

Enhancing Steering Responsiveness in Four-Wheel Steering Steer-by-Wire Systems Using Machine Learning [†]

Amarnathvarma Angani ¹, Teresa Talluri ^{2,3}, Myeong-Hwan Hwang ¹, Kyoung-Min Kim ^{1,2} and Hyun Rok Cha ^{1,*}

¹ Automotive Material & Component R&D Group, Korea Institute of Industrial Technology, 6, 7 Cheomdangwagi-ro 208beon-gil, Buk-gu, Gwangju 600-6114, Republic of Korea; angani@kitech.re.kr (A.A.); han9215@kitech.re.kr (M.-H.H.); kosk33@kitech.re.kr (K.-M.K.)

² Robotics, Korea National University of Science and Technology, Daejeon 34113, Republic of Korea; teressa@kwu.ac.kr

³ Department of Global Convergence, Kwangju Women's University, Gwangju 62396, Republic of Korea

* Correspondence: hrcha@kitech.re.kr

[†] Presented at 8th International Conference on Knowledge Innovation and Invention 2025 (ICKII 2025), Fukuoka, Japan, 22–24 August 2025.

Abstract

Steer-by-wire (SBW) systems in wheel-steering vehicles enhance maneuverability by eliminating mechanical linkages. However, they are susceptible to delays between steering input and pinion response, which can compromise control precision and safety. To mitigate these delays, we developed a machine learning-based compensation method employing a hybrid architecture of convolutional neural networks (CNNs) and gated recurrent units (GRUs) to predict and adjust pinion behavior in real time. The model was trained using experimental data collected from a four-wheel steering test platform, including steering angle inputs, motor signals, and pinion position feedback. By learning the relationship between steering commands and rack force, the model enables dynamic delay correction under both nominal and fault conditions. The system is implemented on an NXP microcontroller and validated through experimental testing, and compared with other hybrid model configurations for performance evaluation. The results demonstrate that the CNN–GRU approach reduces the average steering delay to 3 ms, outperforming conventional PID tuning methods while maintaining high accuracy and system stability.

Keywords: 4-wheel steering; steer-by-wire; steering delay; hybrid model; CNN; GRU; rack force estimation



Academic Editors: Teen-Hang Meen, Chun-Yen Chang and Cheng-Fu Yang

Published: 5 February 2026

Citation: Angani, A.; Talluri, T.; Hwang, M.-H.; Kim, K.-M.; Cha, H.R. Enhancing Steering Responsiveness in Four-Wheel Steering Steer-by-Wire Systems Using Machine Learning. *Eng. Proc.* **2025**, *120*, 58. <https://doi.org/10.3390/engproc2025120058>

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Recent advancements in vehicle technology have increasingly emphasized the development of steer-by-wire (SBW) systems [1]. These systems, particularly those incorporating active front- and rear-wheel steering, offer enhanced vehicle stability compared to conventional steering mechanisms. SBW systems reduce mechanical movement and friction, but their reliability remains a critical concern [2]. In 2013, Nissan became the first manufacturer to mass-produce SBW systems, integrating high safety standards and mechanical redundancy.

Although researchers have focused on improving the reliability of steering systems and reducing faults during driving, there remains a notable gap in addressing delays between steering commands and pinion angle responses [3]. According to ISO 26262 guidelines [4], permissible response delays must not exceed 0.5 milliseconds. Current

research efforts aim to reduce these delays by optimizing SBW architectures through frameworks that incorporate steering commands, pinion angle feedback, and rack force prediction to determine the front wheel angle.

A three-loop proportional–integral–derivative (PID) motor control strategy uses the steering motor angle as an input to the system controller. While this strategy is simple and generally stable, it suffers from instability due to motor dynamics, which limits its effectiveness in improving the overall performance. Figure 1 illustrates the experimental setup of the four-wheel steer-by-wire vehicle used in this study.

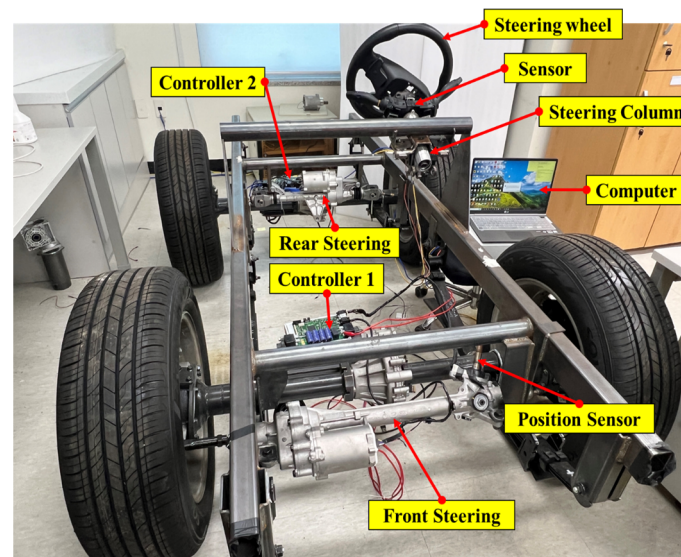


Figure 1. Experimental setup for the optimized steer-by-wire.

Rack displacement has been also researched through equivalent transmission ratio calculations, but these methods address only minor disturbances within the steering system. To enhance safety and reliability, we introduced a deep learning-based rack force prediction method designed to reduce delay and improve system robustness [5]. Efficient and practical algorithms were employed for vehicle signal estimation, leveraging adaptive and deep learning techniques to align with advanced sensor inputs and maintain high performance. Through the experimental determination of steering input and pinion angle, along with the collection of corresponding rack force data, response delay was corrected and minimized using the NXP model-based design toolbox (MBDT) implemented on an embedded controller. Rack force prediction was achieved using a hybrid CNN–GRU model, and the delay between the steering command and pinion angle response was significantly reduced through a hybrid CNN–GNN architecture.

The remainder of this paper is organized as follows: Section 2 presents the experimental model and the deep learning-based framework for delay reduction. Section 3 provides a detailed analysis of rack force prediction. Section 4 discusses the model-based study and compares the experimental results with the predicted outcomes. Finally, Section 5 concludes the paper with a summary of the findings and implications.

2. Optimization of Steer-by-Wire (SBW) System

2.1. Model Construction

Figure 2 presents the motor control strategy combining current, speed, and position control in a unified loop. The system uses cascaded PID controllers for regulating the current, speed, and position, ensuring precise motor behavior [6]. The sensor decoder measures rotor position (θ_m), which is processed through Clarke and Park transforms

to convert three-phase signals into the dq-axis for control. The inverse park transform converts the control signals back to the three-phase reference frame for space vector pulse width modulation generation and motor actuation. Additionally, reinforcement learning is used to adjust PID gains to improve performance under varying conditions [7]. This control architecture ensures accurate and responsive control of the motor in real time.

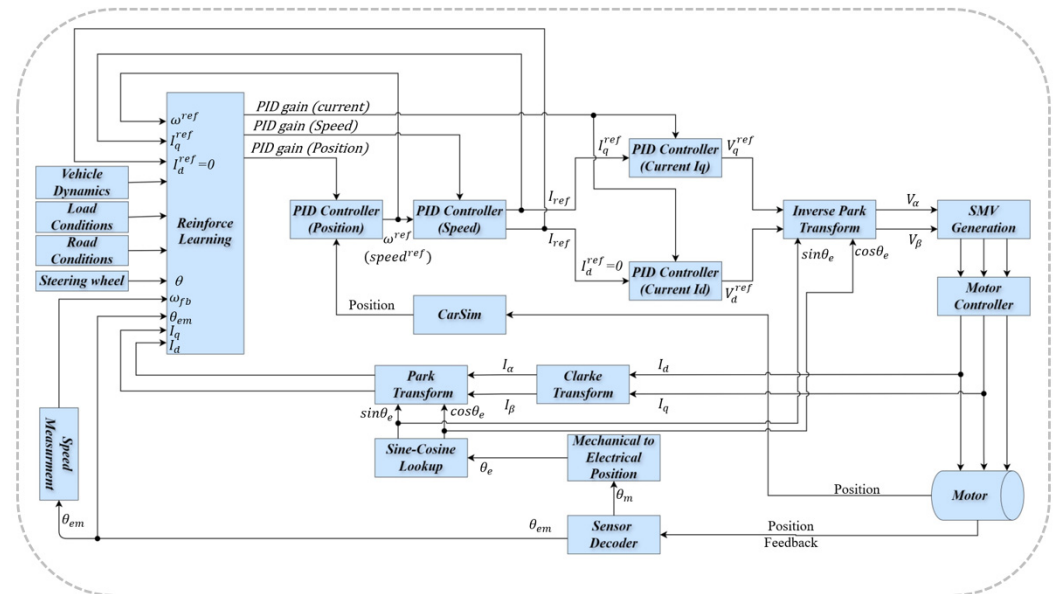


Figure 2. Control strategy of the 4WS NXP-based MBDT system.

2.2. Control Strategy for SBW

Figure 3 presents a hybrid control loop integrating AI prediction and motor control for vehicle systems. The input sensor data from the vehicle signals, such as speed, steering angle, or environmental data, are fed into the system. The deep neural network is used to train the model with CNN + GRU that predicts motor-related variables such as pinion angle or rack force based on the input data. A motor control strategy is employed to generate predictive values that are integrated into a Simulink-based control block, which utilizes PID controllers for regulating current, speed, and position, as illustrated in the previously referenced motor control diagram. Subsequently, these control commands are applied to the vehicle system model, targeting either the electric motor or the steering mechanism to simulate real-time behavior. The system outputs motor position, rack force, or speed, which serve as feedback signals. Within the feedback loop, the output is compared against the predicted values, and the resulting error is used to retrain or fine-tune the neural network. This process establishes a closed-loop learning-based control system that continuously adapts to improve performance.

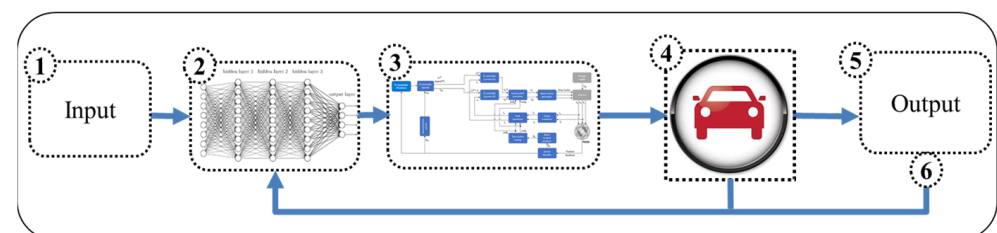


Figure 3. Hybrid learning and control architecture for vehicle motor system.

3. Prediction of Rack Force

3.1. Hybrid CNN and GRU Model

3.1.1. CNN-Based Learner

The steering system plays a critical role in the SBW architecture by capturing steering angle data, which is essential for predictive modeling. To ensure stable and accurate predictions, 2D CNN is employed to automatically extract common features from the input data. The CNN model is trained using the calculated pinion angle responses corresponding to steering commands and their associated delays [8]. To preserve essential features during the pooling process, the network is designed to retain all relevant information. The layers in the CNN are configured with 125 and 64 neurons, respectively, and the rectified linear unit (ReLU) activation function is utilized to mitigate overfitting. The mathematical expression is as follows:

$$f(x) = \max(0, x) \quad (1)$$

where x is the input value of the activation function.

3.1.2. GRU-Based Model

A gated recurrent unit (GRU)-based model is implemented to process continuous input data comprising steering angles, pinion angles, and rack force measurements [9]. This framework enables the generation of updated memory states for each input sequence, enhancing temporal feature learning [10]. The GRU model operates as follows: upon receiving input data, the memory state is updated; the output from the first GRU layer is passed to a second GRU layer; the output from the second layer is then forwarded to a dense layer following a dropout operation; and finally, the model produces a prediction. The dense layers in the GRU model are also configured with 125 and 64 neurons. The final output of the GRU network is the predicted rack force. Table 1 presents the parameters of the hybrid CNN–GRU model. To assess the model's effectiveness in predicting rack force and minimizing delay, two evaluation metrics were employed. The results demonstrate that the proposed hybrid model significantly enhances the performance of the optimized SBW system by improving prediction accuracy and reducing response latency. The hybrid model structure is presented in Figure 4.

MAE and RMSE are calculated using Equations (2) and (3).

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_{\text{actual}} - y_{\text{predicted}}| \quad (2)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_{\text{actual}} - y_{\text{predicted}})^2} \quad (3)$$

where n represents the total data points; i represents the index of each data point; y_{actual} is the actual measured value; and $y_{\text{predicted}}$ is the predicted value.

Table 1. Parameter settings of the proposed hybrid CNN and GRU model.

Model Parameter	Value
Time step	5
Convolution layer kernel size	3×3
Pooling layer kernel size	2×2
Convolutional layer activation function	ReLU
Loss function	MSE
Optimization iteration algorithm	Adam
Dropout ratio/%	10

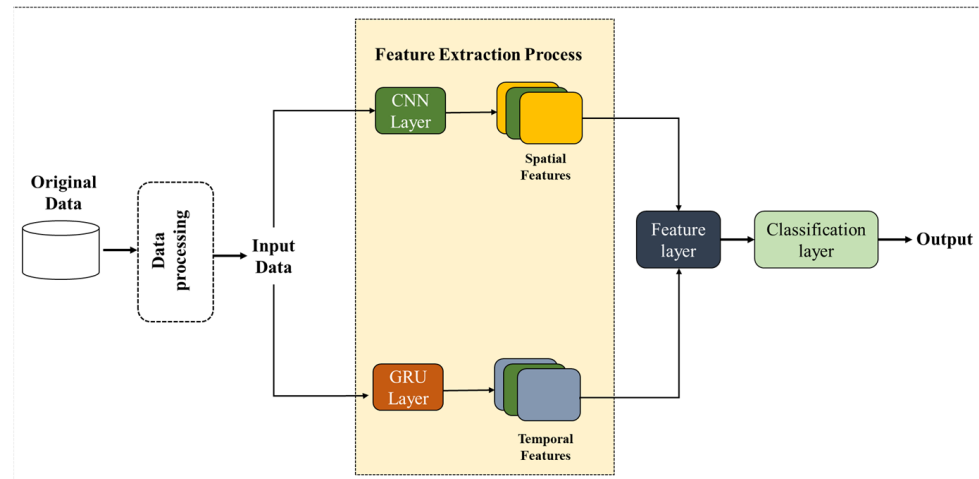


Figure 4. Hybrid model of the CNN and GRU model for the rack force prediction.

3.2. Rack Force Estimation

In the proposed model, the CNN extracts local spatial features from sequential inputs such as the steering angle and pinion response, capturing rapid variations and short-term patterns [11]. GRU processes these features over time to learn the temporal dependencies and dynamics of the steering behavior. This combination enables accurate rack force prediction by leveraging both the spatial and temporal characteristics of the input signals. The rack force (F_{rack}) acting on the steering system is an essential parameter, as it is determined by the moments generated at the tire–road contact patch, specifically the kingpin moment (M_{kp}). This is represented as follows:

$$F_{rack} = \frac{M_{kp}}{r_{gear}} \quad (4)$$

Here, F_{rack} is the force at the steering rack, and M_{kp} is the total moment acting on the kingpin (steering axis) of the wheel. This moment typically includes contributions from lateral tire force, self-aligning torque, and pneumatic trail. r_{gear} is the effective gear ratio (or transmission ratio) of the steering rack and pinion mechanism.

The kingpin moment M_{kp} itself can be further broken down, but for a general overview, this relationship is often sufficient. It emphasizes that the rack force is a result of forces at the wheel.

The self-aligning torque is a critical component influencing steering feel and rack force. It is the moment that tends to return the wheel to a straight-ahead position.

$$M_z = F_y \cdot t_p \quad (5)$$

Here, M_z is the self-aligning torque (moment about the vertical axis of the tire), F_y is the lateral force generated by the tire, and t_p is the pneumatic trail (the effective distance behind the wheel’s geometric center where the lateral force acts). This equation is used to explain one of the primary drivers of the forces that the steering system needs to counteract or transmit.

Data collected from the steering command and the pinion angle were sent to for training in the CNN and GRU hybrid model to estimate the rack force according to the steering and the pinion angle movement (Figure 5) [12]. The final output is the efficient output with the reduced delay. The novelty of the model is the rack force prediction of the steering column, which is achieved for the first time.

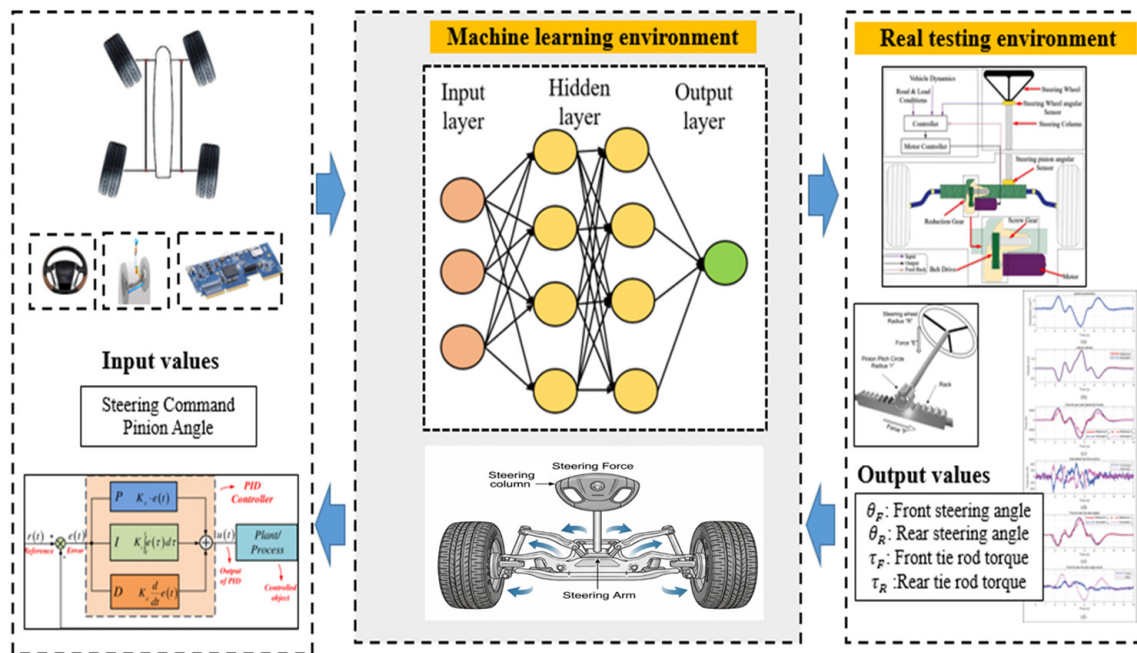


Figure 5. Hybrid model architecture to reduce delay.

4. Results and Discussion

The experiments were performed on a four-wheel vehicle equipped with an optimized steer-by-wire system. Initially, the controller was designed and implemented using MBDT. A PID controller was integrated to automatically minimize the delay between the steering command and the pinion angle response, enabling real-time monitoring of delay characteristics. This automation effectively eliminated the need for the time-consuming manual delay correction process.

The results show the effect of the prediction of the rack force, which reduces the manual effort of reducing the delay (Figure 6). The maximum pinion angle observed during the experiment was approximately $+200^\circ$, and all prediction models—CNN+GRU, BiGRU, GRU, and LSTM—were able to track this value with varying degrees of accuracy. Among these, the CNN+GRU model demonstrated the best performance, achieving a steady-state error of only 0.5° , indicating a highly precise response to the steering command. In contrast, the LSTM model showed the largest deviation, with an error of approximately 5° , highlighting its limitations in steady-state accuracy. In terms of rack force prediction, the CNN+GRU model outperformed the others, maintaining a prediction error within 5 N even during dynamic transitions, steering changes between 10 and 15 s. Meanwhile, LSTM exhibited larger deviations, exceeding 12 N, especially in high-response zones, making them less reliable for real-time control applications. Overall, the results confirm that the proposed CNN+GRU model significantly reduced delay and enhanced prediction accuracy, both in angular tracking and force estimation. This improvement not only lowers the need for manual tuning but also aligns well with the ISO 26262 functional safety standards for automotive steer-by-wire systems.

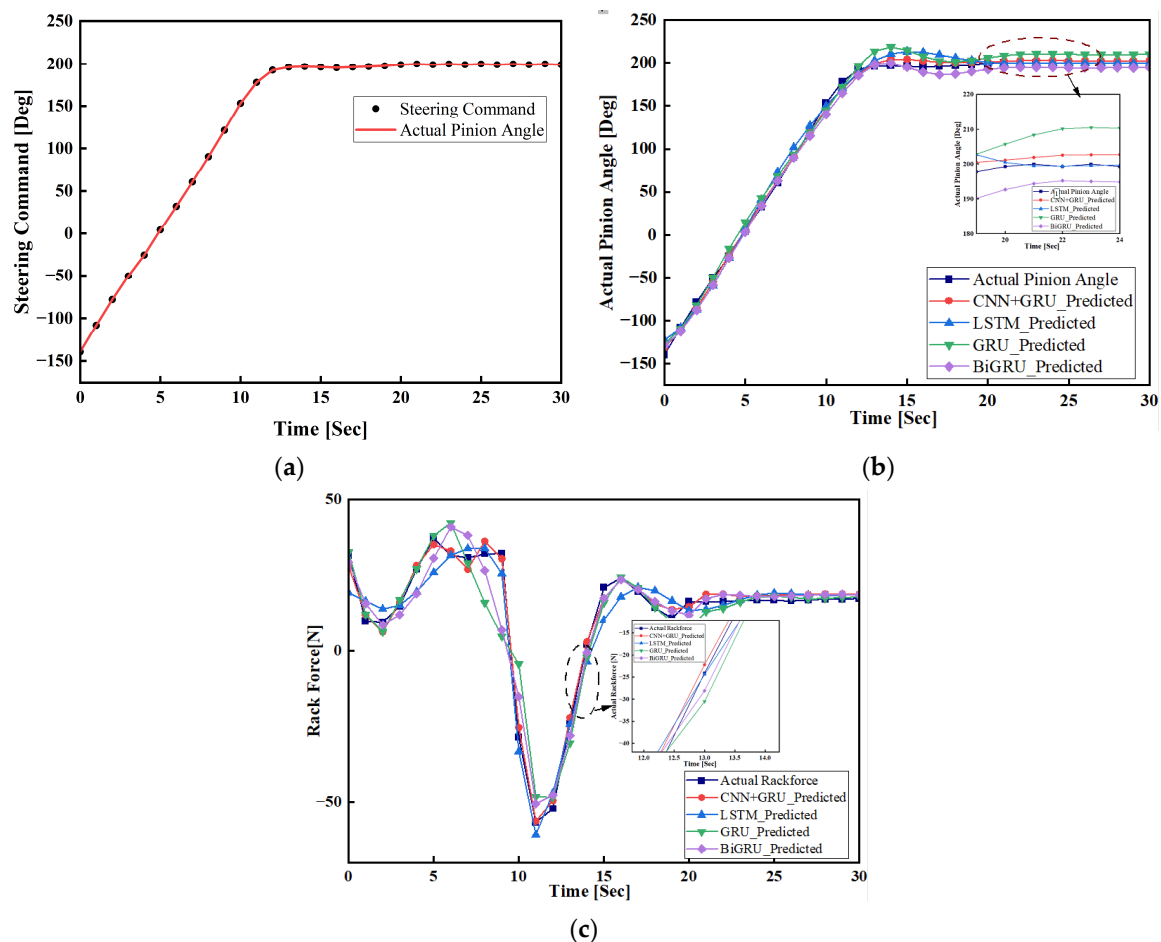


Figure 6. Experimental results: performance comparison of machine learning models for pinion angle tracking and rack force prediction in steer-by-wire systems: (a) experimental result for steering command and actual pinion angle; (b) predicting model for pinion angle; (c) predicting model for rack force.

5. Conclusions

The system developed in this study addresses the critical challenge of steering delays in four-wheel steering SBW systems through the development of a novel hybrid CNN+GRU model. The system facilitates real-time pinion behavior adjustment and accurate rack force prediction, a capability explored for the first time concerning the steering column. The experimental validation results revealed a significant reduction in the average steering delay to 3 milliseconds, notably surpassing traditional control methodologies. Furthermore, the CNN+GRU model exhibited superior accuracy in both pinion angle tracking (with a steady-state error of approximately 0.5°) and rack force prediction (maintaining errors within 5 N), demonstrating a distinct advantage over comparative models such as BiGRU, GRU, and LSTM. The system significantly enhances control precision and system stability and aligns with stringent ISO 26262 functional safety standards, substantially mitigating the need for laborious manual tuning.

Author Contributions: Conceptualization, A.A. and H.R.C.; methodology, A.A. and H.R.C.; software, A.A. and H.R.C.; validation, A.A. and K.-M.K.; formal analysis, A.A.; investigation, A.A. and H.R.C.; resources, H.R.C. and M.-H.H.; data curation, A.A.; writing—original draft preparation, A.A., T.T. and K.-M.K.; writing—review and editing, T.T.; supervision, H.R.C.; funding acquisition, M.-H.H., T.T. and A.A. are co-first authors as they contributed equally for this research. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Korea Institute of Industrial Technology as Development of an Industrial Skateboard with Longitudinal and Transverse Expansion Operations under Grant KEIT 20022022.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

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

References

1. Yu, L.; Bai, X. Control Strategies for Steer-By-Wire Systems: An Overview. *Technologies* **2025**, *13*, 6.
2. Yin, H.; Wang, Z.; Liu, J. Steer-by-wire control algorithm using a dual-layer closed-loop model. *Sci. Rep.* **2024**, *14*, 28536. [[CrossRef](#)] [[PubMed](#)]
3. Hang, P.; Xia, X.; Chen, X. Handling stability advancement with 4WS and DYC coordinated control: A gain-scheduled robust control approach. *IEEE Trans. Veh. Technol.* **2021**, *70*, 3164–3176. [[CrossRef](#)]
4. ISO 26262; Road Vehicles—Functional Safety (Part 2: ISO 17288-2:2011 Part 2: Steering-Pulse Open-Loop Test Method). International Organization for Standardization (ISO): Geneva, Switzerland, 2011. Available online: <https://www.iso.org/standard/53604.html> (accessed on 15 June 2025).
5. Weiskircher, T.; Fankem, S.; Ayalew, B. Rack force estimation for electric power steering. In Proceedings of the International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Boston, MA, USA, 2–5 August 2015; American Society of Mechanical Engineers: New York, NY, USA, 2015; Volume 57106, p. V003T01A007.
6. Fankem, S.; Weiskircher, T.; Müller, S. Model-based rack force estimation for electric power steering. *IFAC Proc.* **2014**, *47*, 8469–8474. [[CrossRef](#)]
7. Su, C.; Li, H.; Qiao, B.; Wu, X. Personalized Steering Feel Design for Steer-by-Wire Systems Based on the Rack Force Estimation. *Int. J. Automot. Technol.* **2023**, *24*, 1151–1161. [[CrossRef](#)]
8. Bhardwaj, A.; Gillespie, B.; Freudenberg, J. Estimating rack force due to road slopes for electric power steering systems. In *2019 American Control Conference (ACC), Philadelphia, PA, USA, 10–12 July 2019*; IEEE: New York, NY, USA, 2019; pp. 328–334. [[CrossRef](#)]
9. Bhardwaj, A.; Slavin, D.; Walsh, J.; Freudenberg, J.; Gillespie, R.B. Rack Force Estimation for Driving on Uneven Road Surfaces. *arXiv* **2020**, arXiv:2006.16319. [[CrossRef](#)]
10. Wang, Y.; Wang, Y.; Tie, M. Hybrid adaptive learning neural network control for steer-by-wire systems via sigmoid tracking differentiator and disturbance observer. *Eng. Appl. Artif. Intell.* **2021**, *104*, 104393. [[CrossRef](#)]
11. Hang, P.; Chen, X.; Fang, S.; Luo, F. Robust control for four-wheel-independent-steering EV with steer-by-wire system. *J. Dyn. Syst. Meas. Control* **2017**, *18*, 785–797.
12. Zhao, X.; Zhao, L. Steering angle tracking control of steer-by-wire system with prescribed performance under sensor failure. *Mechatronics* **2025**, *161*, 106354. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.