Journal of Institute of Control, Robotics and Systems (2025) 31(2):114-123
https://doi.org/10.5302/J.ICROS.2025.24.0263 ISSN:1976-5622 eISSN:2233-4335

고속도로 주행 판단을 위한 CDR (커리큘럼 기반 도메인 랜덤화)을 사용한 강화학습 방법

Reinforcement Learning for Highway Driving Decision-making Using CDR (Curriculum based Domain Randomization)

용 재 형, 김 민 성, 박 태 형*
(Jae-Hyoung Yong[1], Min-sung Kim[2], and Tae-Hyoung Park[3,*])
[1]Department of Intelligent System and Robotics, Chungbuk National University
[2]Department of Control Robot Engineering, Chungbuk National University
[3]Department of Intelligent System and Robotics Chungbuk National University and Chungbuk National University Hospital

Abstract: Incorrect autonomous driving decisions on highways can lead to traffic congestion and accidents. Therefore, accurate decision-making in highways is essential. However, decision-making in highways is a challenging task due to the complexity of traffic situations. Thus, reinforcement learning is an appropriate approach for decision-making in highways. However, a domain gap often arises between the training and testing environments due to differences in them. Therefore, a curriculum-based domain randomization approach is applied, which progressively adjusts the level of random parameters. The RL model is trained using the SUMO simulator and tested in the MORAI simulator, where white noise is added to simulate real-world conditions. Consequently, applying curriculum domain randomization has the lowest collision rate.

Keywords: domain randomization, reinforcement learning, decision making, autonomous driving

I. 서론

자율주행에서의 판단은 라이다, 카메라, IMU, GPS등을 사용하여 환경을 인지한 정보를 바탕으로 판단한다. 판단은 규칙 기반의 방법(rule-based)과 학습 기반의 방법(learning-based)으로 수행할 수 있다. 규칙 기반 방법은 규칙 설계자가 설계하지 않은 상황을 마주했을 때는 큰 사고로 이어질 수 있다는 단점이 있다. 고속도로에서는 예상을 벗어난 수많은 주행 상황이 존재하므로 규칙 기반 방법으로 고속도로 주행 판단을 하기에는 적절하지 않다.

학습 기반 방법의 대표적인 방법은 강화학습이다. 강화학습은 정해진 학습을 수행하는 것이 아닌, 에이전트가 탐험(exploration)과 활용(exploitation)을 반복하며 가장 최적의 행동을 탐색하는 방법이다. 수만 번의 탐험과 활용을 반복해 다양한 상황에 대해 학습하기 때문에 규칙 기반 방법과 달리 고속도로와 같은 다양한 주행 판단 상황에 사용하기 적합하다. 이에 고속도로 주행 판단은 강화학습을 이용한 방법이 선호되고 관련 연구들이 많이 진행되고 있다[1-3].

강화학습은 수만 번의 시도를 통해 학습해야 하므로 시뮬레이터를 이용해 학습하게 된다. 그러나 그림 1과 같이

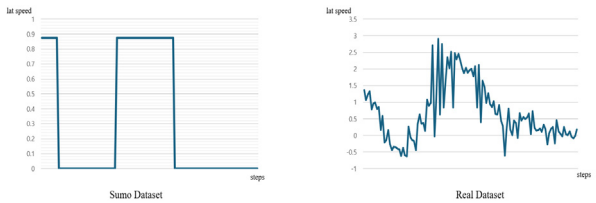


그림 1. 시뮬레이션 데이터(SUMO, 좌)와 실제 데이터(우).
Fig. 1. Simulation data (SUMO, Left) and real data (Right).

학습 환경인 소스 도메인과 모델을 실제로 적용하려는환경인 타겟 도메인 간의 차이가 생기는 문제가 있다. 이러한 차이를 도메인 갭(domain gap)이라고 한다. 도메인 갭이 발생하면 실제 환경과 학습 환경에서 서로 다른 결과를 내는 문제가 생긴다[4-6]. 이러한 문제는 자율주행 상황에서 큰 사고를 야기할 수 있다.

도메인 갭을 줄이는 방법으로는 도메인 적응[7]과 도메인 랜덤화[8]방법이 있다. 도메인 적응은 소스 도메인에 타겟 도메인의 데이터를 적응시키며 학습하는 방법이다. 고속도로와 같은 환경은 충돌 직전의 타겟 도메인 데이터 취득

*Corresponding Author
Manuscript received November 5, 2024; revised December 5, 2024; accepted December 26, 2024
용재형: 충북대학교 지능로봇공학과 대학원생(2023298007@chungbuk.ac.kr, ORCID 0009-0005-1400-6539)
김민성: 충북대학교 제어로봇공학과 대학원생(kim_min_sung@naver.com,ORCID 0000-0001-8786-7077)
박태형: 충북대학교 지능로봇공학과 및 충북대학교 병원 교수(taehpark@cbnu.ac.kr, ORCID 0000-0002-3695-344x)
※ 본 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 지역지능화혁신인재양성사업임(IITP-2025-RS-2020-II201462).

이 어려워 도메인 적응을 통해 도메인 갭 문제를 해결하기에는 적절하지 않다.

반면, 도메인 랜덤화 방법은 타겟 도메인 데이터를 사용하지 않고 소스 도메인 데이터에 변형을 주어 도메인 갭을 해결하는 방법이다. 다만, 학습 데이터인 소스 도메인 데이터에 변형을 주어 학습을 하다 보니 성능이 떨어지는 문제가 발생한다.

이를 해결하고자 본 논문에서는 소스 도메인의 데이터에 변형을 점진적으로 주는 방법인 커리큘럼 기반 도메인 랜덤화(curriculum domain randomization, CDR) 방법을 사용한다.

로봇공학 및 매니퓰레이터 분야에서 커리큘럼 기반 도메인 랜덤화를 사용한 연구[9]는 진행중이나 자율 주행에 적용한 연구는 없다. 이에 본 논문에서는 고속도로 주행 판단에 커리큘럼 기반 도메인 랜덤화 방법을 적용해 강화학습을 진행하고, 학습 시뮬레이터(SUMO)와 검증 시뮬레이터(MORAI)의 차이를 두어 이를 검증한다.

II. 기존 연구

1. 도메인 랜덤화

도메인 갭을 줄이는 방법인 도메인 랜덤화는 그림 2 (A)와 같이 소스 도메인을 변경하여 학습하는 방법으로 랜덤 파라미터가 어느 특정 간격 내에 위치하고 각 파라미터를 해당 범위 내에서 균등하게 샘플링하는 방법이다. 랜덤 파라미터는 소스 도메인에서 무작위로 조정되는 변수로 동역학, 다양한 유형의 센서 노이즈 등이 변형된 형태로 나타낸다. 따라서 도메인 랜덤화 방법을 사용하면 에이전트는 랜덤 파라미터를 통해 다양한 환경에서 학습할 수 있다. 그 결과 랜덤 파라미터로 학습된 모델은 학습 환경인 소스 도메인이 아닌 여러 다양한 타겟 도메인에서도 강인하게 작동한다.

Tobin et al.는 이미지 기반의 도메인 랜덤화를 강화학습에 처음 제안하였고[8], 실제 이미지를 사용하지 않고 훈련 이미지의 텍스쳐, 조명, 카메라의 위치와 같은 요소를 랜덤 파라미터로 설정해 학습하였다. 그 결과 모델이 실제 이미지를 만났을 때도 효과적으로 작동할 수 있게 하였다.

Kontes et al.는 자율주행에서의 도메인 랜덤화를 적용한 연구이다[10]. Kontes et al.는 차량의 속도, 장애물과의 상대거리를 랜덤 파라미터로 설정하였다. 이후 CARLA 시뮬레이터에서 학습을 진행하고, CARLA 시뮬레이터의 출력 값에 변형을 줘 환경을 다르게 하여 검증하였다.

Niu et al.도 마찬가지로 자율주행에서 도메인 랜덤화를 적용한 연구를 진행하였고 SUMO 시뮬레이터에서 학습을 진행하고, SUMO 시뮬레이터의 출력 값의 변형을 줘 환경을 다르게 하여 검증하였다[11].

2. 커리큘럼 도메인 랜덤화

기존 도메인 랜덤화는 랜덤 파라미터를 무작위로 샘플링하여 학습 과정 중 같은 상황과 환경에 대해 다른 행동을 선택해 학습 효율성이 떨어지는 문제가 있었다. 이를 해결하기 위해 능동적 도메인 랜덤화(active domain randomization, ADR) [12]이 연구되었다.

ADR은 에이전트의 학습 상황에 따라 학습의 환경 난이도를 무작위로 점차 증가시키는 방식이다. ADR은 확률적

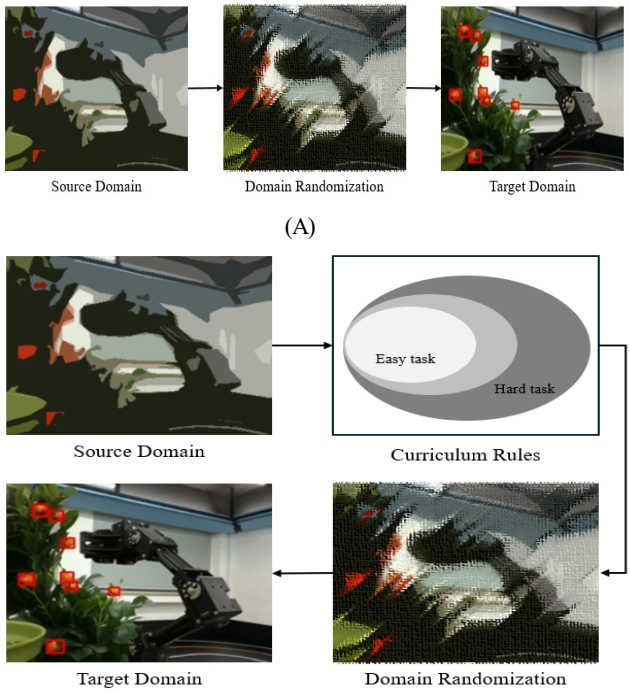


그림 2. 기존 도메인 랜덤화 (A), 기존 커리큘럼 도메인 랜덤화 (B).
Fig. 2. Domain randomization (A), Curriculum domain randomization (B).

가치 기울기 정책(stochastic value gradient policy, SVPG)기반[13]으로 난이도를 증가 시키기 때문에 난이도 조절이비구조적이고, 체계적인 순서가 없어 적당한 난이도 조절이 이루어지지 않는 단점이 있다.

이러한 문제를 해결하고자 커리큘럼 도메인 랜덤화(curriculum domain randomization, CDR) [14] 방법이 연구되었다. CDR은 그림 2 (B)와 같이 단계적 커리큘럼을 통해 학습 난이도 조절을 한다. 따라서 에이전트는 쉬운 난이도에서 시작해서 어려운 난이도 까지 순차적으로 학습해 안정적인 성능 향상을 이룬다. Aflakian et al.은 마찰 계수와 강성을 점진적으로 학습시켜 매니퓰레이터에서의 정교한 업무수행을 가능하게 한다[9]. 자율주행에 CDR을 적용한 연구는 진행 중이지 않아 본 논문에서는 이를 적용한다.

III. 주행 판단을 위한 커리큘럼 도메인 랜덤화 적용

1. 고속도로 주행 판단 문제 정의

고속도로는 먼 거리를 편리하게 이동할 수 있는 목적으로 설계된 도로로 고속 주행이 가능하다. 또한, 교통 신호가 없고 교통 흐름에 방해되지 않도록 차선 유지 및 추월하며 주행하는 도로이다. 따라서 먼 거리를 효율적으로 주행하기 위해서 빠른 속도로 주행해야 하며 상대 차량이 앞을 가로막고 있을 경우 교통 흐름을 위해 추월하면서 충돌하지 않고 주행해야 한다.

따라서 고속도로 주행 판단은 다음과 같이 정의한다.

- 주변에 차량이 없으면 최고 속도로 주행한다.
- 상대 차량이 자차와 같은 차선이고 앞을 가로막고 있으면 차선을 변경하여 주행한다.

- 상대 차량이 다른 차선에서 끼어들어 주행하면 속도를 줄이거나 차선을 변경하여 주행한다.

2. 강화학습 환경 정의

강화학습 환경 모델링은 마르코프 결정과정(Markov decision process, MDP)를 따른다. MDP는 (S, A, P, R, γ)로 구성된 튜플 형태로 이루어져 있다. 튜플 형태란 원소의 순서가 있는 집합 형태이다. 표 1은 강화학습 환경 정의 및 커리큘럼 도메인 랜덤화 적용을 위한 기호 정의이다.

2.1 상태 공간(State Space)

매 타임 스텝마다 시뮬레이터 환경에서 나오는 상태 공간 S는 식 (1)과 같다.

$$S = \left(\boldsymbol{z}_0, \cdots, \boldsymbol{z}_{n_{veh}}\right) \tag{1}$$

i 번째 차량의 상태 벡터 z_i는 식 (2)와 같다. 도로의 곡률과 관계없이 차량의 위치를 일관적으로 나타낼 수 있고, 계산의 편리성을 위해 전역 좌표 (x, y)대신 프레네(frenet) 좌표계로 변환한 (s,d)를 사용한다[15].

$$\boldsymbol{z}_i = [s_i, d_i, v_{s,i}, v_{d,i}]^T \tag{2}$$

2.2 행동 공간(Action Space)

행동 a는 표 2와 같이 총 9개의 행동으로 구성된다. 행동은 종 방향 행동과 횡 방향 행동의 결합한 형태로 이루어져 있고, 가속과 감속은 각각 에이전트에 1 m/s^2, -3 m/s^2을 취한다.

2.3 보상 함수(Reward Function)

에이전트의 최적 행동은 현재 상태에서 어떠한 행동을 하였을 때, 현재 보상 값과 미래 보상 값의 기댓값의 합이 가장 높은 행동으로 결정된다. 따라서 보상 모델에 따라, 강화학습의 목표가 결정된다. t 시점에서 보상 함수는 식 (3)과 같다.

$$r_t = \frac{v_{s,0}}{v_{\max,0}} \tag{3}$$

자차의 속도가 높을수록 에이전트가 높은 보상 값을 획득할 수 있도록 설계한다. 또한, 충돌 방지를 위해 충돌하였을 때 -10의 페널티를 부과하고, 무분별한 차선 변경을 방지하기 위해 차선을 변경할 때 -1의 페널티를 부과한다.

3. D3QN (Dueling Deep Q Network)

강화학습은 에이전트가 환경과 상호작용하며 최적의 정책(π^*)을 구하는 과정이다. 상태와 행동을 각각 s , a라고 할 때, 최적의 정책(π^*)은 최적의 상태 행동 가치 함수 $Q^*(s,a)$을 통해 구한다. $Q^*(s,a)$는 현재의 보상 값과 미래 보상에 대한 기댓값의 합으로 식 (4)와 같다. 이때 보상 모델 R_t는 매 타임 스텝(t)에 대해 식 (5)을 따른다.

$$Q^*(s,a) = \max_{\pi} \mathbb{E}[R_t | s_t = s, a_t = a, \pi] \tag{4}$$

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \tag{5}$$

Q-learning은 Q 값을 식 (4)와 식 (5)를 이용하여 직접 계산하여 구한다. 그러나 상태 공간과 행동 공간의 크기가 커질수록 계산하여 구하기는 쉽지 않다. 이에 DQN (Deep Q Network) [16]에서는 신경망을 이용하여 Q 값을 추정한다. 신경망의 가중치를 θ라고 할 때, $Q(s,a) \approx Q(s,a;\theta)$로 근사할 수 있다. 가중치($\theta$)는 식 (6)의 손실 함수를 이용하여 최적값을 구한다.

$$L(\theta) = \mathbb{E}_M[(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s,a;\theta))^2] \tag{6}$$

DQN은 행동을 학습할 때와 평가할 때 같은 네트워크를 사용하기 때문에 Q 값이 크게 추정되어 최적의 행동을 선택하지 못하는 문제가 있다. 이에 DDQN (Double Deep Q Network) [17]을 통해 해결하였다. DDQN은 식 (7)와 같이 행동을 학습할 때와 평가할 때 서로 다른 네트워크를 사용하여 Q 값을 DQN보다 정확하게 추정하였다.

$$L(\theta) = \mathbb{E}_M[(r + \gamma Q(s', \max_{a'} Q(s', a'; \theta); \theta^-) - Q(s,a;\theta))^2] \tag{7}$$

한편 DQN에서의 Q 값의 성능을 개선시키기 위해 Dueling DQN [18]이 개발되었다. Dueling DQN은 식 (8)과 같이 상태 가치 함수 $V^*(s,a)$와 행동 가치 함수 $A^*(s,a)$로 나누어 최적의 Q 값을 구한다.

표 1. 기호 정의.

Table 1. Symbols definition.

기호	설명
s_i, d_i	i 번째 차량의 주행 거리, 횡 방향 거리
$v_{s,i}, v_{d,i}$	i 번째 차량의 주행 방향 속도, 횡 방향 속도
n_{veh}	도로 위 차량 대수
ξ	네트워크 입력
$v_{\max,i}$	i 번째 차량 최대 속력
$r_{\max}$	센서의 최대 관측 범위
$d_{\max}$	도로의 횡 방향 최대 길이
k_{sensor}	센서의 신호 감쇠 계수
θ_{res}	센서의 분해능 값
epi	현재 에피소드 수
$epi_{\max}$	최대 에피소드 수
θ , θ^-	신경망의 가중치, 타겟 신경망의 가중치
S, z_i	상태 공간, i 번째 차량 상태 벡터
A, a	행동 공간, 행동
R	보상 함수
γ	할인 인자

표 2. 행동 a.

Table 2. Action a.

행동	설명	
	종 방향 행동	횡 방향 행동
a_0	유지	유지
a_1	가속	유지
a_2	감속	유지
a_3	유지	좌측 차선 변경
a_4	가속	좌측 차선 변경
a_5	감속	좌측 차선 변경
a_6	유지	우측 차선 변경
a_7	가속	우측 차선 변경
a_8	감속	우측 차선 변경

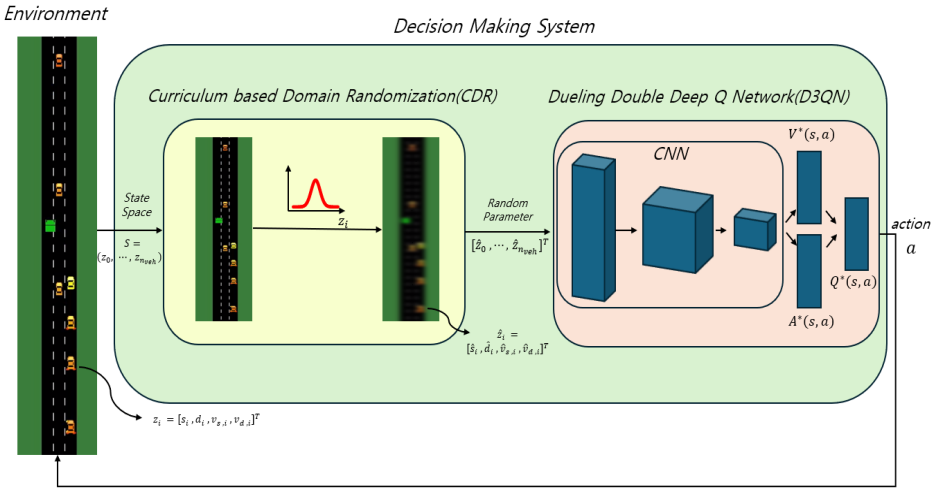


그림 3. 커리큘럼 도메인 랜덤화를 사용한 주행 판단 구성도.

Fig. 3. Driving decision-making framework using CDR.

$$Q^*(s,a) = V^*(s,a) + A^*(s,a) \tag{8}$$

D3QN [19]은 Dueling DQN과 DDQN의 장점을 합쳐 식 (7)과 식 (8)을 결합한 방법이다.

4. 커리큘럼 도메인 랜덤화 적용

도메인 갭을 줄이는 고속도로 주행 판단을 하기 위해서 그림 3과 같이 커리큘럼 도메인 랜덤화 방법을 적용하고 D3QN을 이용해 학습하였다.

랜덤화 범위는 랜덤 파라미터가 가질 수 있는 값의 상한과 하한을 범위로 나타낸 것을 말하며, 균등 분포(uniform distribution)나 정규 분포(normal distribution)를 따른다. 상태 공간에 있는 값들이 차량의 상태 값이기 때문에 랜덤화 방식은 정규 분포로 적용한다. 본 논문에서는 상대 차량의 거리와 에피소드를 기준으로 랜덤화 범위를 점진적으로 조절할 수 커리큘럼을 생성해 적용한다.

객체가 멀리 있을수록 관측이 어렵고 노이즈가 많이 생긴다는 특성[20]으로 인해 그림 4와 같이 상대 차량이 멀리 있을 때 랜덤 파라미터의 분산 값을 크게 하고, 상대 차량이 가까이 있을 때 랜덤 파라미터의 분산의 값을 작게 하여 거리에 따라 랜덤 파라미터의 값을 다르게 설정한다. 따라서 멀리 있는 차량에 대해서는 랜덤화 범위가 큰 랜덤 파라미터를 가지게 되고, 가까이 있는 차량에 대해서는 랜덤화 범위가 작은 랜덤 파라미터를 가지게 된다. 식 (9)는 i 번째 차량에 대한 랜덤 파라미터 분산 값이다.

$$\sigma_{sensor,i} = \frac{s_i - s_0}{r_{\max}} k_{sensor} \tag{9}$$

또한, LiDAR 센서를 사용했다고 가정하면 센서의 분해능 특성에 따라 k_{sensor}는 식 (10)과 같이 정의한다.

$$k_{sensor} = \tan(\theta_{res}) \tag{10}$$

또한, 학습 경험에 따라서 에이전트의 학습 경험이 적을 때에는 랜덤화 범위를 작게 하고, 학습 경험이 쌓였을 때는 랜덤화 범위를 크게 한다. 학습 초반에는 랜덤화 범위가 작으므로 랜덤 파라미터가 최적의 선택을 하는데 미치는 영향이 미비해 학습의 안정성을 해치지 않는다. 학습 중후반에는 학습된 가중치로 인해 랜덤화 범위가 크더라도 학습의 안정성을 해치지 않는다.

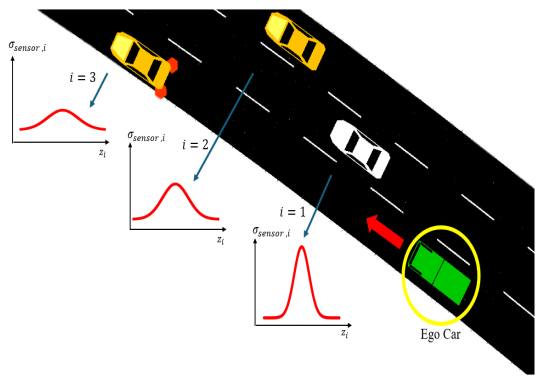


그림 4. 상대 차량의 거리에 따른 랜덤화 범위.

Fig. 4. Randomization range based on the relative distance of surrounding vehicle.

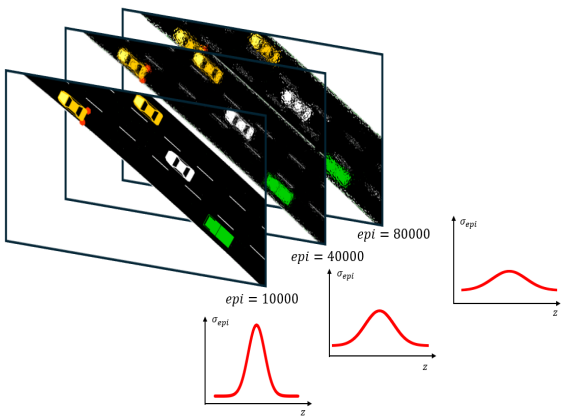


그림 5. 에피소드에 따른 랜덤화 범위.

Fig. 5. Randomization range based on episodes.

따라서 그림 5와 같이 에피소드의 수에 따라 에피소드 수가 작으면 랜덤 파라미터의 분산 값을 작게 하고, 에피소드의 수가 크면 랜덤 파라미터의 분산 값을 크게 한다. 즉 에피소드가 커질수록 랜덤화 범위가 커지는 것을 의미한다. 식 (11)은 t 시점의 에피소드가 epi일 때의 랜덤 파라미터의 분산 값이다.

$$\sigma_{epi} = \frac{epi}{epi_{\max}} \tag{11}$$

랜덤 파라미터의 최종 분산은 식 (9), 식 (11)의 곱으로 식 (12)와 같다. 랜덤 파라미터는 랜덤화 방식은 정규 분포를 따르므로 식 (13)와 같이 나타낸다. 식 (13)은 i 번째 차량에 대한 랜덤 파라미터이다.

$$\sigma_i = \sigma_{sensor,i}\,\sigma_{epi} \tag{12}$$

$$\hat{z}_i \sim N(z_i, \sigma_i^2) \tag{13}$$

D3QN으로 학습하는 네트워크 입력(ξ)은 식 (13)에서 정의한 $\hat{z}_i$를 통해 구할 수 있으며

$$\xi = \left\{\varphi_1, \varphi_2, \varphi_3, \varphi_4, \cdots, \varphi_{4n_{veh}+1}, \varphi_{4n_{veh}+2}, \varphi_{4n_{veh}+3}, \varphi_{4n_{veh}+4}\right\}$$

로 구성된다. 본 네트워크는 고속도로 주행 판단에 대한 네트워크이기 때문에 자차의 차선과 속도, 차선 변경 상태가 고려되고 상대 차량의 위치 및 속도 차선 변경 상태가 고려되어 네트워크 입력으로 구성된다. 자세한 수식과 설명은 표 3과 같고, 신경망 계산의 편리성을 위해 $\xi \in [-1, 1]$로 정규화한다.

따라서 커리큘럼 도메인 랜덤화와 D3QN을 이용해 고속도로 주행 판단의 적절한 행동을 선택하는 알고리즘은 표 4와 같다.

표 3. 네트워크 입력(ξ).
Table 3. Network input (ξ).

설명	수식
자차 차선	$\varphi_1 = \frac{2d_0}{d_{\max}} - 1$
자차 속도	$\varphi_2 = \frac{2v_{s,0}}{v_{\max,0}} - 1$
자차 차선 변경 상태	$\varphi_3 = \mathrm{sgn}(v_{d,0})$
에피소드 종료 상태	$\varphi_4 = T \mathrm{or} F$
i번째 차량과 종방향 상대 거리	$\varphi_{4i+1} = \frac{s_i - s_0}{r_{\max}}$
i번째 차량과 횡방향 상대 거리	$\varphi_{4i+2} = \frac{d_i - d_0}{d_{\max}}$
i번째 차량과 상대 속도	$\varphi_{4i+3} = \frac{v_{s,i} - v_{s,0}}{v_{\max,0}}$
i번째 차량 차선 변경 상태	$\varphi_{4i+4} = \mathrm{sgn}(v_{d,i})$

표 4. 고속도로 주행 판단 알고리즘.
Table 4. Highway decision-making algorithm.

Algorithm 1 : Highway Decision-Making Algorithm

Input: All vehicle state initialization values
$\left\{x_0, y_0, v_{x,0}, v_{y,0}, \cdots, x_{n_{veh}}, y_{n_{veh}}, v_{x,n_{veh}}, v_{y,n_{veh}}\right\}$

while $epi < epi_{\max}$
  while $step < step_{\max}$
    for $i \leftarrow 0$ to n_{veh}

$s_i, d_i, v_{s,i}, v_{d,i} \leftarrow FRENET(x_i, y_i, v_{x,i}, v_{y,i})$
$z_i \leftarrow \left\{s_i, d_i, v_{s,i}, v_{d,i}\right\}$
    end
    for $i \leftarrow 1$ to n_{veh}
$\sigma_{sensor,i} = \frac{s_i - s_0}{r_{\max}} \tan(\theta_{res})$
$\sigma_{epi} = \frac{epi}{epi_{\max}}$
$\sigma_i = \sigma_{sensor,i}\sigma_{epi}$
$\hat{z}_i \leftarrow N(z_i, \sigma_i)$
    end
$\xi \leftarrow \neq TWORKSETUP(\hat{z})$
$a \leftarrow D3QN(\xi)$
$x', y', v_x', v_y' \leftarrow ENVIRONMENT(a)$
  if collision **then**
    break
  else
$x, y, v_x, v_y \leftarrow x', y', v_x', v_y'$
  end
end

V. 시뮬레이션 및 결과

1. 학습 환경

본 논문은 저 사양 시뮬레이터인 SUMO (Simulation of Urban MObility) 시뮬레이터에서 지능형 자동차 부품 진흥원 주행 시험로의 도로를 HDMap (High Definition Map)기반으로 자체 제작해 학습하였다. SUMO 시뮬레이터는 차량의 횡 방향 이동이 조향각으로 움직이는 것이 아닌 차선 번호로 움직이기 때문에 매우 확장성이 높고, 다양한 시나리오 상황을 연출할 수 있기 때문에 강화학습으로 사용하기 좋다. 상대 차량의 주행 모델은 고속도로 환경이기 때문에 IDM [21] 모델을 사용하였다.

실험에 사용한 학습 파라미터는 표 5와 같고 실험 노이즈는 학습할 때 사용한 랜덤 파라미터 값과는 다르게 균등 분포 하여 적용하였다. 시나리오의 자세한 내용은 표 6과 같다. 시나리오 1, 2 에서는 도메인 랜덤화 방법을 적용하지 않은 D3QN (Only-D3QN) [22], 기존 도메인 랜덤화를 적용한 D3QN (DR-D3QN) [10], 커리큘럼 기반 도메인 랜덤화를 적용한 D3QN (CDR-D3QN) 방법을 비교해 학습 자체의 성능이 어떤지 소스 도메인과 그와 유사한 환경에서 비교

하였다. 시나리오 3, 4, 5 ,6에서는 칼만 필터를 적용해 소스 도메인의 입력 값과 유사한 필터링 결과를 네트워크 입력으로 사용한 D3QN (filter-D3QN), Only-D3QN, CDR-D3QN을 비교하는 실험을 수행하였다.

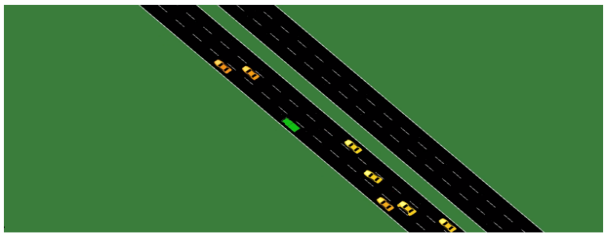
2. 실험 결과

그림 7은 최대 에피소드까지 학습을 수행하였을 때의 학습 그래프이다. Q 값과 보상 값이 클수록 성능이 좋다는 것을 의미한다. 학습 시에는 네트워크 입력에 아무 변동도 주지 않는 Only-D3QN 방법의 성능이 가장 뛰어나고 랜덤 파라미터를 적용한 DR-D3QN 방법의 성능이 가장 좋지 않았다. CDR-D3QN 방법은 Only-D3QN과 비슷한 성능을 보이다 랜덤화 범위가 커졌을 때 차이가 나는 것을 확인할 수 있다.

표 5. D3QN 학습 파라미터.

Table 5. The training hyperparameters for D3QN.

파라미터	값
Discount facctor	0.99
Learning rate	0.0005
Initial exploration constant	1
Final exploration constant	0.05
Max episode	80000
number of Vehicle	20
batch size	32
Number of Conv layer	2
Number of FC layer	2
Sensor Resolution	2



(A)

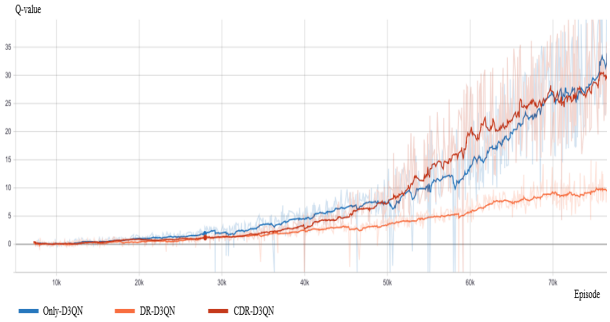
(B)

그림 6. SUMO 시뮬레이터 (A)와 MORAI 시뮬레이터 (B).

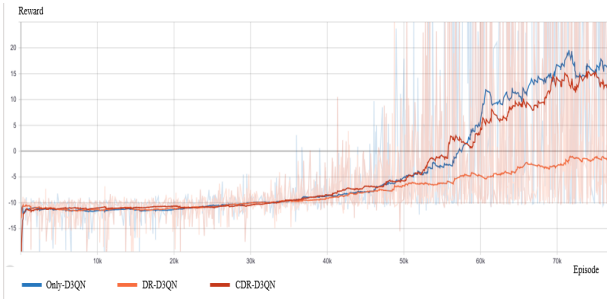
Fig. 6. SUMO simulator (A) and MORAI simulator (B).

또한 연산량도 Only-D3QN은 1 스텝 당 9ms, DR-D3QN과 CDR-D3QN은 10ms 소요한 것으로 도메인 랜덤화 방법이 연산량에 큰 영향을 미치지 않는 것을 확인하였다.

표 7은 학습 환경인 SUMO 시뮬레이터 상에서 실험한 결과이다. 35회 에피소드를 실행 후 그 값의 평균을 구해 충돌률과 평균 속도를 구하였다. 시나리오 1과 시나리오 2는 표 3과 같이 실험 노이즈 추가 여부에 따른 시나리오 구분이다. 시나리오 1에서는 Only-D3QN이 충돌률이 가장 낮고 평균 속도가 가장 높은 것을 확인할 수 있었는데 이는 학습 환경과 동일한 시나리오에서 검증하였기 때문이다.



(A)



(B)

그림 7. 에피소드에 따른 Q 값 (A), 보상 값 (B).

Fig. 7. Noise distribution by relative distance (A) and episode (B).

표 7. SUMO 시뮬레이터 실험 결과.

Table 7. Result of SUMO simulator.

	Scenario 1		Scenario 2	
	충돌률 (%)	평균 속도 (m/s)	충돌률 (%)	평균 속도 (m/s)
Only D3QN [22]	5.71	25.71	25.71	25.22
DR-D3QN [10]	17.14	14.28	14.28	15.52
CDR-D3QN	11.42	24.61	17.14	24.01

표 6. 실험 시나리오.

Table 6. Experiment scenario.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6
시뮬레이터	SUMO	SUMO	MORAI	MORAI	MORAI	MORAI
실험 노이즈 범위	0	[-1.5 , 1.5]	0	[-1.5 , 1.5]	[-2.0 , 2.0]	[-2.5 , 2.5]
상대 차량	20	20	50	50	50	50
상대 차량 속도 범위(m/s)	[0, 35]	[0, 35]	[0, 30]	[0, 30]	[0, 30]	[0, 30]

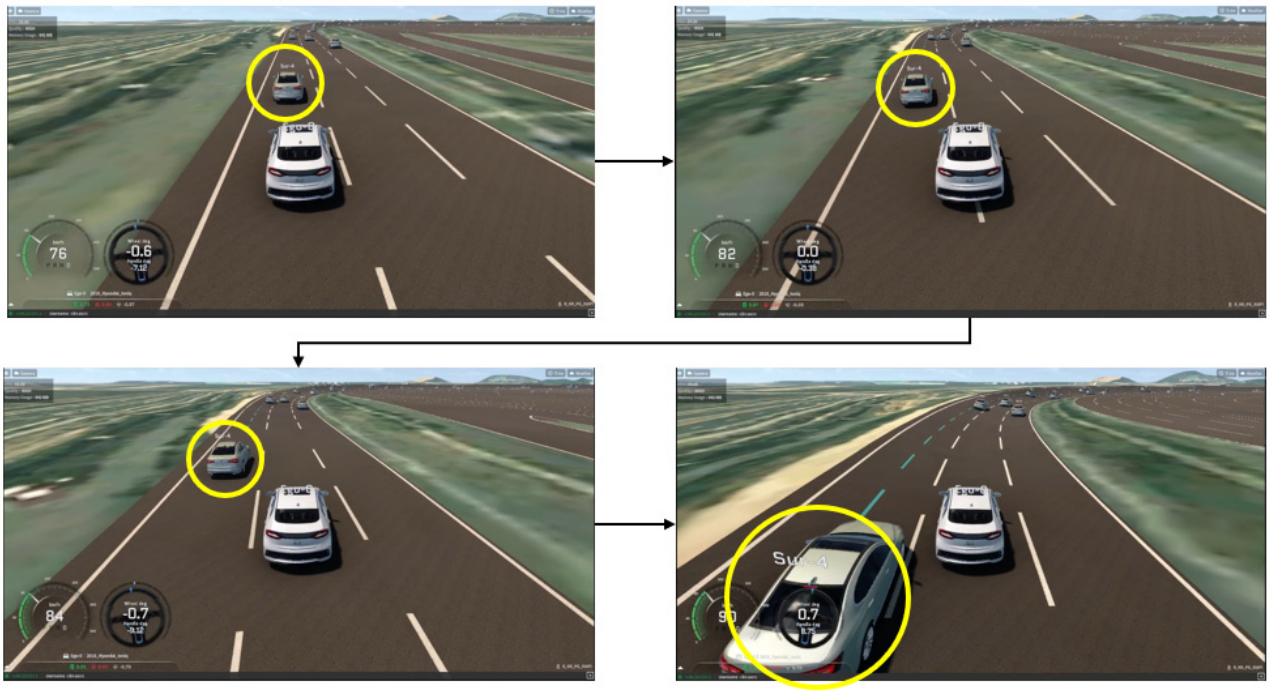


그림 8. 추월 시퀀스.

Fig. 8. Overtaking sequence.

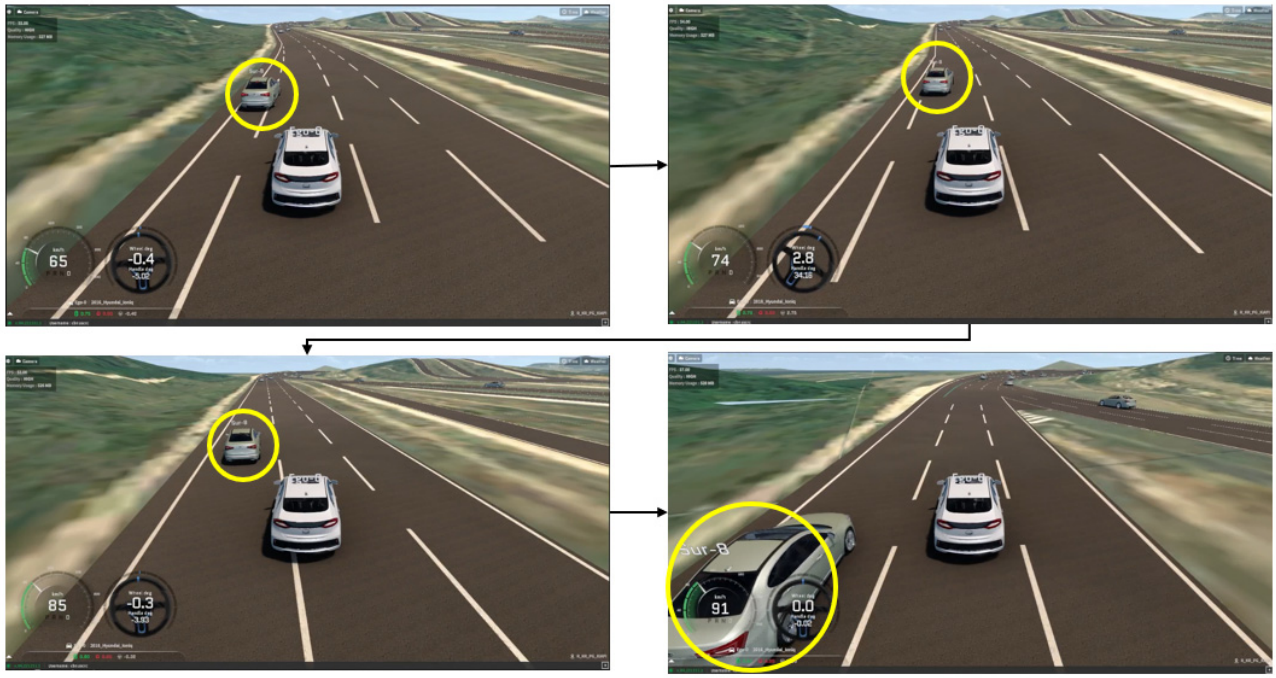


그림 9. 차량이 끼어드는 상황에 대한 추월 시퀀스.

Fig. 9. Overtaking sequence for a vehicle in a merging situation.

표 8. MORAI 시뮬레이터 실험 결과.

Table 8. Result of MORAI simulator.

	Scenario 3		Scenario 4		Scenario 5		Scenario 6	
	충돌률 (%)	평균 속도 (m/s)	충돌률 (%)	평균 속도 (m/s)	충돌률 (%)	평균 속도 (m/s)	충돌률 (%)	평균 속도 (m/s)
Only-D3QN [22]	20.00	19.87	30.00	20.72	36.66	20.08	26.66	19.62
filter-D3QN	6.66	21.19	13.33	20.90	23.33	20.92	13.33	20.03
CDR-D3QN	0.00	20.07	3.33	20.29	10.00	20.17	6.66	19.67

반면 실험 노이즈를 추가한 시나리오 2에서는 Only-D3QN의 충돌률이 가장 높게 나왔고 DR-D3QN의 충돌률이 가장 낮게 나왔다.

Only-D3QN의 충돌률이 가장 높은 이유는 노이즈가 추가된 다양한 상황에서의 학습이 부족해 충돌률이 가장 높았다. DR-D3QN의 충돌률이 가장 낮은 이유는 에이전트의 속도가 느려서, 상대 차들이 이미 다 에이전트 차량을 추월하는 결과들이 많았기 때문이다. CDR-D3QN은 속도도 고속도로 환경에 맞게 유지하고 충돌률 또한 Only-D3QN보다 낮은 것을 확인할 수 있었다.

또한 그림 10은 시나리오 1과 시나리오 2에서의 CDR-D3QN의 행동의 변화를 그래프로 나타낸 것으로 x축은 스텝 수 y축은 행동이다. 감속하는 행동은 a_2, a_5, a_8인데 그림 10에서 보면 같은 상황이지만 실험 노이즈가 추가된 시나리오 2에서는 안전한 주행을 위해 감속을 더 많이 하는 것을 확인할 수 있다.

표 8은 학습 환경과 다른 환경인 MORAI 시뮬레이터에서 에피소드 30회를 실험한 결과이다. MORAI 시뮬레이터는 SUMO 시뮬레이터와 다르게 시뮬레이터 입력 값을 차선 번호 값으로 주지 않기 때문에 각 차선에 대한 waypoint을 생성해 PurePursuit 횡 방향 제어기로 제어기를 구성해 차선 변경 및 속도 변경을 실시하였다. filter-D3QN의 칼만 필터 모델은 기구학적 모델을 적용해 실험을 진행하였다.

MORAI 시뮬레이터는 위치 정보가 GPS 데이터로 나오기 때문에 UTM 데이터로 변환하면 오차 및 노이즈가 발생한다. 시나리오에 따라 실험 노이즈를 다르게 추가하였는데 실험 노이즈를 추가하지 않은 시나리오 3의 충돌률이 가장 낮고, 실험 노이즈의 범위가 [-2.0, 2.0]인 시나리오 5의 충돌률이 전반적으로 가장 높았다. 실험 노이즈의 범위가 [-2.5, 2.5]인 시나리오 6의 충돌률보다 시나리오 5의 충돌률이 더 높은 이유는 시나리오 6의 경우 실험 노이즈를 과도하게 추가해 상대 차량이 도로에 벗어난 경우가 발생하였기 때문이다.

시나리오 3, 4, 5, 6 모두 Only-D3QN의 충돌률이 가장 높았다. CDR-D3QN은 시나리오 3, 4, 5, 6 모두 충돌률이 가장 낮았다. 이는 에이전트 차량이 주변 상대차량이 있을 때 안전거리를 더 유지하며 주행하려는 경향을 보였기 때문이다. filter-D3QN은 평균 속도와 충돌률이 CDR-D3QN 보다 약간 높았는데 이는 Only-D3QN의 네트워크 기반이었기 때문이다. Only-D3QN, filter-D3QN 모두 안전거리를 덜 유지하며 공격적으로 주행하려는 경향을 보였기 때문에 위와 같은 결과를 보였다.

그림 8, 9는 MORAI 시뮬레이터에서 CDR-D3QN의 추월 시퀀스이다. 그림 8, 9의 노란 원 안에 있는 차량은 상대 차량을 의미한다. 같은 차선에 상대 차량이 느린 속도로 주행하고 있으면 그림 8과 같이 추월 주행한다. 특히 그림 9와 같이 상대 차량이 갑자기 끼어드는 예상치 못한 상황에서도 CDR-D3QN은 강인한 성능을 보였다.

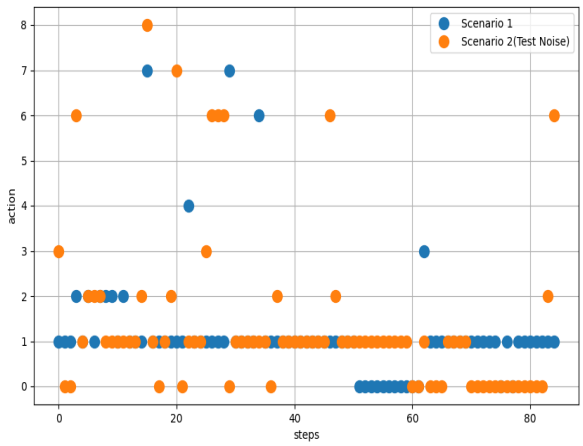


그림 10. 시나리오 별 행동 변화 그래프.

Fig. 10. Graph depicting action variation by scenario.

VI. 결론

본 논문은 고속도로 충돌 회피 상황에서 도메인 갭을 줄이고 효율적인 학습을 위해 랜덤 파라미터의 난이도를 점진적으로 조절하는 커리큘럼 도메인 랜덤화 방법을 적용하였다. 커리큘럼 도메인 랜덤화 방법은 로봇 공학 및 매니퓰레이터 분야에 대해서는 연구가 진행 중이나 자율주행에 적용한 연구는 진행되지 않았다.

실험은 도메인 갭이 생기는 환경을 MORAI 시뮬레이터 및 SUMO 시뮬레이터에 실험 노이즈를 추가해 단계별로 구성하였다. 기존 네트워크만 이용한 방법, 기존 도메인 랜덤화, 필터링 된 데이터를 기존 네트워크에 적용한 방법을 커리큘럼 도메인 랜덤화 방법과 비교하였다. 커리큘럼 도메인 랜덤화를 적용한 방법의 충돌률이 개선됨을 확인하였고 평균 속도 또한 비슷한 성능을 보임을 확인하였다.

그러나 본 논문은 거리에 따라서만 랜덤 파라미터를 다르게 설계하였기 때문에 실제 환경에서 다른 영향에 의해 발생하는 노이즈는 고려하지 않아 실제 차량으로 실험을 진행할 시 사고가 생길 위험성이 있다. 따라서 향후 여러 영향을 고려해 커리큘럼을 구성하여 연구, 실제 차량으로 실험할 계획이다.

REFERENCES

[1] J. Koo, K. Kim, and J. Jung, “Formation control and collision avoidance of multiple USVs based on multi-agent reinforcement learning,” *Journal of Institute of Control Robotics and Systems (in Korean),* vol. 30, no. 12, pp. 1398-1405, 2024.

[2] M.-S. Kim, G. Eoh, and T.-H. Park, “Decision making for self-driving vehicles in unexpected environments using efficient reinforcement learning methods,” *Electronics,* vol. 11, no. 11, pp. 1685, 2022.

[3] J. Park, D. Oh, and H. J. Kim “A survey on collision avoidance for multi-robot systems,” *Journal of Institute of Control, Robotics and Systems (in korean),* vol. 30, no. 4, pp. 402-411, 2024.

[4] Y. J. R. Chu, T. H. Wei, J. B. Huang, Y. H. Chen, and L.

Wu, “Sim-to-real transfer for miniature autonomous car racing,” arXiv preprint arXiv:2011.05617, 2020.

[5] M. U. Yavas, T. Kumbasar, and N. K. Ure, “A real-world reinforcement learning framework for safe and human-like tactical decision-making,“ *IEEE Transactions on Intelligent Transportation Systems,* vol. 24, no. 11, pp. 11773-11784, 2023.

[6] X. Hu, S. Li, T. Huang, B. Tang, R. Huai, and L. Chen, “How simulation helps autonomous driving: A survey of sim2real, digital twins, and parallel intelligence,” *IEEE Transactions on Intelligent Vehicles,* vol. 9, no. 1, pp. 593-612, 2023.

[7] Y. Zhang, B. Deng, H. Tang, L. Zhang, and K. Jia, “Unsupervised multi-class domain adaptation: Theory, algorithms, and practice,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 44, no. 5, pp. 2775-2792, 2020.

[8] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and O. Abbeel, “Domain randomization for transferring deep neural networks from simulation to the real world,” *IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vancouver, BC, Canada, pp. 23-30, 2017.

[9] A. Aflakian, J. Hathaway, R. Stolkin, and A. Rastegarpanah, “Robust contact-rich task learning with reinforcement learning and curriculum-based domain randomization,” *IEEE Access,* vol. 12, pp. 103461-103472, 2024.

[10] G. D. Kontes, D. D. Scherer, T. Nisslbeck, J. Fischer, and C. Mutschler, “High-speed collision avoidance using deep reinforcement learning and domain randomization for autonomous vehicles,” *IEEE 23rd International Conference on Intelligent Transportation Systems,* Rhodes, Greece, pp. 1-8, 2020.

[11] H. Niu, J. Hu, Z. Cui, and Y. Zhang, “Dr2l: Surfacing corner cases to robustify autonomous driving via domain randomization reinforcement learning,” *Proc. of the 5th International Conference on Computer Science and Application Engineering,* Sanya, China, pp. 1-8, 2021.

[12] B. Mehta, M. Diaz, F. Golemo, C. J. Pal, and L. Paull, “Active domain randomization,” *Proc. of the Conference on Robot Learning,* vol. 100, pp. 1162-1176, 2020.

[13] Y. Liu, P. Ramachandran, Q. Liu, and J. Peng, “Stein variational policy gradient,” arXiv preprint arXiv: 1704.02399, 2017.

[14] Raparthy. S. C, Mehta. B, Golemo. F, and Paull. L, “Generating automatic curricula via self-supervised active domain randomization,” *arXiv preprint arXiv: 2002.07911,* 2020

[15] M. S. Kim, J. H. Lee, T. L. Kim, and T. H. Park, “Frenet frame based local motion planning in racing environment,” *23rd International Conference on Control, Automation and Systems,* Yeosu, Korea, pp. 951-957, 2023.

[16] V. Mnih, “Playing atari with deep reinforcement learning,” *arXiv preprint arXiv:1312.5602,* 2013.

[17] H. Van Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double q-learning,” *Proc. of the AAAI Conference on Artificial Intelligence,* vol. 30, no. 1, 2016.

[18] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, “Dueling network architectures for deep reinforcement learning,” *Proc. of The 33rd International Conference on Machine Learning,* New York, USA, vol. 48, pp. 1995-2003, 2016.

[19] M. Sewak, “Deep q network (dqn), double dqn, and dueling dqn: A step towards general artificial intelligence,” *Deep Reinforcement Learning*, Springer, Singapore, pp. 95-108, 2019.

[20] P. Sun, W. Wang, Y. Chai, G. Elsayed, A. Bewley, X. Zhang, C. Sminchisescu, and D. Anguelov, “Rsn: Range sparse net for efficient, accurate lidar 3d object detection,” *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 5725-5734, 2021.

[21] A. Kusari, P. Li, H. Yang, N. Punshi, M. Rasulis, S. Bogard, and D. J. LeBlanc, “Enhancing SUMO simulator for simulation based testing and validation of autonomous vehicles,” *IEEE Intelligent Vehicles Symposium,* Aachen, Germany, pp. 829-835, 2022.

[22] C. J. Hoel, K. Wolff, and L. Laine, “Automated speed and lane change decision making using deep reinforcement learning,” *21st International Conference on Intelligent Transportation Systems,* Maui, Hl, USA, pp. 2148-2155, 2018.

용 재 형

2023년 충북대학교 전자공학부 졸업. 2023년 ~ 현재 충북대학교 지능로봇공학과 석사과정. 관심분야는 경로계획, 최적 제어, 강화학습.

김 민 성

2020년 충북대학교 기계공학부 졸업. 2022년 충북대학교 제어로봇공학과 석사. 2022년 ~ 현재 충북대학교 지능로봇공학과 박사과정. 관심분야는 경로 계획, 최적 제어, 강화학습.

박 태 형

1988년 서울대 제어계측공학과 졸업. 1990년 동 대학원 석사. 1994년 동 대학원 박사. 1994년~1997년 삼성테크윈(주) 선임연구원 1997년~현재 충북대학교 지능로봇공학과 교수. 2000년~2001년 토론토 대학교 방문교수. 2012년~2017년 대한전기학회 로보틱스 및 자동차 연구회 위원장. 2020년~현재 과기정통부 지정 Grand ICT 연구센터(산업인공지능 연구센터) 센터장. 관심분야는 자율주행, 로보틱스, 산업인공지능 등.