, we have proven that the proposed 4-layer CNN-2 stacked LSTM classifies driving behavior as high performance.

Table 6. Confusion matrix of DGIST campus driving behavior classification results.

Test: DGIST Campus		**Output Label**		
Input: 5 s	**Networks**	**Defensive**	**Normal**	**Aggressive**
Defensive	1D-CNN	95.11%	3.68%	0.21%
	2D-CNN	91.91%	7.71%	0.39%
	LSTM	58.91%	2.02%	39.07%
	CNN-LSTM	94.84%	5.13%	0.04%
	4CNN-LSTM	97.45%	2.4%	0.14%
	4CNN-2LSTM	97.74%	2.12%	0.14%
Normal	1D-CNN	5.58%	90.83%	3.59%
	2D-CNN	4.32%	94.79%	0.89%
	LSTM	40.05%	2.35%	57.60%
	CNN-LSTM	1.95%	94.64%	3.41%
	4CNN-LSTM	2.53%	97.03%	0.44%
	4CNN-2LSTM	0.97%	97.83%	1.2%
Aggressive	1D-CNN	1.11%	5.91%	92.99%
	2D-CNN	1.5%	3.56%	94.95%
	LSTM	9.55%	2.53%	87.92%
	CNN-LSTM	0.42%	2.8%	96.78%
	4CNN-LSTM	0.79%	4.17%	95.04%
	4CNN-2LSTM	0.0%	2.22%	97.78%

Table 7. Confusion matrix of urban road driving behavior classification results in simulation.

Test: Urban road		**Output Label**		
Input: 5 s	**Networks**	**Defensive**	**Normal**	**Aggressive**
Defensive	1D-CNN	89.72%	9.8%	0.48%
	2D-CNN	92.64%	6.67%	0.69%
	LSTM	91.58%	1.72%	6.69%
	CNN-LSTM	81.7%	18.19%	1.17%
	4CNN-LSTM	94.62%	5.38%	0.0%
	4CNN-2LSTM	94.14%	5.82%	0.05%
Normal	1D-CNN	36.52%	60.04%	3.44%
	2D-CNN	3.44%	93.86%	2.69%
	LSTM	71.54%	6.29%	22.18%
	CNN-LSTM	10.87%	87.97%	1.17%
	4CNN-LSTM	2.63%	97.1%	0.27%
	4CNN-2LSTM	1.5%	97.82%	0.69%
Aggressive	1D-CNN	5.11%	9.88%	85.01%
	2D-CNN	1.71%	16.27%	82.01%
	LSTM	38.56%	2.84%	58.60%
	CNN-LSTM	0.31%	6.67%	93.02%
	4CNN-LSTM	0.31%	1.5%	98.19%
	4CNN-2LSTM	0.03%	1.12%	98.85%

Table 8. Confusion matrix of highway driving behavior classification results in simulation.

Test: Highway		**Output Label**		
Input: 5 s	**Networks**	**Defensive**	**Normal**	**Aggressive**
Defensive	1D-CNN	92.46%	7.02%	0.52%
	2D-CNN	97.3%	2.68%	0.02%
	LSTM	92.58%	4.56%	2.87%
	CNN-LSTM	93.59%	6.41%	0.0%
	4CNN-LSTM	95.02%	4.91%	0.07%
	4CNN-2LSTM	99.48%	0.45%	0.07%
Normal	1D-CNN	30.91%	65.45%	3.64%
	2D-CNN	2.45%	97.28%	0.28%
	LSTM	62.86%	19.35%	17.79%
	CNN-LSTM	21.28%	78.51%	0.21%
	4CNN-LSTM	4.4%	95.17%	0.43%
	4CNN-2LSTM	0.73%	99.21%	0.06%
Aggressive	1D-CNN	3.04%	6.95%	90.01%
	2D-CNN	0.42%	0.22%	99.36%
	LSTM	14.6%	8.8%	76.60%
	CNN-LSTM	0.51%	2.88%	96.61%
	4CNN-LSTM	0.13%	0.38%	99.49%
	4CNN-2LSTM	0.0%	0.06%	99.94%

4.3. Effectiveness of Prediction Result Sharing

The effectiveness of driving behavior sharing to surrounding vehicles was evaluated via simulation. The simulation environment consists of a straight road of ~5 km, with a ego vehicle and a target vehicle. A ego driving vehicle is a vehicle that uses ACC to follow speed while maintaining a safe distance from the target vehicle. A target vehicle is a vehicle whose driving behavior changes over time into defensive, normal, and aggressive behaviors. The proposed behavior sharing system is applied to both the ego vehicle and the target vehicle, and the target vehicle shares its own driving behavior through V2X communication, and the ego vehicle receives the target vehicle's driving behavior. The safe distance must be set so that the ego vehicle does not collide even if the target vehicle suddenly brakes [46]. Here, we add a variable called safety factor that serves to change the safe distance according to driving behavior. The safety factor can be set manually, and when the safety factor is 1, a safety distance equal to the default safety distance is assigned. As defensive behavior is the safest driving behavior, a default safety distance is assigned by setting the safety factor to 1. To provide an additional safety distance as the aggressiveness increases, we set the safety factor to 1.1 for normal behavior, and 1.3 for aggressive behavior. Therefore, if the target vehicle drives aggressively, the ego vehicle will have a desired distance of 30% more than the basic safety distance.

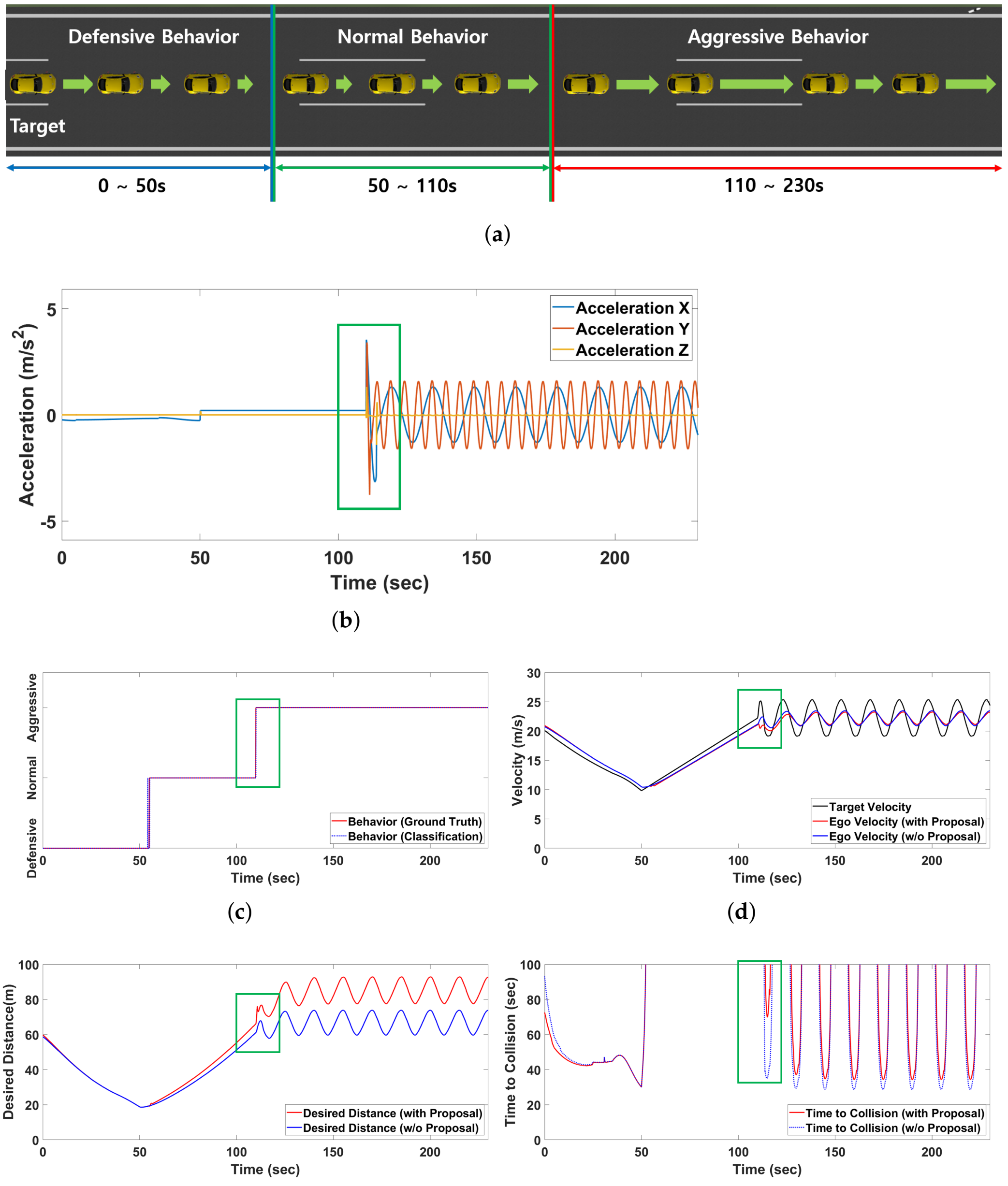
In the simulation, the target vehicle performs 50 s of defensive behavior, 60 s of normal behavior, and 120 s of aggressive behavior. To clearly show the effect of the target vehicle's aggressive behavior to the ego vehicle, the target vehicle was set to perform periodic acceleration and deceleration. Figure 14a represents the movement of the target vehicle in the simulation. The target vehicle decelerates at a maximum of 0.2658 m/s^2 from 0 to 50 s in defensive behavior, and accelerates at a constant rate of 0.2 m/s^2 in normal behavior from 50 to 110 s as shown in Figure 14b. Aggressive behavior is expressed an acceleration of 1.3 m/s^2 to -1.3 m/s^2 through a *sin* function at a 15-second cycle from 110 to 230 s. As shown in Figure 14c, the driving behavior of the target vehicle through the 4 layer CNN-2 stacked LSTM showed high-accuracy classification results.

Figure 14d shows the driving velocity of the target vehicle and the ego vehicle. The black line represents the velocity of the target vehicle, and the blue and red lines represent the velocity of the ego vehicle without V2X-based behavior sharing system and with V2X-based behavior sharing system, respectively. When the target vehicle has defensive and normal behavior, the velocity of the ego vehicle is similar regardless of the application of the proposed system. However, the desired distance of the ego vehicle increases with the target vehicle's behavior, as shown in Figure 14e. At about 110 s, the target vehicle accelerates, but the ego vehicle slows down as the desired distance is increased for safety, as shown in the green area in Figure 14e. The risk of collision with the target vehicle is reduced while inducing an increase in Time to collision (TTC) of approximately 3 s or more compared to the case where V2X based behavior sharing system was not applied as shown in the Figure 14f. As a result of the simulation experiment, we confirmed that the sharing of driving behavior of a vehicle can have a positive effect on traffic safety.

4.4. Real-World Test Results of Driving Behavior Prediction

The 4-layer CNN-2 stacked LSTM-V2X system was installed in two vehicles and verified in a real road environment. In a real-world test, human drivers select a label for their driving behavior as one of defensive, normal, and aggressive. Driving behavior is self-defined, and the guidelines of the AAA Foundation and DMV were followed in the same way as the dataset generation. The behavior classification accuracy was measured by comparing the driving behavior received through V2X with the driving behavior labeled by the driver. Figure 15 shows the path of the vehicle driven for real world test. The red line indicates the driving path of the vehicle and the arrow indicates the driving direction. As GPS information is not received in the tunnel section, it is not displayed on the path. Figure 16a–c shows the 3-axis acceleration information about driving behaviors labeled by the driver in the behavioral comparison section marked in yellow in Figure 15. The more aggressive the driver's driving behavior, the greater the change in 3-axis acceleration. In addition, the amount of change in the z-axis is large because the aggressive driving behavior does not reduce the speed sufficiently on the speed bump.

Table 9 shows the classification accuracy of each network based on real world driving data. Similar to simulation, the proposed 4-layer CNN-2 stacked LSTM has the best results with over 98% performance. In addition, although ~85% of the dataset used for network training consists of simulation data, it showed high classification accuracy in the real environment. The reason that the data set contains about 15% of real-world data is partly because the simulated road and the real-world environment does not match. For example, speed bumps are not configured on the simulated road to generate the data set. In addition, errors may occur depending on the condition of the road surface, tire and sensor noise in real world driving environment. From the experimental results, we demonstrate the possibility of high-performance classification of driving behaviors in various real world environments from a large number of simulated driving data for general road environments and some real world data.



(**a**)

(**b**)

(**c**) (**d**)

(**e**) (**f**)

Figure 14. Experimental results of safety improvement of the proposed system in the ACC scenario: (**a**) ACC simulation environment. (**b**) Three-axis acceleration of the target vehicle. (**c**) Behavior classification results. (**d**) Driving velocity. (**e**) Desired distance. (**f**) Time to collision.

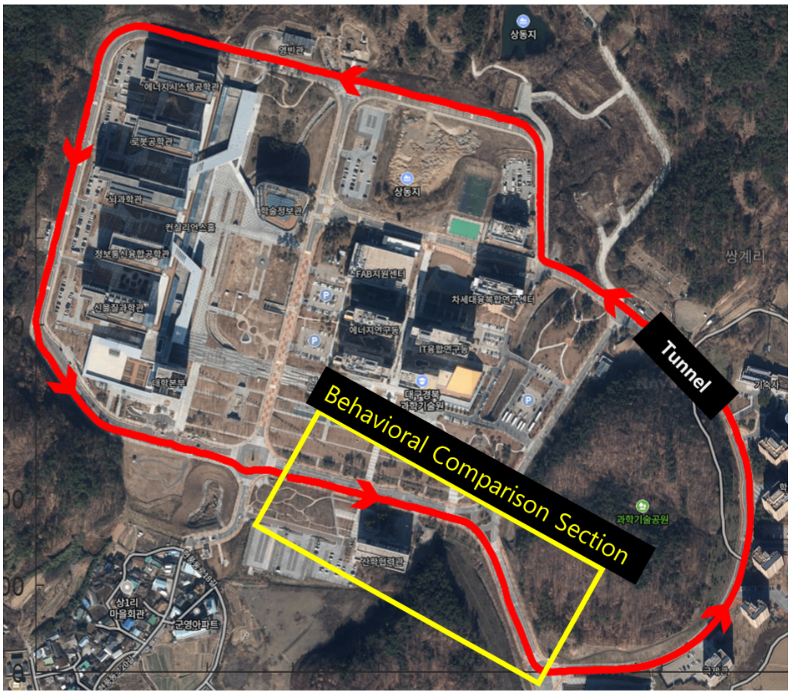
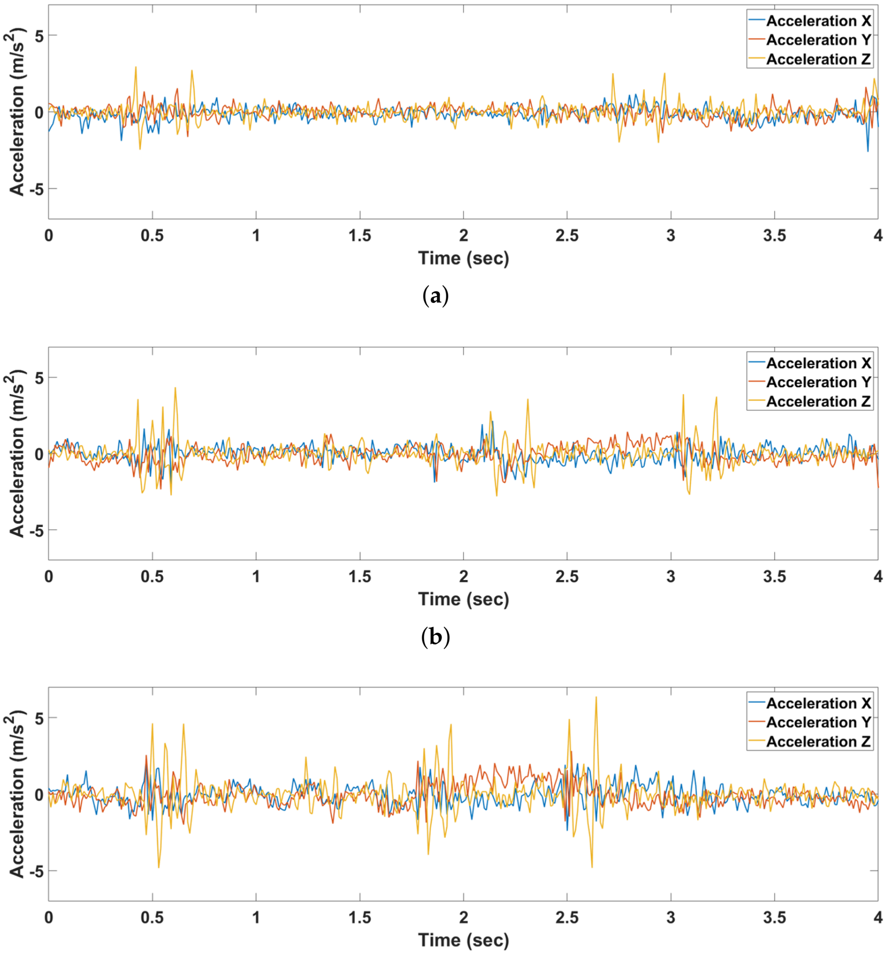


Figure 15. Driving path of a vehicle in real-world test.



(**a**)

(**b**)

(**c**)

Figure 16. Comparison of 3-axis accelerations according to driving behavior: (**a**) Defensive driving. (**b**) Normal driving. (**c**) Aggressive driving.

Table 9. Confusion matrix of driving behavior classification results in real world DGIST campus.

Test: Real World DGIST Campus		**Output Label**		
Input: 5 s	**Networks**	**Defensive**	**Normal**	**Aggressive**
Defensive	1D-CNN	97.51%	2.13%	0.36%
	2D-CNN	99.15%	0.64%	0.21%
	LSTM	90.58%	6.51%	2.92%
	CNN-LSTM	95.8%	4.2%	0.0%
	4CNN-LSTM	98.08%	1.53%	0.39%
	4CNN-2LSTM	98.65%	0.428%	0.002%
Normal	1D-CNN	2.49%	96.93%	0.58%
	2D-CNN	2.7%	96.41%	0.89%
	LSTM	54.67%	21.44%	23.89%
	CNN-LSTM	2.31%	97.24%	0.44%
	4CNN-LSTM	0.62%	98.49%	0.89%
	4CNN-2LSTM	0.013%	99.82%	0.04%
Aggressive	1D-CNN	0.64%	1.86%	97.51%
	2D-CNN	1.37%	2.37%	96.26%
	LSTM	13.47%	14.85%	71.69%
	CNN-LSTM	0.37%	8.54%	91.09%
	4CNN-LSTM	0.21%	1.11%	98.67%
	4CNN-2LSTM	0.0%	0.16%	99.84%

4.5. Prediction Result Sharing via V2X

The proposed system transmits and receives the predicted driving behavior message in the form of BSM every 100 ms through V2X communication. As a result of the experiment, the number of transmitted and received BSM packets was measured the same. Figure 17 shows the distribution of communication delay. The average delay of V2X communication was observed as 4.8 ms, the minimum delay was 1.2 ms, and the maximum delay was 13.3 ms. Therefore, the proposed system has proven to be effective in predicting and sharing aggressive driving behavior even in real environments.

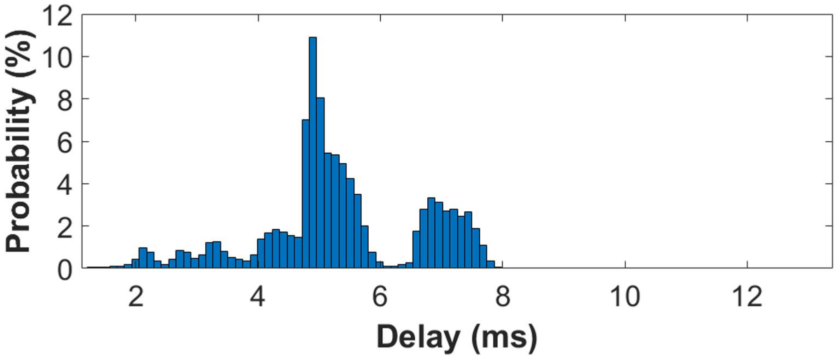


Figure 17. V2X communication delay distribution.

5. Conclusions

We propose a system that broadcasts the predicted behavior through V2X with a 4-layer CNN-2 stacked LSTM for driving behavior classification using only 3-axis acceleration. For 4-layer CNN-2 stacked LSTM training, 15% of real world DGIST campus driving data and 85% of virtual world driving data were used. The virtual world is similar to the real world, including about 240 traffic vehicles with randomly assigned driving motions, such as speeding and lane changes, as well as uphill, downhill and speed limit road environments. The virtual world driving data was collected by building DGIST campus, urban road and highway based on the connection between IPG CarMaker and a driving simulator that mimics real vehicles. The performance of the proposed system showed the best performance compared to 1D-CNN, 2D-CNN, LSTM, CNN-LSTM, and 4-layer CNN-LSTM, which are state-of-the-art techniques. The effectiveness of sharing driving behavior was evaluated through ACC scenarios constructed in simulations. The desired distance of the ego vehicle is changed through the assigned safety factor, according to the driving behavior of the target vehicle. If the target vehicle has aggressive driving behavior, the ego vehicle increases the desired distance. Compared to the case where the proposed system was not applied, the TTC increased by about 3 s or more. To evaluate the proposed system in the real world, we built a prototype, including IMU and Jetson TX2 board. As a result of driving tests on DGIST campus, it had a driving behavior classification performance of about 97% or more. The classified driving behavior is mounted on the vehicle BSM conforming to the J2735 standard via V2X communication and transmitted to nearby vehicles. As a result of the transmission and reception of the BSM, the expected driving behavior was shared with surrounding vehicles with a small delay of 4.8 ms on average. Therefore, systems and prototypes that classify driving behavior with only 3-axis accelerations and share it with the traffic system have been validated. Simulation tests and real world tests confirmed that the proposed system can help traffic safety. Through this, the development of various driver support systems, such as a system that warns the driver when the driver inputs an aggressive driving command, and more intelligent ACC systems, can be expected. Although a comprehensive classification of driving aggression has been performed, a more detailed classification may be necessary for driving safety. For example, there may be a scenario of changing lanes without a signal or changing several lanes at once. The remaining research task may be to construct a large number of scenarios for aggressive driving that may occur on the road and to classify driving aggression in detail.

Author Contributions: Conceptualization, S.K.K. and K.-D.K.; methodology, software, investigation, formal analysis, validation, simulation, writing—original draft preparation, S.K.K.; validation, simulation, J.H.S.; validation, simulation, J.Y.Y.; supervision, project administration, writing—review and editing, K.-D.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (No. 2019R1F1A1059496) and the DGIST R&D Program of the Ministry of Science and ICT (21-CoE-IT-01).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Prakash, C.D.; Akhbari, F.; Karam, L.J. Robust Obstacle Detection for Advanced Driver Assistance Systems Using Distortions of Inverse Perspective Mapping of a Monocular Camera. *Rob. Auton. Syst.* **2018**, *114*, 172–186. [CrossRef]
2. Feng, D.; Haase-Schutz, C.; Rosenbaum, L.; Hertlein, H.; Glaser, C.; Timm, F.; Wiesbeck, W.; Dietmayer, K. Deep Multi-Modal Object Detection and Semantic Segmentation for Autonomous Driving: Datasets, Methods, and Challenges. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 1341–1360. [CrossRef]
3. Rajaram, R.N.; Ohn-Bar, E.; Trivedi, M.M. RefineNet: Refining Object Detectors for Autonomous Driving. *IEEE Trans. Intell. Veh.* **2016**, *1*, 358–368. [CrossRef]
4. Santini, S.; Albarella, N.; Arricale, V.M.; Brancati, R.; Sakhnevych, A. On-Board Road Friction Estimation Technique for Autonomous Driving Vehicle-Following Maneuvers. *Appl. Sci.* **2021**, *11*, 2197.
5. Brunetti, A.; Buongiorno, D.; Trotta, G.F.; Bevilacqua, V. Computer Vision and Deep Learning Techniques for Pedestrian Detection and Tracking: A Survey. *Neurocomputing* **2018**, *300*, 17–33.
6. Wu, Z.; Qiu, K.; Gao, H. Driving Policies of V2X Autonomous Vehicles Based on Reinforcement Learning Methods. *IET Intell. Transp. Syst.* **2020**, *14*, 331–337.
7. Jung, C.; Lee, D.; Lee, S.; Shim, D.H. V2X-Communication-Aided Autonomous Driving: System Design and Experimental Validation. *Sensors* **2020**, *20*, 2903. [CrossRef]

8. Waschl, H.; Schmied, R.; Reischl, D.; Stolz, M. A Virtual Development and Evaluation Framework for ADAS—Case Study of a P-ACC in a Connected Environment. In *Control Strategies for Advanced Driver Assistance Systems and Autonomous Driving Functions*; Springer International Publishing: Cham, Switzerland, 2019; pp. 107–131.
9. Malik, S.; Khan, M.A.; El-Sayed, H. Collaborative Autonomous Driving-A Survey of Solution Approaches and Future Challenges. *Sensors* **2021**, *21*, 3783. [CrossRef] [PubMed]
10. Jahangiri, A.; Berardi, V. J.; Ghanipoor Machiani, S. Application of Real Field Connected Vehicle Data for Aggressive Driving Identification on Horizontal Curves. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 2316–2324. [CrossRef]
11. Hennessy, D.A.; Wiesenthal, D.L. Traffic Congestion, Driver Stress, and Driver Aggression. *Aggress. Behav.* **1999**, *25*, 409–423. [CrossRef]
12. Chung, W.-Y.; Chong, T.-W.; Lee, B.-G. Methods to Detect and Reduce Driver Stress: A Review. *Int. J. Automot. Technol.* **2019**, *20*, 1051–1063.
13. Yang, Q.; Chen, H.; Chen, Z.; Su, J. Introspective False Negative Prediction for Black-Box Object Detectors in Autonomous Driving. *Sensors* **2021**, *21*, 2819. [CrossRef]
14. Zhu, M.; Wang, Y.; Pu, Z.; Hu, J.; Wang, X.; Ke, R. Safe, Efficient, and Comfortable Velocity Control Based on Reinforcement Learning for Autonomous Driving. *Transp. Res. Part C Emerg. Technol.* **2020**, *117*, 102662. [CrossRef]
15. Road Traffic Injuries. Available online: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries (accessed on 9 September 2021).
16. Aydelotte, J.D.; Brown, L.H.; Luftman, K.M.; Mardock, A.L.; Teixeira, P.G.R.; Coopwood, B.; Brown, C.V.R. Crash Fatality Rates after Recreational Marijuana Legalization in Washington and Colorado. *Am. J. Public Health* **2017**, *107*, 1329–1331. [CrossRef]
17. American Automobile Association: Aggressive Driving: Research Update. Available online: https://www.aaafoundation.org/sites/default/files/AggressiveDrivingResearchUpdate2009.pdf (accessed on 9 September 2021).
18. Muhammad, K.; Ullah, A.; Lloret, J.; Ser, J.D.; de Albuquerque, V. H. C. Deep Learning for Safe Autonomous Driving: Current Challenges and Future Directions. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 4316–4336.
19. Islam, M.; Mannering, F. A Temporal Analysis of Driver-Injury Severities in Crashes Involving Aggressive and Non-Aggressive Driving. *Anal. Methods Accid. Res.* **2020**, *27*, 100128.
20. Tong, W.; Hussain, A.; Bo, W.X.; Maharjan, S. Artificial Intelligence for Vehicle-to-Everything: A Survey. *IEEE Access* **2019**, *7*, 1. [CrossRef]
21. IPG Automotive GmbH. CarMaker. Available online: https://ipg-automotive.com/products-services/simulation-software/carmaker/ (accessed on 9 September 2021).
22. Mihelj, J.; Kos, A.; Sedlar, U. Implicit Aggressive Driving Detection in Social VANET. *Procedia Comput. Sci.* **2018**, *129*, 348–352.
23. Ma, Y.; Zhang, Z.; Chen, S.; Yu, Y.; Tang, K. A Comparative Study of Aggressive Driving Behavior Recognition Algorithms Based on Vehicle Motion Data. *IEEE Access* **2019**, *7*, 8028–8038. [CrossRef]
24. Rongben, W.; Lie, G.; Bingliang, T.; Lisheng, J. Monitoring Mouth Movement for Driver Fatigue or Distraction with One Camera. In Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No.04TH8749), Washington, WA, USA, 3–6 October 2004; IEEE: New York, NY, USA, 2005.
25. Devi, M. S.; Bajaj, P. R. Driver Fatigue Detection Based on Eye Tracking. In Proceedings of the 2008 First International Conference on Emerging Trends in Engineering and Technology, Nagpur, India, 16–18 July 2008; IEEE: New York, NY, USA, 2008.
26. Dai, J.; Teng, J.; Bai, X.; Shen, Z.; Xuan, D. Mobile Phone Based Drunk Driving Detection. In Proceedings of the 4th International ICST Conference on Pervasive Computing Technologies for Healthcare, Munich, Germany, 22–25 March 2010; IEEE: New York, NY, USA, 2010.
27. Johnson, D.A.; Trivedi, M.M. Driving Style Recognition Using a Smartphone as a Sensor Platform. In Proceedings of the 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), Washington, DC, USA, 5–7 October 2011; IEEE: New York, NY, USA, 2011.
28. Singh, P.; Juneja, N.; Kapoor, S. Using Mobile Phone Sensors to Detect Driving Behavior. In Proceedings of the 3rd ACM Symposium on Computing for Development—ACM DEV'13, Bangalore, India, 11–12 January 2013; ACM Press: New York, NY, USA, 2013.
29. Yurtsever, E.; Lambert, J.; Carballo, A.; Takeda, K. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access* **2020**, *8*, 58443–58469. [CrossRef]
30. Chhabra, R.; Verma, S.; Krishna, C. R. A Survey on Driver Behavior Detection Techniques for Intelligent Transportation Systems. In Proceedings of the 2017 7th International Conference on Cloud Computing, Data Science & Engineering—Confluence, Noida, India, 12–13 January 2017; IEEE: New York, NY, USA, 2017.
31. Alkinani, M. H.; Khan, W. Z.; Arshad, Q. Detecting Human Driver Inattentive and Aggressive Driving Behavior Using Deep Learning: Recent Advances, Requirements and Open Challenges. *IEEE Access* **2020**, *8*, 105008–105030. [CrossRef]
32. Alluhaibi, S.K.; Al-Din, M.S.N.; Moyaid, A. Driver Behavior Detection Techniques: A Survey. *Int. J. Appl. Eng. Res.* **2018**, *13*, 8856–8861.
33. Carvalho, E.; Ferreira, B. V.; Ferreira, J.; de Souza, C.; Carvalho, H. V.; Suhara, Y.; Pentland, A. S.; Pessin, G. Exploiting the Use of Recurrent Neural Networks for Driver Behavior Profiling. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; IEEE: New York, NY, USA, 2017.

34. Shahverdy, M.; Fathy, M.; Berangi, R.; Sabokrou, M. Driver Behavior Detection and Classification Using Deep Convolutional Neural Networks. *Expert Syst. Appl.* **2020**, *149*, 113240.
35. Mekruksavanich, S.; Jitpattanakul, A. LSTM Networks Using Smartphone Data for Sensor-Based Human Activity Recognition in Smart Homes. *Sensors* **2021**, *21*, 1636. [CrossRef]
36. Zebin, T.; Sperrin, M.; Peek, N.; Casson, A. J. Human Activity Recognition from Inertial Sensor Time-Series Using Batch Normalized Deep LSTM Recurrent Networks. *Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.* **2018**, *2018*, 1–4. [PubMed]
37. Mumcuoglu, M. E.; Alcan, G.; Unel, M.; Cicek, O.; Mutluergil, M.; Yilmaz, M.; Koprubasi, K. Driving Behavior Classification Using Long Short Term Memory Networks. In Proceedings of the 2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE), Turin, Italy, 2–4 July 2019; IEEE: New York, NY, USA, 2019.
38. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural. Comput.* **1997**, *9*, 1735–1780. [CrossRef]
39. Li, F.; Liu, M.; Zhao, Y.; Kong, L.; Dong, L.; Liu, X.; Hui, M. Feature Extraction and Classification of Heart Sound Using 1D Convolutional Neural Networks. *EURASIP J. Adv. Signal Process.* **2019**, *2019*, 59. [CrossRef]
40. Kiranyaz, S.; Ince, T.; Gabbouj, M. Real-Time Patient-Specific ECG Classification by 1-D Convolutional Neural Networks. *IEEE Trans. Biomed. Eng.* **2016**, *63*, 664–675. [CrossRef]
41. Uddin, M. Z.; Hassan, M.M. Activity Recognition for Cognitive Assistance Using Body Sensors Data and Deep Convolutional Neural Network. *IEEE Sens. J.* **2019**, *19*, 8413–8419. [CrossRef]
42. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural. Comput.* **2019**, *31*, 1235–1270. [CrossRef]
43. Gupta, R.; Gupta, A.; Aswal, R. Time-CNN and Stacked LSTM for Posture Classification. In Proceedings of the 2021 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 27–29 January 2021; IEEE: New York, NY, USA, 2021.
44. AAA Exchange Advocacy. Communicatio. Education. Available online: https://exchange.aaa.com/safety/driving-advice/aggressive-driving/ (accessed on 28 October 2021).
45. Department of Motor Vehicles. New York State. Available online: https://dmv.ny.gov/about-dmv/chapter-8-defensive-driving (accessed on 28 October 2021).
46. Magdici, S.; Althoff, M. Adaptive Cruise Control with Safety Guarantees for Autonomous Vehicles. *IFAC-PapersOnLine* **2017**, *50*, 5774–5781. [CrossRef]