A Driving Decision Strategy(DDS) Based on Machine learning for an autonomous vehicle

SuRak Son
Computer Engineering
Catholic Kwandong University
Gangwon-do, Korea
Sonsur@naver.com

YiNa Jeong
Computer Engineering
Catholic Kwandong University
Gangwon-do, Korea
lupinus07@nate.com

ByungKwan Lee
Computer Engineering
Catholic Kwandong University
Gangwon-do, Korea
bklee@cku.ac.kr

***Abstract*—A current autonomous vehicle determines its driving strategy by considering only external factors (Pedestrians, road conditions, etc.) without considering the interior condition of the vehicle. To solve the problem, this paper proposes "A Driving Decision Strategy(DDS) Based on Machine learning for an autonomous vehicle" which determines the optimal strategy of an autonomous vehicle by analyzing not only the external factors, but also the internal factors of the vehicle (consumable conditions, RPM levels etc.). The DDS learns a genetic algorithm using sensor data from vehicles stored in the cloud and determines the optimal driving strategy of an autonomous vehicle. This paper compared the DDS with MLP and RF neural network models to validate the DDS. In the experiment, the DDS had a loss rate approximately 5% lower than existing vehicle gateways and the DDS determined RPM, speed, steering angle and lane changes 40% faster than the MLP and 22% faster than the RF.**

Keywords—Genetic Algorithm, Driving Strategy, Machine learning, Autonomous Vehicles

I. INTRODUCTION

Currently, global companies are developing technologies for advanced self-driving cars, which is in the 4th stage. Self-driving cars are being developed based on various ICT technologies, and the principle of operation can be classified into three levels of recognition, judgment and control. The recognition step is to recognize and collect information about surrounding situations by utilizing various sensors in vehicles such as GPS, camera, and radar. The judgment step determines the driving strategy based on the recognized information. Then, this step identifies and analyzes the conditions in which the vehicle is placed, and determines the driving plans appropriate to the driving environment and the objectives. The control step determines the speed, direction, etc. about the driving and the vehicle starts driving on its own. An autonomous driving vehicle performs various actions to arrive at its destination, repeating the steps of recognition, judgment and control on its own [1].

However, as the performance of self-driving cars improves, the number of sensors to recognize data is increasing. An increase in these sensors can cause the in-vehicle overload. Self-driving cars use in-vehicle computers to compute data collected by sensors. As the amount of the computed data increases, it can affect the speed of judgment and control because of overload. These problems can threaten the stability of the vehicle. To prevent the overload, some studies have developed hardware that can perform deep-running operations inside the vehicle, while others use the cloud to compute the vehicle's sensor data. On the other hand, existing studies use only real-time data such as images and sensor data currently collected from vehicles to determine how the vehicle is driving. This paper proposes a Driving Decision Strategy(DDS) Based on Machine learning for an autonomous vehicle which reduces the in-vehicle computation by generatings big data on vehicle driving within the cloud and determines an optimal driving strategy by taking into account the historical data in the cloud. The proposed DDS analyzes them to determine the best driving strategy by using a Genetic algorithm stored in the Cloud.

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2018R1A2B6007710).

II. RELATED WORK

Currently, most autonomous vehicles use machine learning (deep learning) to determine the driving of the vehicle. The following studies show how to run a machine running on a vehicle or in the cloud to affect the vehicle.

A. Vehicle Cloud

In [2], it proposed a synchro-ballistic control approach based on cloud model for the sake of reducing the angle error. First, the mechanism model of steering gear system is introduced. Second, the structure of synchro-control system of twin-rudder is proposed based on the master-slave control strategy. Third, synchro-ballistic controller based on cloud model is designed to solve the nonlinearity and uncertainty of system. Finally, the designed controller is tested via simulation under two different situations.

B. Machine Learning in Vehicle

In [3], it mines the double layers of hidden states of vehicle historical trajectories, and then selects the parameters of Hidden Markov Model(HMM) by the historical data. In addition, it uses a Viterbi algorithm to find the double layers hidden states sequences corresponding to the just driven trajectory. Finally, it proposes a new algorithm for vehicle trajectory prediction based on the hidden Markov model of double layers hidden states, and predicts the nearest neighbor unit of location information of the next k stages.

In [4], it proposes an optional ensemble extreme learning machine modeling technique to improve the wastewater quality predictions, due to the low accuracy and unstable performance of the conventional wastewater quality measurements. An extreme learning machine algorithm is added to the optional ensemble frame as the component model because it runs faster and provides better generalization performance than other machine learning algorithms. The ensemble extreme learning machine model gets over variations in different tests of simulations on a single model. The optional ensemble based on a genetic algorithm is used for ruling out some bad components from all available ensembles to diminish the computation complexity and increase the generalization performance.

III. A Design of A Driving Strategy Algorithm

The DDS analyzes the driving habits of an autonomous vehicle by using a genetic algorithm based on big data stored in the Cloud. The Genetic algorithm generates a gene sequence by using the autonomous vehicle's driving information and determines the optimal driving strategy by using a combination between genes. Because the genetic algorithm requires a lot of data for learning, the safety of an autonomous driving vehicle cannot be guaranteed if a gene sequence is learned only with real-time driving data. To address this, the DDS in this paper can select an optimal driving strategy for autonomous driving because it learns the genes by using the vehicle information accumulated past in the cloud.

A. An input of the DDS

The type of data used by the DDS consists of the training data sets for training a genetic algorithm and the output data analyzed after training. Here, one training data set has a training input and a training output. The training input means the values of the road state and the sensor message received from the vehicle, and the training output means the values of the Driving Information. The DDS normalizes data to learn the genetic algorithm. The training data sets used to learn the algorithm are expressed as equation (1).

$$T = \{\{TX_1, TY_1\}, \{TX_1, TY_2\}, ..., \{TX_{10000}, TY_{1000}\}\} \qquad (1)$$

Here, TX means a training input and TY means a training output. To learn the genetic algorithm, 10000 of the past data are used and the values for all slopes and curvatures must be the same. The DDS generates an initial set of chromosomes for RPM, speed, steering angle, and lane change to analyze the training input data. The RPM of the initial set of chromosomes will be selected as 1,000-6000, the speed, 0-250, and the steering angle, 0-90 and whether or not the lane is changed will be selected as 0 or1 randomly. The set of chromosomes generated by the genetic algorithm is evaluated according to the slope and curvature of the current road, and the optimal driving strategy is determined by the slope and curvature of the road as the generation passes. The DDS learns a set of chromosomes for 100 generations and applies a set of learned chromosomes to an autonomous vehicle. The DDS normalizes them by dividing RPM by 10000, steering angle by 100 and speed by 1000. Finally, all values of Y exist between 0 and 1. equation (2) represents a set of chromosomes in the genetic algorithm.

$$Y = \{y_r = RPM/10000,\ y_a = Angle/100,\ y_s = Speed/1000,\ y_{lc} = \text{Whether the vehicle has changed lanes}\} \qquad (2)$$

The DDS generates a set of initial 100 chromosomes normalized between 0 and 1.

B. A learning of the DDS

The DLS learns the genetic algorithm by receiving training data sets and an initial set of chromosomes. First, the DDS receives 100 sets of initial chromosomes from the DDS that are randomly set for learning RPM, speed, steering angle, and lane change. The definition of an initial set of chromosomes has already been described in section A. The DDS computes the suitability by using equation (3) for 100 sets of randomly generated chromosomes. A typical genetic algorithm compares one training data with a set of chromosomes to find the least different values, or to find the best values among chromosomes. However, various driving strategies on road driving can exist in the same environment. Therefore, the DDS of this paper compares 10000 training outputs with 100 sets of chromosomes to calculate the suitability of a set of chromosomes.

$$\mathrm{Ch}(Y_k) = -1 * \left|\frac{\sum_{i=1}^{10000}(TY_i - Y_k)}{100}\right| \qquad (3)$$

Here, TX means a training input and TY means a training output. To learn the genetic algorithm, 10000 of the past data are used and the values for all slopes and curvatures must be

$$\mathrm{P}(Y_k) = \frac{\mathrm{Ch}(Y_k)}{\sum_{i=1}^{10000}\mathrm{Ch}(Y_i) * 100} \qquad (4)$$

$$\text{parents} = \text{for}(k = 1, k < 100, k++)\ \text{random}(Y, Y_k, P(Y_k)) \qquad (5)$$

$$\text{child} = \frac{\text{parent1} + \text{parents2}}{2} \qquad (6)$$

1 to 100 and $\mathrm{Ch}(Y_k)$ means the suitability of Y_k. Once the suitability of each chromosome is computed, the DLS sets the probability of selection as much as the suitability of each chromosome, and randomly selects two from 100 chromosomes to combine the two genes. The DLS uses an Arithmetic Crossover method to combine these two genes. The Arithmetic Crossover method computes the average of two sets of parent chromosomes to produce child chromosomes. equation (4) determines the selected probability according to the suitability of each set of chromosomes, and equation (5) randomly selects the set of genes by applying $P(Y_k)$ from the entire chromosome set Y. equation (6) shows the generation of a child chromosomes.

The DDS generates 0.05% of the child genes as mutant genes randomly, and a set of mutant chromosomes by using reverse operations. The DDS randomly selects one of the child gene sets generated, and randomly selects one of the genes from that set. Next, it reverses the selected gene value using a fixed maximum value that the gene can have. In this way, the DDS makes 0.05 percent of the total set of genes mutant genes.

When a set of 100 child chromosomes is generated and the mutation operation is finished, the DDS again repeats the mutation computation process from equation (3). If the learning times are specified up to 1000, and a set of chromosomes with a suitability less than 0.001 is generated, the DDS terminates learning of the genetic algorithm and stores the speed, RPM, lane change, and steering angle for slope and curvature of the road.

C. An Usage of the DDS

The DDS computes an appropriate set of chromosomes for a particular slope and curvature of the driving road, generates an optimal driving table, and stores the set of chromosomes in the optimal driving table. Because the genetic algorithm computes the optimal driving strategy according to specific slopes and curvature by analyzing sufficient historical data, it does not need to perform deep-running, machine-running, etc. operations in real-time. The DDS stores the slope and curvature of the road used in the traing data set in the Slope and Curve field of the optimal driving table and the speed, steering angle, RPM, and lane change of the optimal driving strategy determined using the training data set in the Speed, Angle, RPM, and LaneChange field of the optimal driving table. The DDS searches for a set of chromosomes with similar slope and curvature values in the optimal driving table after inputting the slope and curvature of the vehicle which is currently in driving. If it exists, the DDS transmits it to the vehicle to implement the optimal driving strategy.

IV. EXPERIMENTAL RESULTS

This section shows the performance analysis of the DDS. To analyze the performance of the DDS, it was compared with a Multilayer Perceptron(MLP) and Random Forest(RF) in the accuracy of the analyzed driving strategies and the computation time. The experiment was conducted in a virtual environment and Table 1 shows the environment in which the experiment was conducted.

TABLE I. ENVIRONMENT OF THE EXPERIMENT

CPU	*GPU*	*RAM*	*OS*
Intel i5-7400	Geforce GTX 1050	DDR4 8GB	Window 10 Education

Fig. 1 shows the accuracy of the driving strategy computed by DDS, MLP and RF. To measure the accuracy of the DDS, the difference between the driving strategy determined by the DDS, MLP and RF and the safe driving of the actual vehicle is computed as a percentage. The accuracy of DDS, MLP and RF was measured in determining the 10 driving strategies. The experiment result shows that the DDS has a higher accuracy of about 0.3% than MLP and about 2.5% than RF. That is, the accuracy of the DDS does not differ significantly from that of the MLP and the DDS analyzes the data more accurately than the RF.

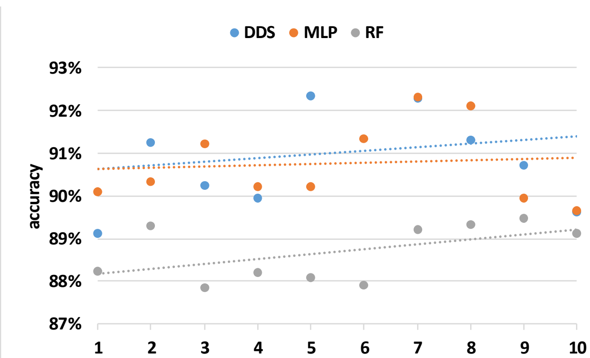

Fig. 1. The comparison of the DDS with MLP and RF in accuracy

Fig. 2 shows the computational time it takes for DDS, MLP and RF to determine the optimal driving strategy when they receive sensor messages from an autonomous vehicle. As with the computational time experiment, the computational time of DDS, MLP and RF were measured in determining 10 driving strategies. In the experiment, the DDS determined the optimal driving strategy about 22% faster than RF and 40% faster than MLP and sent it to an autonomous vehicle. That is, the DDS has a similar accuracy to MLP can determine how the vehicle is driving faster, has an higher accuracy of 25% than RF and can determine how the vehicle is driving 22% faster. In conclusion, the ODSS can transmit data about 10% more accurately than existing in-vehicle gateways, and determine the vehicle's optimal driving strategy faster and more accurately than neural network models and real-time machine running methods.

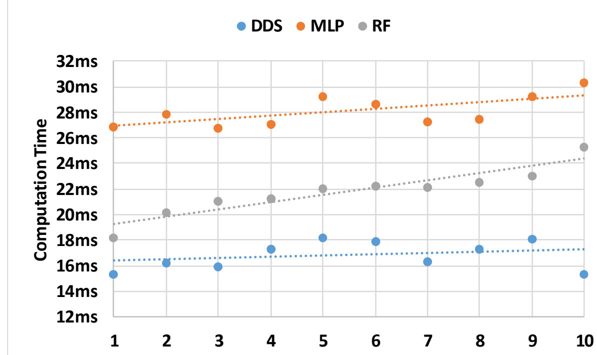

Fig. 2. The comparison of the DDS with MLP and RF in computation time

V. CONCLUSION

This paper proposed a Driving Decision Strategy. It executes the genetic algorithm based on accumulated data to determine the vehicle's optimal driving strategy according to the slope and curvature of the road in which the vehicle is driving and visualizes the driving and consumables conditions of an autonomous vehicle to provide drivers.

To verify the validity of the DDS, experiments were conducted on the DDS to select an optimal driving strategy by analyzing data from an autonomous vehicle. Though the DDS has a similar accuracy to the MLP, it determines the optimal driving strategy 40% faster than it. And the DDS has a higher accuracy of 22% than RF and determines the optimal driving strategy 20% faster than it. Thus, the DDS is best suited for determining the optimal driving strategy that requires accuracy and real-time.

Because the DDS sends only the key data needed to determine the vehicle's optimal driving strategy to the cloud and analyzes the data through the genetic algorithm, it determines its optimal driving strategy at a faster rate than existing methods. However, the experiments of the DDS were conducted in virtual environments using PCs, and there were not enough resources for visualization. Future studies should test the DDS by applying it to actual vehicles, and enhance the completeness of visualization components through professional designers.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2018R1A2B6007710).

REFERENCES

[1] Y.N. Jeong, S.R.Son, E.H. Jeong and B.K. Lee, “An Integrated Self-Diagnosis System for an Autonomous Vehicle Based on an IoT Gateway and Deep Learning, ” Applied Sciences, vol. 8, no. 7, july 2018.

[2] Yukiko Kenmochi, Lilian Buzer, Akihiro Sugimoto, Ikuko Shimizu, “Discrete plane segmentation and estimation from a point cloud using local geometric patterns, ” International Journal of Automation and Computing, Vol. 5, No. 3, pp.246-256, 2008.

[3] Ning Ye, Yingya Zhang, Ruchuan Wang, Reza Malekian, “Vehicle trajectory prediction based on Hidden Markov Model, ” The KSII Transactions on Internet and Information Systems, Vol. 10, No. 7, 2017.

[4] Li-Jie Zhao, Tian-You Chai, De-Cheng Yuan, “Selective ensemble extreme learning machine modeling of effluent quality in wastewater treatment plants, ” International Journal of Automation and Computing, Vol.9, No.6, 2012