Event-Triggered Deep Q-Network Control for Driving Safety and Performance in Vision-Based Autonomous Vehicles*

Sejin Kwon[1], Sang Jun Lee[2], Wookyong Kwon[3] and Yongsik Jin[3]

Abstract— **In this study, we investigate a novel event-triggered control scheme to enhance driving performance and safety in autonomous vehicles. Although the proportional–integral–derivative (PID) control algorithm is widely used for driving control in most commercial vehicles due to its simplicity and ease of implementation and maintenance, it suffers from inherent performance and safety limitations. To overcome these limitations, we propose an event-triggered control scheme using deep Q-networks (DQNs). In the proposed approach, a DQN-based auxiliary controller is implemented to support the PID controller of the autonomous vehicle when the driving errors increase significantly. An event-triggering criterion is established to specify the moments when the DQN-based auxiliary controller intervenes. The DQN is activated when the lane-keeping error of the PID controller exceeds a predefined threshold. The effectiveness of the proposed approach is verified through an autonomous lane-keeping control task using the Udacity simulator. Simulation results show that the proposed method enhances both driving performance and safety. Moreover, the event-triggered control scheme reduces computational burden.**

I. INTRODUCTION

As autonomous driving becomes commercialized, achieving reliable and adaptive control remains a critical challenge, especially under unknown dynamic or uncertain driving conditions [1]. The proportional–integral–derivative (PID) controller remains widely used in commercial vehicles due to its simplicity and ease of deployment [2]. However, because of its simplicity, the PID controller suffers from fundamental limitations in unknown dynamics and uncertain environments. PID control, in particular, lacks the adaptability needed to handle nonlinearities and sudden changes commonly encountered in real-world driving, especially when facing unexpected situations that were not considered during the controller tuning phase. Moreover, it cannot appropriately respond to abnormal conditions.

*This work was supported by the Institute of Information & Communications Technology Planning & Evaluation(IITP)–Innovative Human Resource Development for Local Intellectualization program grant funded by the Korea government(MSIT)(IITP-2025-RS-2024-00439292), and Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government [25ZD1130, Regional Industry ICT Convergence Technology Advancement and Support Project in Daegu-GyeongBuk (Robot)].*Corresponding authors: Yongsik Jin*

[1]Sejin Kwon is with the Department of Electronic and Electrical Engineering, Dankook University, Gyeonggi-do, Republic of Korea tpwls836@dankook.ac.kr

[2]Sang Jun Lee is with Division of Electronic Engineering, Jeonbuk National University, Jeonju, Republic of Korea (e-mail: sj.lee@jbnu.ac.kr).

[3]Yongsik Jin and Wookyong Kwon are with the Intelligent Robot Systems Laboratory, Electronics and Telecommunications Research Institute (ETRI), Daegu, Republic of Korea. yongsik@etri.re.kr, wkwon@etri.re.kr

To overcome these limitations, reinforcement learning (RL) has emerged as a promising alternative [3], [4], [5], [6]. RL agents learn control policies through interaction with the environment, allowing them to handle complex dynamics that traditional model-based approaches cannot easily capture. In particular, value-based methods such as Deep Q-Networks (DQNs) have shown strong performance in discrete control tasks and have been applied to various applications [7], [8]. However, fully replacing classical controllers with RL in safety-critical domains remains challenging due to issues such as training instability, lack of robustness across scenarios, and high computational demand. In particular, the high computational cost of RL makes real-world implementation challenging.

On the other hand, event-triggered control has received increasing attention in recent years as a means to improve control efficiency by reducing unnecessary computations and interventions [9], [10], [11], [12], [13]. Unlike time-triggered approaches, which continuously execute control actions regardless of system state, event-triggered strategies activate control updates only when a specified condition is met—typically when system performance degrades beyond a predefined threshold. By triggering control updates only when necessary, this scheme minimizes resource usage and enables the practical deployment of complex controllers like DQN within real-time systems. Previous methods have aimed to conserve network resources by reducing how frequently essential control inputs are transmitted. However, our method addresses event-triggered control from a new perspective to enhance control performance and safety. This topic has not been sufficiently studied, indicating room for further investigation. Instead of fully relying on RL, selectively activating it when needed is a more efficient approach. In vehicle control, a PID controller can serve as the primary module for lane keeping under normal conditions, while a DQN-based auxiliary controller is activated through an event-triggered mechanism to prevent the vehicle from entering unsafe states.

In this paper, we propose a novel event-triggered hybrid control framework that enhances the performance and safety of autonomous lane-keeping control. A DQN-based auxiliary controller is trained to complement the behavior of a standard PID controller. An event-triggering condition based on lane deviation error is used to determine when the DQN should intervene, allowing it to take over control in complex scenarios where PID performance is insufficient. Thus, the proposed auxiliary controller is activated when the lane tracking error exceeds the threshold of the PID controller, and it functions to reduce the error. The proposed approach is implemented and validated in the Udacity self-driving car simulator, demonstrating improved lane-keeping performance, especially in challenging road segments such as curves and turns. Through this work, we show that selective activation of RL-based controllers can enhance system adaptability without sacrificing the stability and efficiency of classical control methods.

II. PROPOSED APPROACH

In this section, we present a hybrid control system for vision-based autonomous driving. First, the driving error in the image is defined. Then, the proposed control system and the DQN model employed are described.

A. Vision-based Autonomous Vehicle

To estimate the position of the lane, we employed a deep learning-based object detection approach using a customized YOLOv11 model fine-tuned for lane detection [14]. To train the YOLOv11 model, we constructed a custom dataset by annotating lane boundaries in frames captured from the center camera of the Udacity simulator. Roboflow was used to manage the annotation process and generate training-ready datasets with appropriate augmentations. For each frame, the YOLOv11 model detects the bounding boxes of the left and right lane markings. Rather than using full boundary coordinates, the center points of these bounding boxes are extracted and used to approximate the lane center.

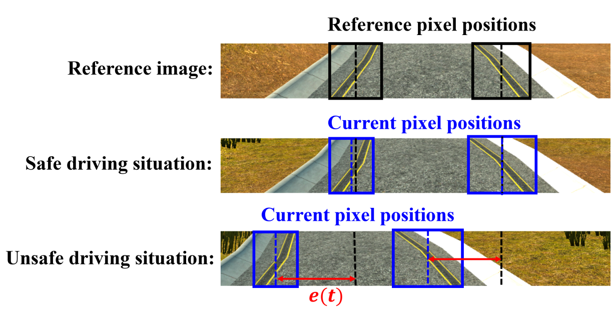

Fig. 1. Vison-based lane detection for vehicle control

To compute the lane-following error, we adopted a simplified form of Image-Based Visual Servoing (IBVS) [15]. The positional deviation is calculated as the difference between the detected lane center and a predefined reference point that represents the ideal lane centerline in image pixel coordinates. Although this approach does not use the full geometry of the lane boundaries, it serves as a practical solution for real-time control and allows the lane-following system to respond quickly to changes in lane position. These computed deviations are used as input to both the PID controller and the DQN-based auxiliary controller. Fig. 1 presents lane detection results along with the corresponding error computation. The black and blue boxes indicate the detected lanes. The black dashed line represents the reference lane position in the image, while the blue dashed line denotes the current lane position in the image.

B. Event-Triggered Control with Deep Q-Network

The proposed lane tracking error model can be expressed as follows:

$$x(t+1) = A(t)x(t) + B(t)(u_0(t) + \rho(e(t))u_r(t)), \qquad (1)$$

where $x(t)$ is the state of the error system. $u_0(t)$ is the PID control input, and $u_r(t)$ is the control input of DQN. $A(t)$ and $B(t)$ are time-varying matrices that represent vehicle lateral dynamics. $\rho(e(t))$ is the triggering function and is defined as follows:

$$\rho(e(t)) = \begin{cases} 1 & \text{if } e(t)^T Pe(t) \geq T \\ 0 & \text{if } e(t)^T Pe(t) < T, \end{cases} \qquad (2)$$

where T is a predefined threshold and P is a weighting constant.

As shown in Fig. 2, when the weighted error is larger than a predefined criterion, the DQN controller is used alongside the PID controller; otherwise, only the PID controller is used. Namely, in safe driving situations, only the PID controller is employed; otherwise, in unsafe situations, the DQN controller is used alongside the PID controller.

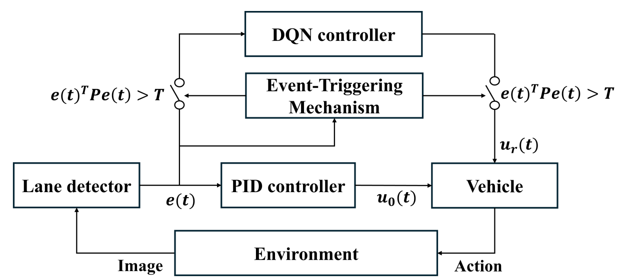

Fig. 2. Proposed event-triggered control scheme ($e(t)$: control error, $u_0(t)$: PID control input, $u_r(t)$: DQN controller input, P: weighting constant, T: predefined threshold)

C. Deep Q-Network for enhancing Autonomous driving

To complement the baseline PID controller in complex driving scenarios such as sharp curves or rapid lateral deviations, we employ a DQN as an auxiliary controller. DQN is a value-based reinforcement learning algorithm that combines Q-learning with deep neural networks to approximate optimal action-value functions. In each time step, the agent selects an action based on the current state and updates the Q-values using the Bellman equation to maximize cumulative future rewards. Our DQN is trained to generate corrective steering actions that improve tracking accuracy when the lane-following error exceeds a predefined threshold. This setup allows the control system to retain the reliability and simplicity of PID in most situations, while leveraging the adaptability of reinforcement learning when needed.

The DQN observes a compact state vector consisting of the lateral lane deviation $e_x(t)$, longitudinal lane deviation $e_y(t)$, and the previous final hybrid steering command—that is, the sum of the PID controller's output and the previous DQN correction. The action space is discretized into nine steering correction levels, each representing a fixed incremental adjustment to the current PID output.

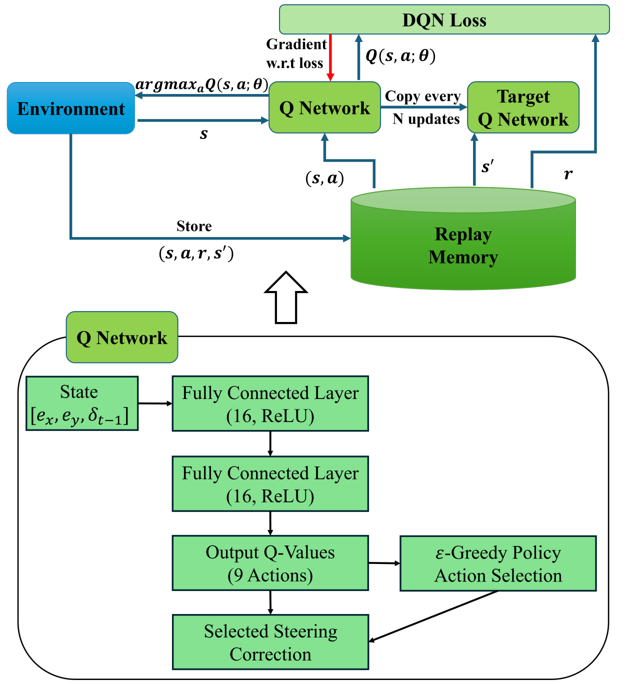

Fig. 3. Deep Q-Network for Event-triggered control for vision-based autonomous vehicles

The Q-network is implemented as a fully connected feed-forward neural network with two hidden layers of 16 neurons each, using ReLU activation functions. The network takes the 3-dimensional state as input and outputs Q-values for the 9 discrete actions. To ensure stable learning, we use a target network along with an experience replay buffer that stores up to 10000 transitions. During training, mini-batches of 64 randomly sampled transitions are used to update the Q-network. This helps reduce temporal correlation among samples and improves training robustness. An ε-greedy exploration strategy is applied, where ε decays exponentially from 1.0 to 0.05 over 1000 steps. This decay schedule allows sufficient early-stage exploration while gradually shifting toward exploitation as the agent gains more experience. The Q-network is trained online during simulation using the Adam optimizer with a learning rate of 0.001. Figure 3 depicts the Deep Q-Network used in the simulation.

One of the key contributions of our control design is the formulation of a novel reward function for training the DQN agent in autonomous driving. The reward at each time step is designed to penalize large tracking errors, sudden changes in steering, and excessive deviation beyond a predefined safety threshold. Formally, the reward R_t is defined as:

$$R(t) = -|e_x(t)| - \alpha \cdot |\delta(t) - \delta(t-1)| - \beta \cdot \max(0, |e_x(t)| - \gamma) \tag{3}$$

where $e_x(t)$ denotes the lateral lane deviation, and δ_t represents the final hybrid steering command resulting from the combination of the PID output and the DQN correction at time step t. The first term penalizes raw lane deviation. The second term, weighted by α, penalizes abrupt changes in steering to promote smooth control. The third term, weighted by β, applies an additional penalty only when the deviation exceeds a predefined threshold γ, encouraging strong correction only when necessary. This reward formulation balances accuracy and smoothness under normal conditions while enabling aggressive adjustments in critical scenarios.

When the lane deviation remains below the defined threshold, the PID controller operates independently. Once the deviation surpasses this threshold, the DQN is activated and its output is added to the PID control signal, forming a combined steering command. This integration ensures that the DQN contributes only when necessary, minimizing computational overhead while enhancing control performance in challenging situations.

III. EXPERIMENTAL RESULTS

A. Simulation Environment

All experiments were conducted in the Udacity self-driving car simulator, a popular open-source platform for prototyping autonomous driving algorithms in a controlled yet visually realistic environment. The simulator features a variety of road types, including bridges, lane-marked roads, and unmarked segments, as well as diverse curvature and road width conditions, allowing for a comprehensive assessment of control performance. The virtual vehicle is equipped with three front-facing RGB cameras—left, center, and right—mounted on the front bumper. However, only the center camera was utilized for both training and deployment of the control algorithm. Images were captured at a resolution of 320×160 pixels at an approximate frame rate of 25 frames per second. The control loop operated at 10 Hz to emulate real-time driving conditions. Experimental evaluation focused on two major types of road segments: straight lanes and sharp curves. The latter were used to assess the hybrid controller's adaptability in complex and nonlinear driving scenarios. All experiments were conducted on a consistent track under fixed environmental conditions to ensure fair comparison across different control schemes.

B. Autonomous Driving Simulation

The effectiveness of the proposed method is evaluated through a comparison with a baseline that employs only conventional PID control.

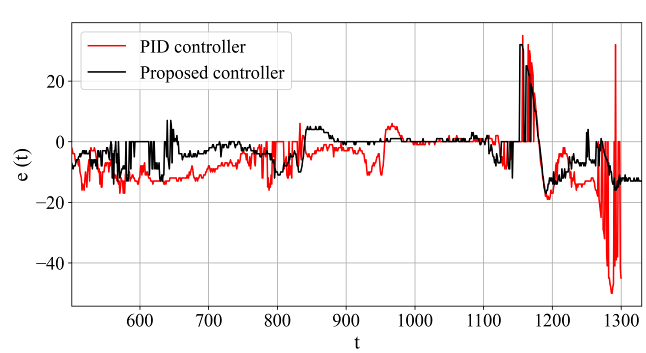

Fig. 4. Comparison of lane tracking errors

Fig. 4 presents a comparison between the lane tracking errors of the PID controller and the proposed method. The red solid line represents the error associated with the PID controller, whereas the black solid line corresponds to the proposed controller. After time step 1100, the vehicle enters a sharp turning section. While the proposed method continues to reduce control errors in a stable manner, the PID controller causes the vehicle to deviate from the road and come to a stop near time step 1300. These results indicate that the proposed controller provides auxiliary inputs to the PID controller, thereby improving the overall safety of the driving system.

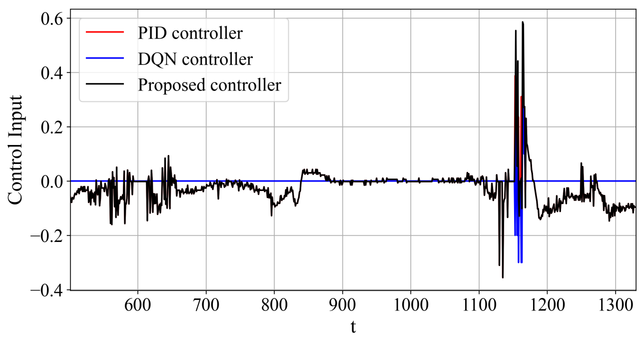

Fig. 5. Control Input (Proposed controller=PID controller+DQN controller)

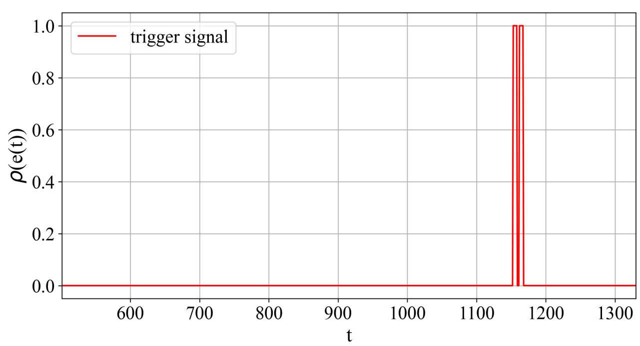

Fig. 6. Trigger signal in the simulation

Figs. 5 and 6 show the control input and the event-triggering signal, respectively. In Fig. 5, the proposed controller represents the combined control input of the PID controller and the DQN controller. As shown in Figs. 5 and 6, the DQN controller generates control inputs only when a trigger signal is activated. This confirms that the proposed approach significantly reduces computational complexity. Consequently, the DQN controller contributes to enhanced safety and demonstrates its practical efficiency.

IV. CONCLUSIONS

In this study, we presented an event-triggered hybrid control framework that combines a PID controller with a DQN to enhance lane-keeping safety in autonomous driving. The DQN was selectively activated based on a lane deviation threshold, while a custom reward function encouraged smooth and accurate control. This approach reduced unnecessary computations and demonstrated improved stability and adaptability over the baseline PID controller in simulation experiments.

To further improve the proposed framework, we plan to explore Proximal Policy Optimization (PPO) as an alternative to DQN [16]. Although PPO typically requires longer training times, it offers notable advantages, including improved training stability via a clipped surrogate objective and the ability to handle continuous action spaces. These properties have the potential to enable more precise and adaptive steering in future implementations.

In addition, while a simplified form of IBVS was adopted in this work, future work will focus on utilizing full geometric information by incorporating all four corner points of the detected lane boundaries. This improvement will enable more accurate estimation of lane deviation, potentially leading to further gains in control precision and robustness.

Finally, we will analyze the stability of the proposed DQN-based event-triggered control system using the Lyapunov stability method.

REFERENCES

[1] A. G. Cunningham, E. Galceran, R. M. Eustice, and E. Olson, "MPDM: Multipolicy decision-making in dynamic, uncertain environments for autonomous driving," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1670–1677.

[2] R. Marino, S. Scalzi, and M. Netto, "Nested pid steering control for lane keeping in autonomous vehicles," *Control Engineering Practice*, vol. 19, no. 12, pp. 1459–1467, 2011.

[3] N. Wang, Y. Gao, H. Zhao, and C. K. Ahn, "Reinforcement learning-based optimal tracking control of an unknown unmanned surface vehicle," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 7, pp. 3034–3045, 2020.

[4] X. Dai, C.-K. Li, and A. B. Rad, "An approach to tune fuzzy controllers based on reinforcement learning for autonomous vehicle control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 3, pp. 285–293, 2005.

[5] S.-Y. Oh, J.-H. Lee, and D.-H. Choi, "A new reinforcement learning vehicle control architecture for vision-based road following," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 3, pp. 997–1005, 2000.

[6] A. Folkers, M. Rick, and C. Büskens, "Controlling an autonomous vehicle with deep reinforcement learning," in *2019 IEEE intelligent vehicles symposium (IV)*. IEEE, 2019, pp. 2025–2031.

[7] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband *et al.*, "Deep q-learning from demonstrations," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[8] Z. Lin, T. Zhao, G. Yang, and L. Zhang, "Episodic memory deep q-networks," *arXiv preprint arXiv:1805.07603*, 2018.

[9] W. P. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *2012 ieee 51st ieee conference on decision and control (cdc)*. IEEE, 2012, pp. 3270–3285.

[10] A. Girard, "Dynamic triggering mechanisms for event-triggered control," *IEEE Transactions on Automatic Control*, vol. 60, no. 7, pp. 1992–1997, 2014.

[11] X.-M. Zhang, Q.-L. Han, and B.-L. Zhang, "An overview and deep investigation on sampled-data-based event-triggered control and filtering for networked systems," *IEEE Transactions on industrial informatics*, vol. 13, no. 1, pp. 4–16, 2016.

[12] L. Liu, X. Li, Y.-J. Liu, and S. Tong, "Neural network based adaptive event trigger control for a class of electromagnetic suspension systems," *Control Engineering Practice*, vol. 106, p. 104675, 2021.

[13] M. Mazo and P. Tabuada, "On event-triggered and self-triggered control over sensor/actuator networks," in *2008 47th IEEE conference on decision and control*. IEEE, 2008, pp. 435–440.

[14] R. Khanam and M. Hussain, "Yolov11: An overview of the key architectural enhancements," *arXiv preprint arXiv:2410.17725*, 2024.

[15] A. Hajiloo, M. Keshmiri, W.-F. Xie, and T.-T. Wang, "Robust online model predictive control for a constrained image-based visual servoing," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 4, pp. 2242–2250, 2015.

[16] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.