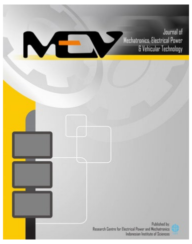

Journal of Mechatronics, Electrical Power, and Vehicular Technology 15 (2024) 105-116

Journal of Mechatronics, Electrical Power,
and Vehicular Technology

e-ISSN: 2088-6985
p-ISSN: 2087-3379

mev.brin.go.id

Design of intelligent cruise control system using fuzzy-PID control on autonomous electric vehicles prototypes

Joko Slamet Saputro [a, *], Miftahul Anwar [a], Feri Adriyanto [a], Agus Ramelan [a],
Putra Maulana Yusuf [a], Fakih Irsyadi [b], Rendra Dwi Firmansyah [c], Tri Wahyu Oktaviana Putri [d]

[a] Department of Electrical Engineering, Faculty of Engineering, Universitas Sebelas Maret
Jl. Ir Sutami no 36A Kentingan Jebres, Surakarta City, Central Java 57126, Indonesia

[b] Department of Electrical Engineering, Vocational College, Gadjah Mada University
Bulaksumur, Caturtunggal, Depok, Sleman Regency, Special Region of Yogyakarta 55281, Indonesia

[c] Research Center for Smart Mechatronics, National Research and Innovation Agency (BRIN)
Jl. Sangkuriang, Dago, Coblong, Bandung City, 40135, Indonesia

[d] Department of Electrical Engineering, Faculty of Engineering, University of Malaya
Pantai Baharu Street, Lembah Pantai, Kuala Lumpur, 50603, Malaysia

Abstract

Electric vehicles provide a solution for using alternative fuels, namely, electricity. Electric vehicles are used for short distances and intercity travel over long distances, increasing the risk of accidents. Cruise control is a technology embedded in vehicles to maintain stable speeds; this system will automatically adjust the vehicle's speed when motion changes cause changes in vehicle speed. This study aims to apply lidar sensors to detect distance in the Intelligent Cruise Control (ICC) system using the fuzzy-PID control method. Testing results were obtained at safe distance inputs of 5, 6, and 7 meters with various object distances. All the tests were carried out; the response systems were obtained with an average settling time of 5 seconds and an average overshoot of 1.53 %. Therefore, the proposed fuzzy-PID method works well for controlling intelligent cruise control systems in autonomous electric vehicle prototypes.

Keywords: fuzzy-PID; intelligent cruise control; autonomous electric vehicles; lidar sensors.

I. Introduction

Fossil fuel vehicles are a means of transportation that is widely used by Indonesian people to carry out daily activities. Based on data obtained from the Indonesian Central Statistics Agency (BPS), the number of two-wheeled motorized vehicles in 2020 reached 115 million and is increasing every year. The increasing number of fossil-based vehicles causes increased air pollution [1]. The city of Jakarta is one of the most densely populated cities with a large number of activities, with PM2.5 concentrations increasing and reaching a peak of 148 $\mu g/m^3$ (micrograms per cubic meter) [2]. To reduce air pollution, alternative energy sources in the form of electricity can be a reliable solution based on the idea of sustainable transportation by developing electric-powered vehicles [3]. Electric vehicles (EVs) have many advantages, including being environmentally friendly, less emissions, and being the most energy-efficient vehicle compared to any other

---

* Corresponding Author. jssaputro89@staff.uns.ac.id (J. S. Saputro)

https://doi.org/10.55981/j.mev.2024.877

Received 15 January 2024; revised 18 July 2024; accepted 20 July 2024; available online 31 July 2024

How to Cite: J. S. Saputro *et al.*, “Design of intelligent cruise control system using fuzzy-PID Control on autonomous electric vehicles prototypes,” *Journal of Mechatronics, Electrical Power, and Vehicular Technology*, vol. 15, no. 1, pp. 105-116, July, 2024.

vehicle [4]. With this benefit, the Indonesian government has declared an Acceleration Program for battery-based EVs for Road Transportation in 2019. It provides incentives for the domestic production of EVs [5].

Electric vehicles are capable of providing mobility transportation within cities and between cities. With the long distance traveled, of course, the risk of accidents will be higher. In Indonesia, traffic accidents have increased from year to year and the reported accidents were always more than 100,000 cases per year [6]. One of the causes of accidents is driver negligence due to loss of focus, resulting in a collision with another vehicle in front. The cruise control system is technology in vehicles that functions to maintain a constant speed without the need to step on the brake pedal and gas pedal. The cruise control system allows the vehicle to run consistently at a speed determined by the driver [7][8]. Along with developments in safety technology, the cruise control system can detect the speed and distance of the vehicle in front of it with the help of a lidar, which is Adaptive Cruise Control (ACC). By using the adaptive cruise control feature, the driver can save fuel [9] due to a more stable engine speed and increased safety in the vehicle used [10]. The adaptive cruise control system embedded in the vehicle is able to increase the speed on the gas pedal (throttle) if the distance set by the driver is too far from the vehicle in front and if the distance is too close to another vehicle, the system will reduce the speed by lowering the engine speed [8]. Cruise Control (CC) systems were created as part of Advanced Driver Assistance Systems (ADAS) to manage the vehicle's speed along the desired speed (referred to as cruise speed) [11]. Several control methods applied to ACC are Proportional Integral Derivation (PID) [7], fuzzy logic [12], Model Predictive Control (MPC) [13] and artificial intelligence approach [14].

Intelligent Cruise Control (ICC) system is part of the cruise control system that can be used on electric-powered vehicles, one of which is four-wheeled electric vehicles [15]. Some intelligent cruise control uses lidar as a sensor that measures the distance of an object by emitting laser light towards it and detecting the reflected light [16] and be able to automatically regulate speed in the intelligent cruise control system, which will be installed in electric vehicles [17]. It is hoped that by applying the lidar sensor to the intelligent cruise control system, the system created can work optimally to control the electric vehicle prototype automatically. The structure of the paper is presented as follows. The system and control method of ICC is described in section 2. The fuzzy-PID controller design and tuning method are also explored in that section. In section 3, the result is being discussed. Graphs regarding various safe distance testing are described, and control parameters are also explained for each test. Finally, the conclusion is given in the last section.

II. Materials and Methods

A. Electric vehicle prototype

Electric vehicles are vehicles powered by alternative fuel, namely electricity, driven by electric motors and accumulators. Electric vehicles utilize an electric motor that derives its power from a battery pack and is controlled by an Electronic Control Unit (ECU) [18]. In this research, the electric vehicle used is a prototype of an assembled electric vehicle with specifications for running in a limited environment.

The electric vehicle prototype used (Figure 1) has a frame design based on the Yukita 8988 children's car and has an AWD 4X4 (All Wheel Drive) driving mode with a high torque low-speed gearbox. The electric motor used is a brushed electric motor direct current with an operating power of 3.6 Watts at a maximum voltage of 12 Volts and a torque of 70 gram-centimeter force (gf.cm).

In the electric vehicle prototype, there are electronic components that have been modified from the initial components. These modified components are useful as supporting safety features so that the vehicle does not get out of line (Line-keeping), namely STM32 (Slave), NVIDIA Jetson (Master), motor DC driver, battery, and brushed motor direct current. All components are interconnected so that it can become a prototype electric vehicle.

There are six contacts made in Figure 2: box 1 is a board for the steering control system, box 2 is a driver for the steering and speed system, box 3 is a Jetson TX2 that functions for line detection processing, box 4 is a

Figure 1. Electric vehicle prototype.

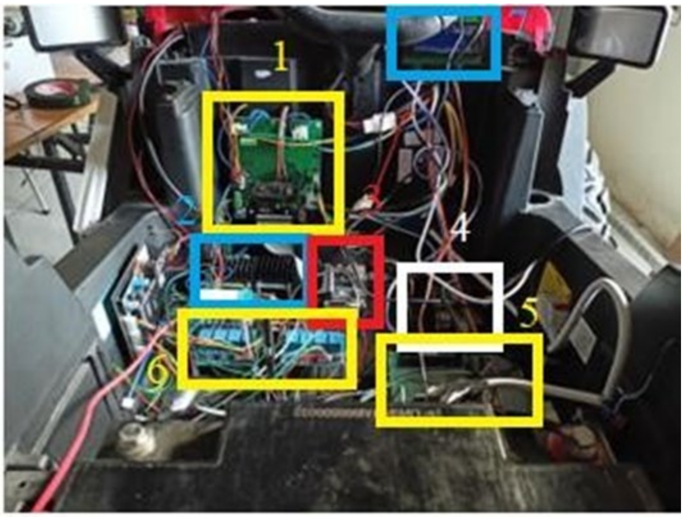

Figure 2. Electric vehicle electronic components.

raspberry pi 4 that functions for sign detection processing, box 5 is a board that functions for intelligent speed assistance actuators, box 6 is a remote PWM switching relay with driver and power for lane-keeping activation, and box 7 is a 20×4 LCD to display speed, detection results, and other information such as lane-keeping on/off conditions.

B. Cruise control

Cruise control system is a technology in vehicles that functions to maintain constant speed without the need to step on the brake and gas pedals [7]. The cruise control system allows the vehicle to run constantly at a speed determined by the driver [19]. Figure 3 is an illustration of the cruise control feature on a vehicle.

The Advance Driver Assistance System (ADAS) is an autonomous electronic system that functions to increase driver safety and security while driving [17][20]. ADAS is an integrated electronic control system where the control is classified based on the sensors used in the sensing system [19]. The sensing system is a type of control that learns and makes decisions based on inputs from the surrounding environment. Figure 4 shows the classification of ADAS features based on the sensors used on the vehicle and the sensors used [17].

Intelligent Cruise Control (ICC) system is part of the cruise control system that can be used on electric-powered vehicles, one of which is electric vehicles [21]. This research aims to apply a radar sensor to detect distance in the intelligent cruise control system that will be installed in electric vehicles. Radar sensors are generally used for parking assist systems so that parked vehicles do not hit nearby objects and to provide a warning to the driver if the vehicle is approaching an object in front of or behind it [8].

As shown in Figure 5 and Figure 6, Intelligent Cruise Control starts with an input sensor, which in this research uses lidar for distance input. The input is calculated by ICC using fuzzy logic. The output of fuzzy membership calculation proceeds into electric speed vehicle control. While the speed controller drives the actuator (electric motor), the speed itself will be the input of the entire algorithm to get better results, high accuracy, and decreasing error rate.

The Intelligent Cruise Control feature embedded in the electric vehicle prototype will help the driver automatically adjust the speed and the desired safe distance to avoid collisions with the vehicle in front of him. In this system, the speed will increase if the distance that has been set starts to move away and the speed will be reduced if the distance is approaching.

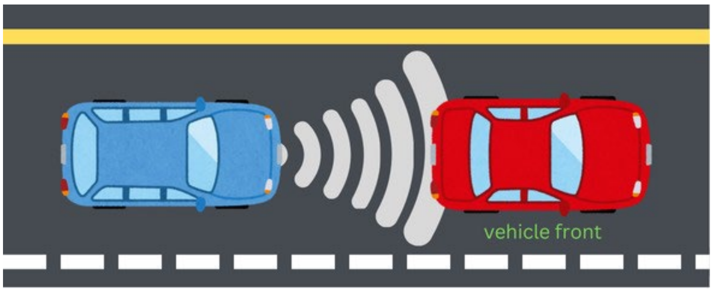

Figure 3. Illustration of cruise control.

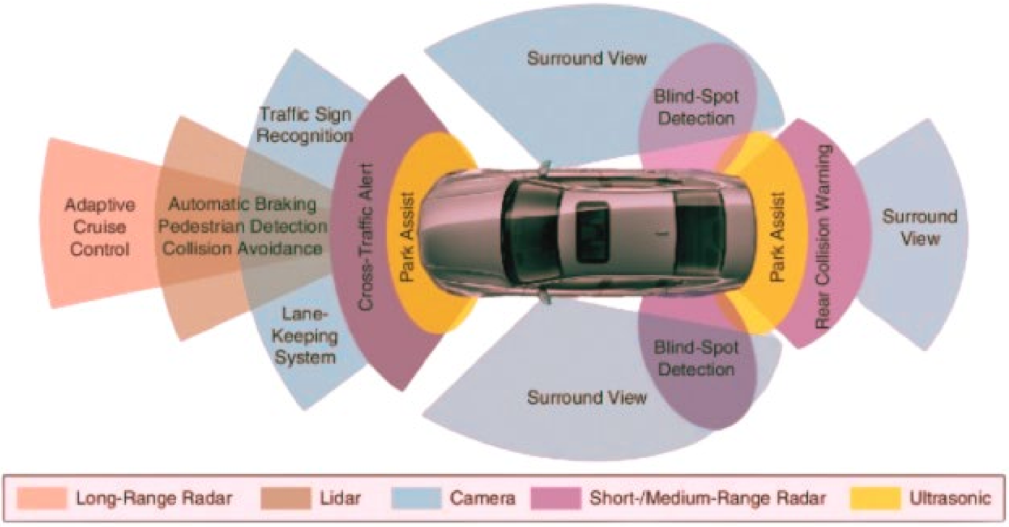

Figure 4. Classification of ADAS features on vehicles [17].

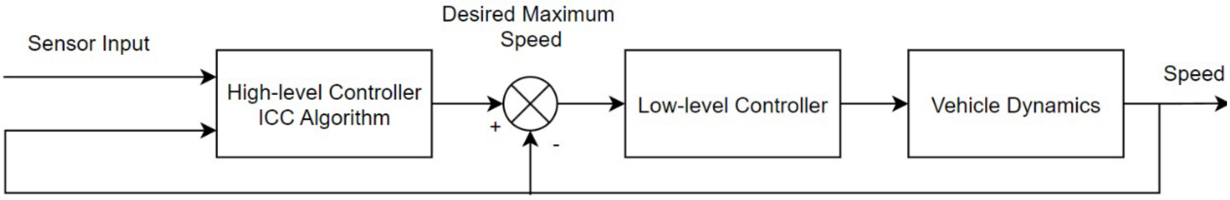

Figure 5. Intelligent cruise control block diagram [8].

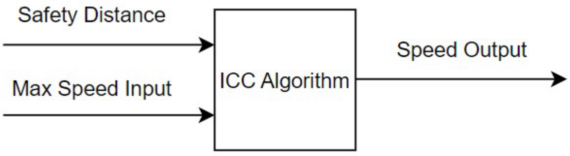

Figure 6. ICC algorithm proceed multi inputs [8].

C. Fuzzy logic

Fuzzy logic is a model designed to handle the transition of information that is incomplete, vague, inexact, and imprecise. A fuzzy set represents a category of objects characterized by a spectrum of membership grades [22]. A larger value indicates a higher level of set membership. A set defined by a membership function is called a fuzzy set. The most commonly used range of membership values is the 0.1 unit interval. Some fuzzy sets represent linguistic concepts such as low, medium, and high. This concept is used to define the state of a variable.

In the fuzzification process, a fuzzy set is obtained from each input variable, and each input has a relationship with the specified output fuzzy set [23]. The fuzzy set of inputs and outputs is defined by a membership function. Suppose A is fuzzy in X, which is defined in equation (1).

$$A = (x, \mu(x)| \mid x \epsilon X \tag{1}$$

A triangular membership function is defined by three parameters: 'a' (the lower bound), 'b' (the peak), and 'c' (the upper bound). The triangular fuzzy numbers are defined using three parameters: the minimum value, the most likely value, and the maximum value.

Values below 'a' and above 'c' have zero membership. The membership degree increases linearly from 0 to 1 as the input value moves from 'a' to 'b' and then decreases linearly from 1 to 0 between 'b' and 'c' as in equation (2).

$$Triangular = \begin{cases} 0 ; x \leq a, \\ \frac{x-a}{b-a} ; a \leq x \leq b, \\ \frac{c-x}{c-b} ; b \leq x \leq c, \\ 0 ; c \leq x, \end{cases} \tag{2}$$

D. Intelligent cruise control system design

The design of the system that has been created by the author consists of hardware design and software design, where the hardware design consists of DC motor control and speed regulation with a lidar sensor. Meanwhile, the software design is programming the Raspberry Pi 4 Model B, which is useful for getting readings from the sensors used in the system and controlling the motor with the algorithm created and the PID control system.

Figure 7 is a diagram of how the entire system works with the Raspberry Pi 4 Model B as the main controller of the system. On the power from the 12 Volts VRLA battery, there is an LM2596 *step-down module* to reduce the voltage to 5 Volts, which is useful as a power source for the TFMini-Plus lidar sensor and Hall Sensor Encoder on the 12 Volts motorbike. The Raspberry Pi gets input from the TFMini-Plus lidar sensor with universal asynchronous receiver-transmitter (UART) communication and gets a pulse count value or output signal from the encoder, which later in the algorithm will be converted into an rotation per minute (RPM) reading. After getting the data readings from the sensors, the Raspberry Pi then produces an output in the form of a pulse with modulation (PWM) value, which regulates the speed of the motor for the BTS7960 driver. With the algorithm created, the PWM value will automatically change based on the distance reading from the TFMini-Plus sensor and getting speed data from the encoder sensor.

In the first stage, motor control design was carried out using Raspberry Pi components, a BTS7960 motor driver, an encoder sensor, and a 12 Volts DC motor. The design of motor control components is carried out by programming the DC motor so that it can rotate at the desired speed by inputting the PWM value, as well as getting the RPM value when the motor rotates with a certain PWM value. Testing is carried out by looking at changes in motor rotation by inputting the PWM value; if it goes well, then it continues to the next stage.

In the second stage, an automatic motor speed control design was carried out based on the distance obtained from the TFMini-Plus lidar sensor. The design uses Raspberry Pi components, a BTS79060 motor driver, an encoder sensor, a 12 Volts DC motor, and a TFMini-plus lidar sensor. Testing is carried out by detecting objects in front of the lidar sensor by varying the actual distance and the distance received by the lidar sensor. If the speed can change with variations in the specified distance and runs well, then all stages will be combined into a unified system.

E. Fuzzy-PID control method

At this stage, it combines two control systems, namely Fuzzy and PID [24]. The fuzzy in this study has input error and delta error; the error is obtained from the difference between the setpoint speed and the actual speed, while the delta error is obtained from the difference between the error value and the previous error. The fuzzy output is the Kd value that will be entered into the PID process. A fuzzy logic controller is

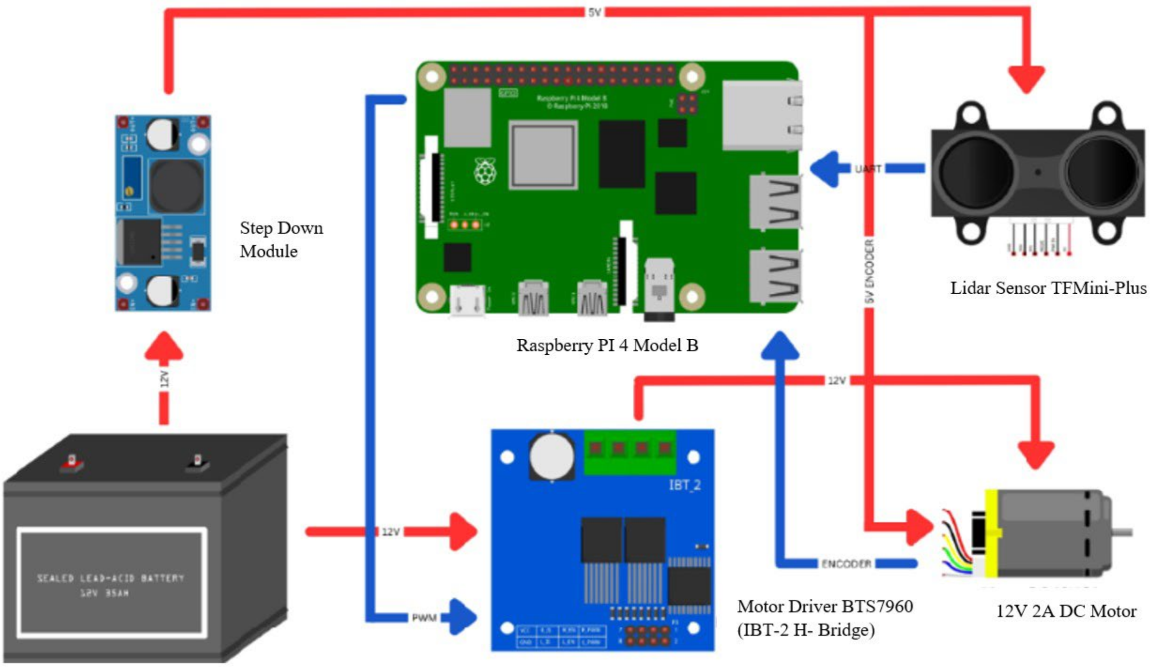

Figure 7. Diagram of electrical system.

presented, which consists of the two input variables, the deviation of the theoretical safe distance and relative distance.

Figure 8 explains the flow of the fuzzy-PID control design, where the error value from the input is entered into the fuzzy block, Kp block, and Ki block. At the output of the fuzzy block, the Kd value is obtained, which is then added to the Kp and Ki values. This value is used as input to the plant block where the plate in this study is a DC motor. By using fuzzy-PID control [25], it is hoped that it can produce a better system response [26] if only using one control system, namely PID. So, by using the fuzzy-PID method, the parameter results will be used in this research and as a reference for later analysis.

When using the fuzzy-PID control method with Simulink, the block can be seen in Figure 9. In the block, there are two controllers, namely the PID controller and the fuzzy logic controller. The PID controller uses a gain block and an integrator as a multiplier for the Ki value and two gains each function for the Kp and Ki values. In the fuzzy process, there is a fuzzy controller block, derivative block, and delay unit. The results of the fuzzy process and PID are combined to become input on the plate.

The response results using the fuzzy-PID control method are presented in Figure 10. The yellow graph represents input or step2, and the blue dot graph represents the calculation result of the transfer function, where the results of overshoot and settling time have lower values when compared to using PID control alone; this has an effect because the Kd parameter uses fuzzy control which will reduce overshoot and settling time.

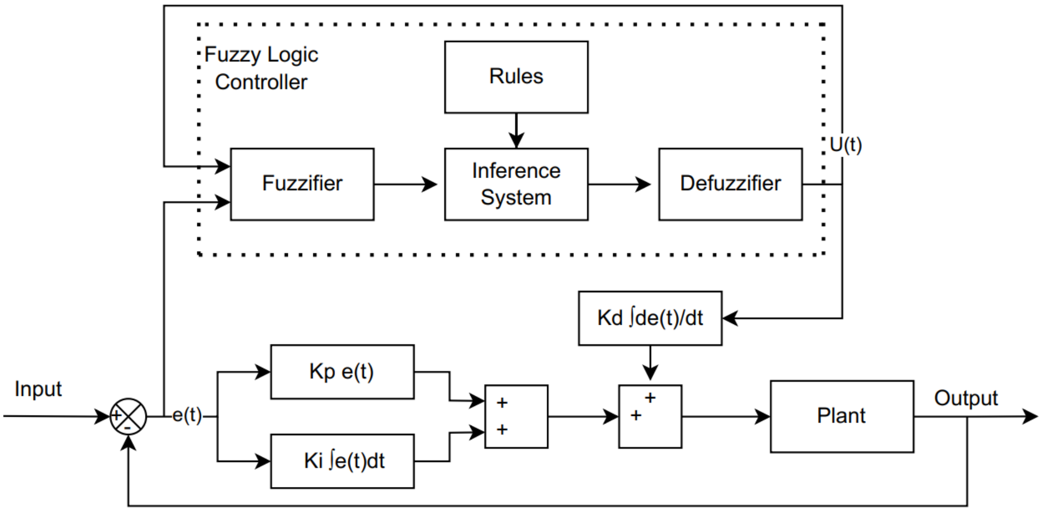

Figure 8. Fuzzy-PID control block diagram.

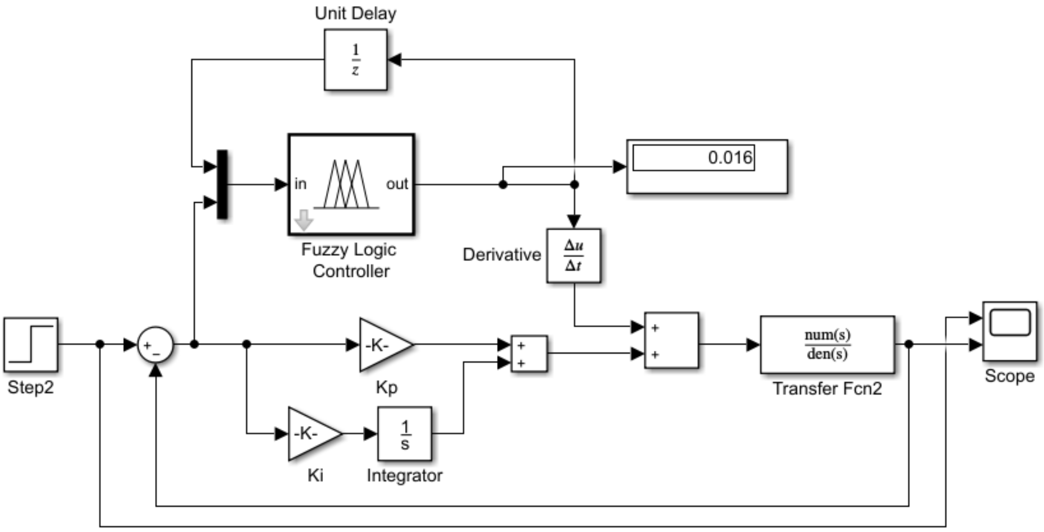

Figure 9. Fuzzy-PID control system simulink block.

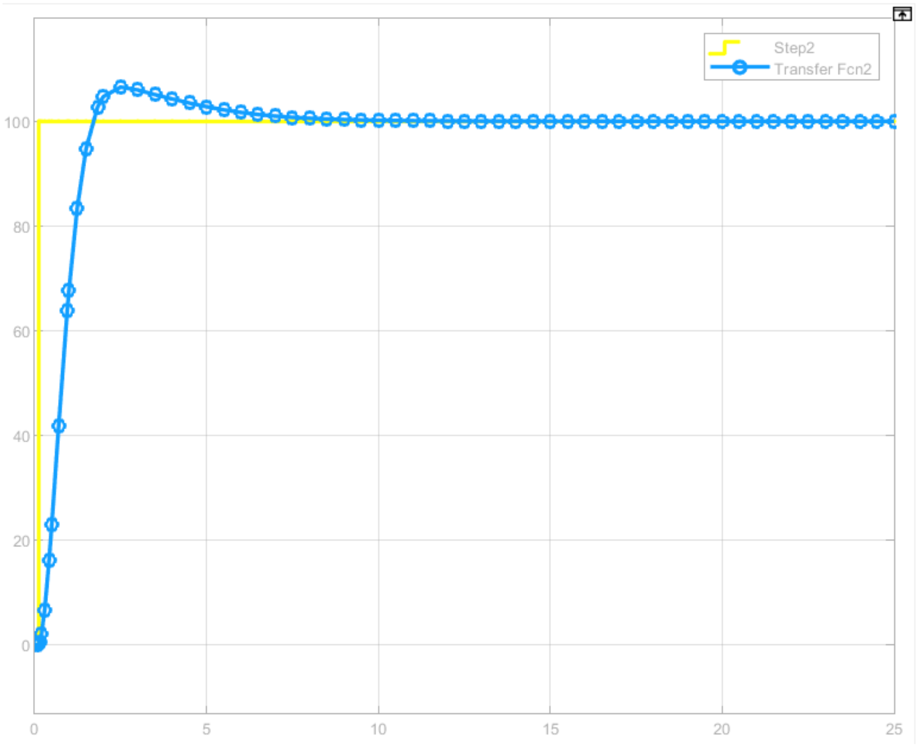

Figure 10. Response on fuzzy-PID control using transfer function.

III. Results and Discussions

A. Safe distance input testing 5 meters

Test results were taken from more than 5 meters with distance variations of 5 meters, 6 meters, 7 meters, and 8 meters. Testing at a distance of 5 meters resulted in values for the sensors and wheels, respectively 0.8 km/h and 0.79 km/h. When the distance sensor reading of the object detected is less precise, the algorithm system sends a control signal to move the motor. If the safe distance value is the same as the object distance, the algorithm will send a control signal with a value of 0. Figure 11 is a detection graph at a safe distance of 5 meters.

The error obtained from an object distance of 5 meters is 0.79 with a percentage error of 0.01 %. With an object distance of 6 meters, the speed readings on the sensor and measuring instrument have an error of 0.05 % and 0.01 %, respectively. The value reading from the sensor is very close and accurate, and the algorithm sends a control signal to the motor to run faster because the object distance is greater than the safe distance.

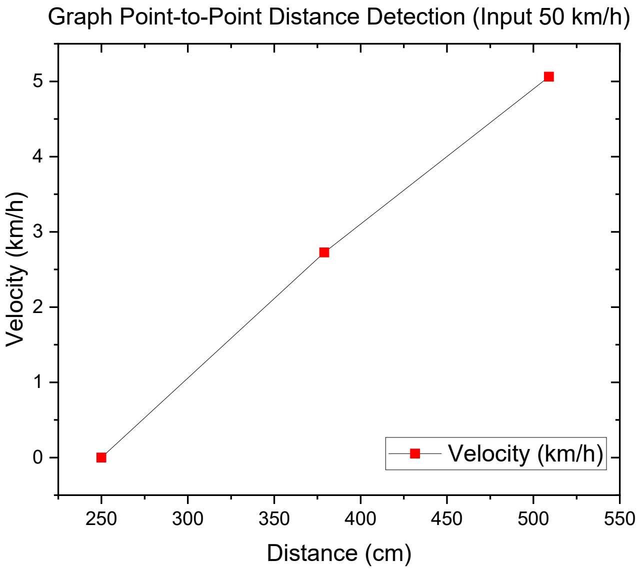

Figure 11. 5-meter detection distance against velocity graph.

Testing with an object distance of 7 meters has a faster speed compared to a distance of 6 meters, meaning that the algorithm system created to avoid collisions with objects in front of it works quite well. The error obtained in testing an object distance of 7 meters was 0.02 %. For testing with an object distance of 8 meters, the resulting speed will be faster. This is influenced by the control signal sent by the system; because the object is far enough away, it will send a control signal that is large enough to go faster.

In the test results for an object distance of more than 5 meters, it can be seen that the value at each point has increased proportionally. This indicates that the algorithm created can run well because by using a safe input distance of 5 meters and testing above 5 meters, the speed increases proportionally. Figure 12 shows a graph of the response between distance and speed. The response to distance measurements contains noise, which causes less stability in speed due to noisy distance readings. The results of the speed response graph can be analyzed for the settling time and overshoot parameters, which are presented in Table 1. The table shows the settling time and overshoot parameters at each increase in object distance. The settling time obtained was 5 to 6 seconds, while for overshoot value was quite small with a percentage below 2 %.

Table 1.
Settling time and overshoot distance of more than 5 meters.

Object point (meter)	Settling time (second)	Overshoot (%)
5 to 6	5.06	1.91
6 to 7	5.94	0.68
7 to 8	4.96	0.80

B. Safe distance input testing 6 meters

The test results were more than 6 meters with distance variations of 4, 5, 6, and 8 meters. Figure 13 is a detection graph at a safe distance of 6 meters. Testing at a distance of 4 meters produced values for the sensors and wheels, respectively, 0.6 km /h to 0.79 km/h. When the distance sensor reading of the object detected is less precise, the algorithm system sends a control signal to move the motor. If the safe distance value is the same as the object distance, the algorithm will send a control signal with a value of 0. The error obtained from an object distance of 5 meters is 0.79, with a percentage error of 0.04 %. With an object distance of 5 meters, the speed readings on the sensor and measuring instrument have an error of 0.07 % and 0.04 %, respectively. The value reading from the sensor is very close and accurate, and the algorithm sends a control signal to the motor to run faster because the object distance is greater than the safe distance.

Testing with an object distance of 6 meters has a faster speed compared to a distance of 5 meters, meaning that the algorithm system created to avoid collisions with objects in front works quite well. The error obtained in testing an object distance of 6 meters was 0.0 - 0.5 %. For testing with an object distance of 6 meters, the resulting speed will be much faster. This is caused by the control signal sent by the system because the object is far enough, it will send a control signal that is large enough to go faster. In the test results for an object distance of more than 6 meters, the value at each point has increased proportionally. This indicates that the algorithm created can run well because by using a safe input distance of 6 meters and testing above 6 meters, the speed increases proportionally.

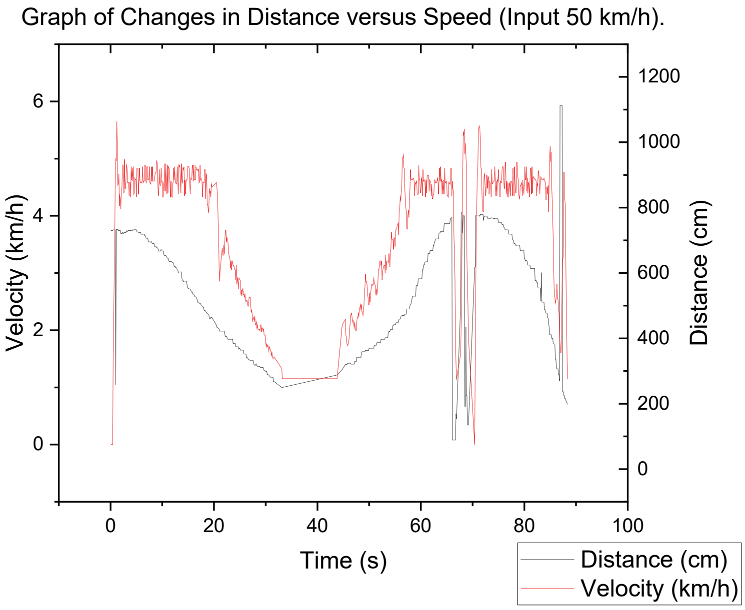

Figure 12. 5-meter detection distance test graph.

In Figure 14, a graph of the response between distance and speed is presented. The graph starts at 0 seconds, the distance input from lidar reads the object at 3.9 meters, and the motor starts to fluctuate. After the first response, the velocity consistently reaches the settling point of 6 km/h. While the object detected under 4 meters, the velocity slowed down; it is based on the Figure 13 algorithm rule. The response to distance measurements contains noise, which causes less stability in speed due to noisy distance readings. The results of the speed response graph can be analyzed for the settling time and overshoot parameters, which are presented in Table 2.

Table 2 shows these parameters for each increase in object distance. The settling time obtained was 5 to 6 seconds, while for overshoot, the value obtained was quite small, with a percentage below 2 %.

Table 2.
Settling time and overshoot distance of more than 6 meters.

Object point (meter)	Settling time (second)	Overshoot (%)
4 to 5	5.01	2.04
5 to 6	6.01	0.79
6 to 7	5.01	0.87

C. Safe distance input testing 7 meters

The test results were more than 7 meters with distance variations of 7 meters, 8 meters, 9 meters, and 10 meters. Figure 15 is a detection graph at a safe distance of 7 meters. Testing at a distance of 7 meters resulted in values for the sensors and wheels amounting to 0.9 km/h and 0.81 km/h, respectively. When the

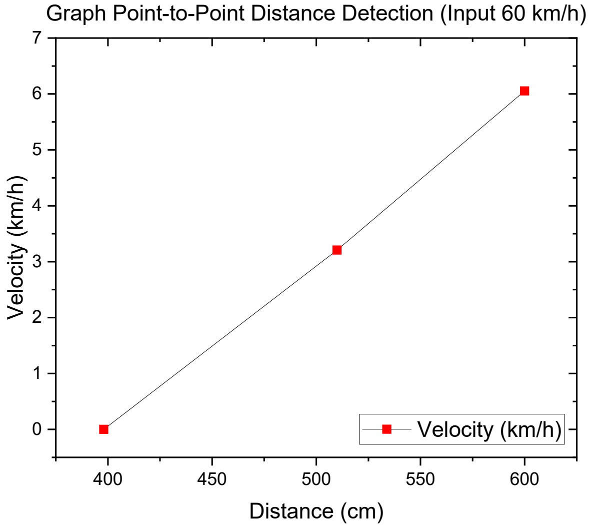

Figure 13. 6-meter detection distance against velocity graph.

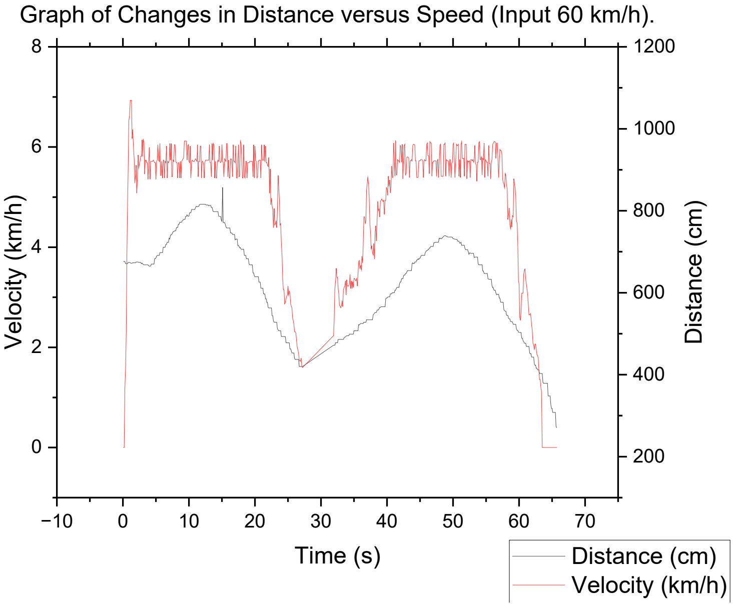

Figure 14. 6-meter detection distance test graph.

distance sensor reading of the object detected is less precise, the algorithm system sends a control signal to move the motor properly. If the safe distance value is the same as the object distance, the algorithm will send a control signal with a value of 0. The error obtained from an object distance of 7 meters is 0.12 %. With an object distance of 8 meters, the speed reading on the sensor and measuring instrument has an error of 0.01 %.

The value reading from the sensor is very close and accurate, and the algorithm sends a control signal to the motor to run faster because the object distance is greater than the safe distance. Testing with an object distance of 9 meters has a faster speed compared to a distance of 8 meters, meaning that the algorithm system created to avoid collisions with objects in front works quite well. The error obtained in testing an object distance of 9 meters was 0.03 %. For testing with an object distance of 10 meters, the resulting speed will be maximum; in other words, the distance of 10 meters is the maximum distance from the sensor reading so that the speed will be maximum with a value of 10 km/h, this is influenced by the control signal sent by the system because the object distance is maximum. Then it will send a large control signal to go faster. In the test results for an object distance of more than 7 meters, it can be seen that the value at each point increases proportionally, and the maximum speed is 10 km/h.

As shown in Figure 16, this indicates that the algorithm created can run well because by using a safe input distance of 7 meters and testing above 7 meters, the speed increases proportionally. The response to distance measurements contains noise, which causes less stability in speed due to noisy distance readings.

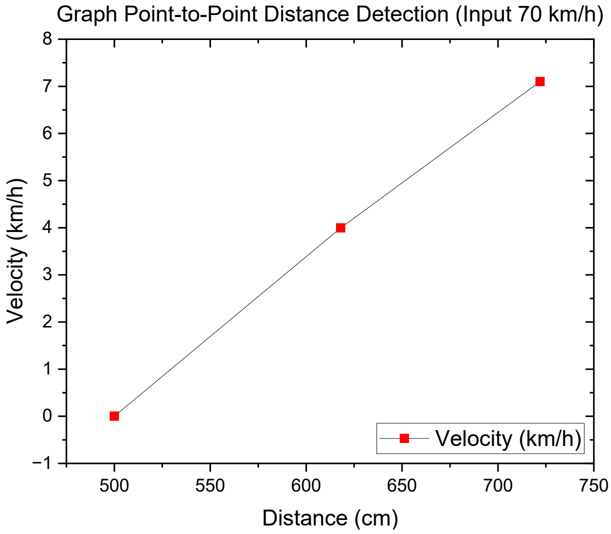

Figure 15. 7-meter detection distance against velocity graph.

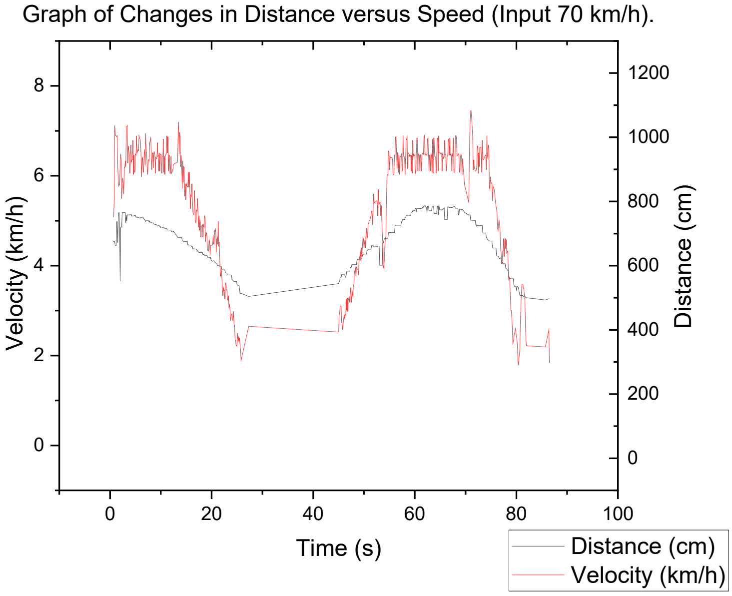

Figure 16. 7-meter detection distance test graph.

Table 3.
Settling time and overshoot distance of more than 7 meters.

Object point (meter)	Settling time (second)	Overshoot (%)
7 to 8	5.54	2.02
8 to 9	5.59	3.42
9 to 10	4.42	1.30

The proportional increase in speed observed when testing beyond a safe input distance of 7 meters demonstrates the robust performance of the developed algorithm. This linearity suggests that the algorithm effectively maintains a safe following distance. However, the presence of noise within the distance measurements introduces instabilities in the speed response. This is likely attributable to the algorithm's sensitivity to fluctuations in the distance sensor readings. A thorough analysis of the speed response graph would yield valuable insights into the system's transient characteristics. Specifically, the parameters of settling time and overshoot, as outlined in Table 3, are considered to quantify the algorithm's responsiveness and the degree to which it surpasses the desired speed setpoint.

Table 3 shows the settling time and overshoot parameters for each increase in object distance. The settling time obtained is ±5 seconds. While this settling time is reasonable, the overshoot value increases significantly when the object distance changes from 8 meters to 9 meters. This large overshoot is evident in the speed response graph. This indicates a potential need for refinement in the control system design. One approach to address this could be using a triangular membership function within a fuzzy logic controller. The triangular membership function would allow for smoother transitions in the control signal, potentially reducing the overshoot and improving stability.

IV. Conclusion

Intelligent cruise control system was successfully designed and works well by using a lidar sensor on an electric vehicle prototype using a Raspberry Pi controller, motor driver, lidar sensor, and DC motor. The intelligent cruise control system can work by adjusting speed automatically with safe distance input. The intelligent cruise control system uses the fuzzy-PID control method, where the PID parameter values are obtained using the fine-tuning method and fuzzy control to adjust the Kd value. Based on the results of all tests above, fuzzy-PID control is able to regulate the speed of the autonomous electric vehicle prototype based on a safe distance with fairly low settling time and overshoot values. As a further development, researchers can use optimal control methods or model predictive control to increase the reliability and speed of system response.

Acknowledgments

The authors would like to thank RKAT PTNBH Universitas Sebelas Maret (UNS) for the Fiscal Year 2023 for providing facilities and support for this research. The authors are grateful for the opportunity and resources provided by LPPM UNS, which have contributed to the successful completion of this study.

Declarations

Author contribution

All authors contributed equally as the main contributors to this paper. All authors read and approved the final paper.

Funding statement

The authors acknowledged the grant received under the Penelitian Unggulan Terapan (PUT-UNS) scheme with Research Assignment Letter Number 228/UN27.22/PT.01.03/2023.

Competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Additional information

Reprints and permission: information is available at https://mev.brin.go.id/.

References

[1] BPS-Statistics Indonesia, "Number of Registered Vehicles by Province and Type of Motor Vehicles (Units), 2022," Retrieved on January, 20, 2024.

[2] H. A. Azies, "Air Pollution in Jakarta, Indonesia Under Spotlight: An AI-Assisted Semi-Supervised Learning Approach." *Proceedings of 2023 International Conference on Data Science and Official Statistics (ICDSOS)* Vol. 2023 No. 1 (2023).

[3] F. Alanazi, "Electric Vehicles: Benefits, Challenges, and Potential Solutions for Widespread Adaptation," *Applied Sciences (Switzerland)*, Vol. 13, No. 10, p. 6016, May 2023.

[4] Z. S. Gelmanova *et al.*, "Electric cars. Advantages and disadvantages," in *Journal of Physics: Conference Series*, Vol. 1015, p. 052029, May 2018.

[5] M. F. N. Maghfiroh, A. H. Pandyaswargo, and H. Onoda, "Current Readiness Status of Electric Vehicles in Indonesia: Multistakeholder Perceptions," *Sustainability*, Vol. 13, no. 23, p. 13177, Nov. 2021.

[6] N. S. Kusumastutie, B. Patria, S. Kusrohmaniah, and T. D. Hastjarjo, "A review of accident data for traffic safety studies in Indonesia," in *IOP Conference Series: Earth and Environmental Science*, Institute of Physics, 2024.

[7] D. A. Purba, M. Nizam, H. Maghfiroh, M. R. A. Putra, and Inayati, "Prototyping Adaptive Cruise Control on Electric Motorcycle," *IOP Conf Ser Mater Sci Eng*, Vol. 1096, No. 1, p. 012076, Mar. 2021.

[8] N. C. Basjaruddin, Kuspriyanto, D. Saefudin, and I. Khrisna Nugraha, "Developing Adaptive Cruise Control Based on Fuzzy Logic Using Hardware Simulation," *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 4, No. 6, pp. 944–951, 2014.

[9] H. Zhang, C. Zhou, C. Wang, and W. Zhao, "An Energy Efficient Control Strategy for Electric Vehicle Driven by In-Wheel-Motors Based on Discrete Adaptive Sliding Mode Control," *Chinese Journal of Mechanical Engineering*, Vol. 36, No. 1, p. 58, Apr. 2023.

[10] Y. He, M. Makridis, G. Fontaras, K. Mattas, H. Xu, and B. Ciuffo, "The energy impact of adaptive cruise control in real-world highway multiple-car-following scenarios," *European Transport Research Review*, Vol. 12, No. 1, p. 17, 2020.

[11] M. M. Brugnolli, B. A. Angélico, and A. A. M. Laganá, "Predictive Adaptive Cruise Control Using a Customized ECU," *IEEE Access*, Vol. 7, pp. 55305–55317, 2019.

[12] K. Alomari, R. C. Mendoza, S. Sundermann, D. Goehring, and R. Rojas, "Fuzzy Logic-based Adaptive Cruise Control for Autonomous Model Car," in *ROBOVIS 2020 - Proceedings of the International Conference on Robotics, Computer Vision and Intelligent Systems*, SciTePress, pp. 121–130, 2020.

[13] J. Guo, W. C. Li, Y. Luo, and K. Li, "Model Predictive Adaptive Cruise Control of Intelligent Electric Vehicles Based on Deep Reinforcement Learning Algorithm FWOR Driver Characteristics," *International Journal of Automotive Technology*, Vol. 24, No. 4, pp. 1175–1187, Aug. 2023.

[14] D. Prastiyanto, E. Apriaskar, R. F. Ibrahim, A. Rumanda, A. A. Manaf, and I. Amelia, "Adaptive Cruise Control tuned by Genetic Algorithm for Safe Distance of Automated Vehicle," in *IOP Conference Series: Earth and Environmental Science*, Institute of Physics, 2023.

[15] US National Highway Traffic Safety Administration, "Evaluation of the Intelligent Cruise Control System Volume II-Appendices Final Report," 1999.

[16] B. Zhou, D. Xie, S. Chen, H. Mo, C. Li, and Q. Li, "Comparative Analysis of SLAM Algorithms for Mechanical LiDAR and Solid-State LiDAR," *IEEE Sens J*, Vol. 23, No. 5, pp. 5325–5338, 2023.

[17] V. K. Kukkala, J. Tunnell, S. Pasricha, and T. Bradley, "Advanced Driver-Assistance Systems: A Path Toward Autonomous Vehicles," *IEEE Consumer Electronics Magazine*, Vol. 7, No. 5, pp. 18–25, Sep. 2018.

[18] C. Dileep, K. Bharadvaj, J. Kavya, G. I. Rani, and G. Swetha, "Design and Performance Estimation of Electric Vehicle," in *2023 IEEE 3rd International Conference on Sustainable Energy and Future Electric Transportation (SEFET)*, pp. 1–5, 2023.

[19] O. Munyaneza, B. B. Munyazikwiye, and H. R. Karimi, "Speed control design for a vehicle system using fuzzy logic and PID controller," in *2015 International Conference on Fuzzy Theory and Its Applications (iFUZZY)*, IEEE, Nov., pp. 56–61, 2015.

[20] X. Li, R. Song, J. Fan, M. Liu, and F.-Y. Wang, "Development and Testing of Advanced Driver Assistance Systems Through Scenario-Based Systems Engineering," *IEEE Transactions on Intelligent Vehicles*, Vol. 8, No. 8, pp. 3968–3973, 2023.

[21] X. W. Chen, J. G. Zhang, and Y. J. Liu, "Research on the Intelligent Control and Simulation of Automobile Cruise System Based on Fuzzy System," *Math Probl Eng*, Vol. 2016.

[22] P. V. S. Reddy, "Fuzzy Logic Based on Belief and Disbelief Membership Functions," *Fuzzy Information and Engineering*, Vol. 9, No. 4, pp. 405–422, 2017.

[23] V. Ivanov, "A review of fuzzy methods in automotive engineering applications," *European Transport Research Review*, Vol. 7, No. 3, p. 29, 2015.

[24] I. Tejado, V. Milanés, J. Villagrá, J. Godoy, H. HosseinNia, and B. M. Vinagre, "Low Speed Control of an Autonomous Vehicle by Using a Fractional PI Controller," *IFAC Proceedings Volumes*, Vol. 44, No. 1, pp. 15025–15030, 2011.

[25] C.-C. Tsai, C.-C. Chan, Y.-C. Li, and F.-C. Tai, "Intelligent Adaptive PID Control Using Fuzzy Broad Learning System: An Application to Tool-Grinding Servo control Systems," *International Journal of Fuzzy Systems*, Vol. 22, No. 7, pp. 2149–2162, Oct. 2020.

[26] G. Prabhakar, S. Selvaperumal, and P. Nedumal Pugazhenthi, "Fuzzy PD Plus I Control-based Adaptive Cruise Control System in Simulation and Real-time Environment," *IETE J Res*, Vol. 65, No. 1, pp. 69–79, Jan. 2019.