

# 장애물과 신호등을 고려한 디지털 가상환경 모델 구현 및 심층강화학습기반 다차선 자율주행 연구

이 재 영\*, 유 상 조°

## Implementation of Digital Virtual Environment Model Considering Obstacles and Traffic Lights, and Research on Multi-Lane Autonomous Driving Based on Deep Reinforcement Learning

Jae-yeong Lee\*, Sang-Jo Yoo°

### 요 약

본 논문에서는 심층강화학습을 활용한 디지털 가상 도로 환경에서의 자율주행 시스템을 제안한다. ML-unity 기 반으로 도로 상에서 다양한 장애물과 신호등이 배치된 다차선 도로 환경을 디지털 가상환경을 이용하여 구현하고, 자율주행 시스템 개발을 위해 차량에 현재 도로 및 주행환경을 관측할 수 있는 여러 센서를 배치 하였다. 장애물, 신호등, 주변 차량 정보를 디지털 가상환경 모델을 통해 획득하고 이를 심층강화학습 모델의 상태 공간에 맵핑하 여 주행거리 및 주행시간 관점의 성능을 최대화 하기 위한 주행 방향 조향 및 속도를 제어 하기 위한 행동을 동 적으로 결정한다. 본 논문에서는 우선순위 경험 재생 기반 DQN(Deep Q-Network)과 융합된 탐험 전략 및 새로운 보상함수 설계를 통해 빠른 학습 및 안정된 주행을 할 수 있도록 시스템을 설계하였다. 디지털 가상 공간에서의 실험을 통해 제안된 시스템은 vanilla DQN 보다 차선 유지, 장애물 회피, 신호등 준수에 따른 주행을 성공적으로 수행함을 검증하였다.

**키워드** : 심층강화학습, 자율주행, 디지털 가상환경, DQN, 우선 순위 경험 재생

**Key Words** : deep reinforcement learning, autonomous driving, digital virtual environment, ML-unity, DQN, PER

### ABSTRACT

In this paper, a self-driving system in a digital virtual road environment utilizing deep reinforcement learning is proposed. Using ML-Unity, a digital virtual environment is created to simulate a multi-lane road with various obstacles and traffic lights. Multiple sensors are deployed on the vehicle to observe the current road and driving environment, facilitating the development of the autonomous driving system. Information about obstacles, traffic lights, and surrounding vehicles is acquired through the digital virtual environment. This information is then mapped to the state space of the deep reinforcement learning model to dynamically determine actions, such as driving direction and speed, to maximize performance in terms of driving distance

\* 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음. (IITP-2021-0-020 52)

• First Author : Inha University Department of Information Communication Engineering, boutly@naver.com, 학생회원

° Corresponding Author : Inha University Department of Information Communication Engineering, sjyoo@inha.ac.kr, 종신회원

논문번호 : 202402-025-D-RN, Received February 6, 2024; Revised March 14, 2024; Accepted March 22, 2024

and time. The paper introduces a system design that combines priority experience replay-based Deep Q-Network (DQN) with exploration strategies and a novel reward function to achieve fast learning and stable driving. Through experiments in the digital virtual space, the proposed system is validated to successfully perform lane-keeping, obstacle avoidance, and compliant driving with traffic signals compared to vanilla DQN.

## 1. 서 론

최근 도로 교통 시스템에서의 자율주행 기술은 급속한 발전을 이루며, 이는 자동차 회사 뿐만 아니라 글로벌 빅테크 기업들도 참여하여 자율주행차 시장의 성장을 촉진하고 있다. 현재 Mercedes-Benz의 ‘Drive Pilot’, HONDA의 ‘Honda Sensing Elite’ 등과 같은 자율주행 기술은 자율주행 레벨 3 인증을 획득하며 기술의 안전성을 입증하고 있다. 한국에서는 SDV(Software Defined Vehicle)로의 전환을 밝힌 현대자동차가 80km/h 주행이 가능한 레벨 3 자율주행 상용화를 선언했고, 기아자동차 또한 대형 전기 SUV(Sports Utility Vehicle)에 레벨 3 자율주행 기술을 적용할 예정이다.

자율주행은 운전자의 개입 없이 차량 스스로 주행하는 기술을 말한다. 국제표준화기구(ISO)에서는 자율주행 수준을 0부터 5까지 6단계로 구분하고 있다. 레벨 0은 운전자가 모든 주행을 담당하는 수준이며, 레벨 5는 차량 스스로 모든 주행이 가능하다. 현재 상용화된 자율주행 기술은 레벨 2~3 수준에 해당한다. 레벨 3 자율주행은 조건부 자동화로 시스템이 일정 조건에서 운전을 제어하나, 시스템이 제어할 수 없는 상황에서는 운전자가 주행을 수행해야하는 단계를 나타낸다. 이는 자율주행의 초기 단계로 시스템이 운전을 제어할 때는 핸들에서 손을 떼는 ‘핸즈오프’와 전방에서 눈을 떼는 ‘아이즈오프’가 가능하다. 이를 넘어선 레벨 4 자율주행은 고도 자동화로 운전자가 필요 없는 자율주행에 돌입한다. 운전자가 필요 없다는 장점을 살려 향후 운송 및 물류 서비스 등 다양한 분야에서 의미있는 활용이 기대된다.

자율주행은 인공지능(AI: Artificial Intelligence) 기술을 기반으로 발전하고 있다. 컴퓨터 비전(computer vision)과 딥 러닝(deep learning) 기술을 활용해 주변 차량, 보행자, 신호등 등의 주변 환경을 인지하고, 이에 수집한 데이터를 기반으로 주행 경로 계획과 차량 제어에 사용한다.<sup>[1,2]</sup>

이러한 기술을 바탕으로 하는 고도 자율주행 차량을 상용화 하기 위해서 기술의 성능 검증과 안정성 평가가 매우 중요하다. 현실의 도로 환경에서는 다양한 주행

시나리오를 설계하고 큰 비용이 소모되기 때문에 제한적인 과정이다. 이러한 문제를 극복하기 위해 시물레이션 기반의 자율주행 기술에 대한 관심이 부상하고 있다.

시물레이션 기반의 자율주행 기술은 가상의 자율주행 차량과 다양한 주행 시나리오 구성하여 학습과 평가를 진행할 수 있다.<sup>[3]</sup> 가상공간에 실물의 환경을 구현하는 디지털 가상환경 (digital virtual environment) 기술은 실제 환경과 유사하면서도 더 다양한 시나리오에 대한 학습과 평가가 가능하기 때문에 비용과 시간이 훨씬 효율적이며, 안정성 및 안전성도 우수하다. 시물레이션 기반의 자율주행 알고리즘 평가와 실제 차량 실험과 비교를 통해 시물레이션 기반의 자율주행 연구가 충분히 가능하며,<sup>[4,6]</sup> 이를 바탕으로 시물레이션 기반의 연구가 빠르게 진화할 것으로 전망되고 있다.

또한, 강화학습(reinforcement learning)과 딥러닝을 결합한 심층강화학습(DRL: Deep Reinforcement Learning)을 활용한 시물레이션 기반 자율주행 연구와,<sup>[7-10]</sup> 더 나아가 시물레이션 상의 성능을 실제 세계에서도 안정적으로 적용하도록 하는 Sim2Real (Simulation to Real World) 연구도 활발히 진행되고 있다.<sup>[11]</sup>

본 논문에서는 심층강화학습을 활용한 디지털 가상도로 환경에서의 자율주행 시스템을 제안한다. ML-unity 기반으로 도로상에 장애물과 신호등이 배치된 다차선 도로 환경을 디지털 가상환경을 이용하여 구현하고, 자율주행 시스템 개발을 위해 차량에 현재 도로 및 주행환경을 관측할 수 있는 여러 센서를 배치 하였다. 장애물, 신호등, 주변 차량 정보를 디지털 가상환경을 통해 획득하고 이를 심층강화학습 모델의 상태 공간에 맵핑하여 주행거리 및 주행시간 관점의 성능을 최대화 하기 위한 주행 방향 및 속도를 제어 하기 위한 행동을 동적으로 결정한다. 본 논문에서는 우선순위 경험 재생 기반 DQN(Deep Q-Network)과 융합된 탐험 전략 및 새로운 보상함수 설계를 통해 빠른 학습 및 안정된 주행을 할 수 있도록 시스템을 설계하였다.

본 논문의 구성은 다음과 같다. 먼저, 2장에서는 논문에서 제안하는 자율주행 시스템의 강화학습 모델과 학습을 위한 디지털 가상 도로 환경을 소개한다. 3장에서는 해당 모델에서 사용되는 우선 순위 재생 기법 기반

DQN과 자율주행을 위해 강화학습에 필요한 상태, 행동, 보상 함수의 세부적인 내용을 설명한다. 4장에서는 심층강화학습기반의 자율주행 시스템을 구현하기 위해 사용하는 추가적인 기법들을 설명한다. 5장에서는 가상 도로 환경에서의 학습을 통해 시스템의 성능 평가 및 분석을 하고, 마지막으로 6장에서는 결론을 내린다.

## II. 강화학습기반 자율주행 시스템

본 장에서는 시뮬레이션 기반의 자율주행 시스템 모델에 대해 설명한다. 시스템 모델의 전체적인 구조는 그림 1과 같다. python API와 ML-unity 간의 데이터 상호작용을 통해 심층강화학습 모델 학습이 진행되고 이로써 차선 유지, 장애물 회피, 신호 준수에 대한 자율주행을 목표로 한다.

ML-unity 상의 차량 에이전트(car agent)가 도로 환경(environment)에서의 주행으로 도로, 장애물, 신호등에 대한 관찰 데이터(observation data)를 수집하고, 행동(action)으로 얻은 보상(reward)과 함께 데이터를 python API로 전송한다. 수집한 관찰 데이터는 전처리하여 심층강화학습에 필요한 상태(state)로 정의하고, 이를 강화학습 모델에 입력하여 학습된다. 모델로부터 결정된 행동은 다시 ML-unity로 전송되어 차량 에이전트가 행동을 수행하면서 학습이 반복된다.

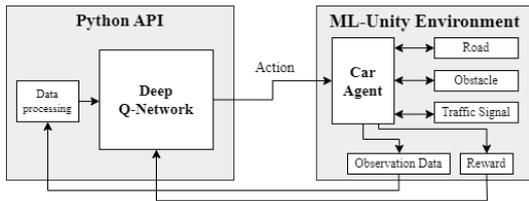


그림 1. 강화학습기반 자율주행 시스템 구조  
Fig. 1. Autonomous driving system architecture

### 2.1 자율주행을 위한 강화학습 모델

본 논문에서는 자율주행 학습을 위해 심층강화학습 방법 중 하나인 DQN을 활용한다. 강화학습은 주어진 환경에서 에이전트가 상태에 따른 행동을 선택하고 이에 대한 보상을 통해 학습이 진행된다. DQN은 Q-learning과 딥러닝을 결합한 방법으로, Q-learning은 각 행동에 대한 행동 가치(Q-value)를 예측하는 강화학습의 방법 중 하나이다. 그러나, 상태와 행동의 수가 많아질수록 Q-learning의 Q-table을 사용한 행동 가치 결정은 불가능에 가깝다.

이 문제를 해결하기 위해 DQN은 Q-table을 대신하

여 딥 러닝을 활용한다. DQN은 딥 러닝을 사용하여 상태를 입력으로 받아 각 행동에 대한 가치를 근사하는 심층 신경망을 구현한다. 이를 통해 복잡한 상태와 행동 공간에 대한 행동 가치를 효율적으로 계산할 수 있다.

DQN은 행동 가치를 이용한 손실 함수(loss function)를 통해 모델 업데이트가 진행된다. 손실함수는 예측된 행동 가치와 실제 행동 가치 간의 차이를 측정하며, 이 차이가 줄어드는 방향으로 모델이 학습된다. 아래 식(1)과 같이 손실 함수를 정의한다.

$$y_{target} = R_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta^{target})$$

$$Loss = \{y_{target} - Q(s_t, a_t; \theta^{pred})\}^2 \tag{1}$$

위 식에서 타겟 값( $y_{target}$ , target value)은 현재 시점  $t$ 에서 얻은 보상  $R_t$ 와 미래 보상을 조절하는 감가율  $\gamma$ 과 다음 상태  $s_{t+1}$ , 다음 행동  $a_{t+1}$ 을 target network에 입력하여 목표하는 최대 행동 가치를 이용하여 계산한다. 현재 상태  $s_t$ , 현재 행동  $a_t$ 을 메인 네트워크인 Q-network에 입력하여 예측한 근사 행동 가치와의 타겟 값과 차이의 제곱을 손실(loss)로 하여 손실이 줄어드는 방향으로 손실 함수의 경사를 계산하여 모델의 파라미터를 업데이트 하는 경사 하강법(gradient descent)을 사용하여 최적화한다. 이 과정을 반복하면서 최종적으로 손실 함수의 전역 최적점(global optimum)을 찾아 가는 방향으로 학습한다.

### 2.2 디지털 가상 가상 도로 환경 모델

본 논문의 자율주행 시스템 학습과 평가를 위해 ML-unity 기반의 디지털 가상 환경을 구축한다. 디지털 가상환경 모델이란 컴퓨터 속에 현실의 가상환경을 만들고, 시뮬레이션을 통해 결과를 미리 예측할 수 있는 기술이다. 이를 통해 가상 환경에서 문제점을 예측하여 실제세계에서의 계획 수립과 효율성을 극대화할 수 있다.

이러한 가상 환경을 구현하기 본 연구에서는 그림 2와 같이 ML-unity에서 unity asset인 ‘cartoon road constructor’를 이용하여 최대 왕복 4차선 도로를 생성하였다. 또한 차선 유지, 장애물 회피, 신호 준수에 대한 학습을 위해 도로 상에 정지/주행 차량 장애물을 포함하여 unity의 3d object를 활용한 신호등을 도로 상에 배치하였다.

해당 도로 환경에서 주행하는 차량 에이전트를 구성하여 모델이 결정한 행동에 따라 자율주행이 이루어지도록 한다. 차량 에이전트에는 ML-unity에서 제공하는 레이 센서와 카메라를 이용해 행동이 각 시점에서 관찰



그림 2. ML-Unity 기반 디지털 가상 환경  
Fig. 2. Virtual road environment based on ML-Unity

데이터를 수집하고, 이를 통해 모델에 입력되는 상태 데이터를 구성한다.

### III. 심층강화학습기반 장애물과 신호등을 고려한 다차선 자율주행 모델

본 장에서는 자율주행 시스템의 핵심 요소인 심층강화학습 모델에 대해 상세히 서술한다. 먼저, 차량 에이전트와 가상 환경이 상호작용하며 학습하는 강화학습 구조를 설명하며, 이를 통해 전체적인 학습의 흐름을 이해한다. 다음으로, 심층 신경망 구조를 활용한 패턴 학습 과정을 서술한다. 또한 차량 에이전트가 학습을 위해 관찰하고 해석해야 하는 상태 공간과, 가능한 행동 선택지를 이루는 행동 공간에 대해 설명한다. 마지막으로, 차량 에이전트가 어떤 행동을 선택하게 될지 결정하는데 중요한 역할을 하는 보상 함수 설계 구조를 다룬다.

#### 3.1 자율주행 심층강화학습 모델

본 논문은 우선순위 경험재생 (prioritized experience replay) 방법을 활용한 DQN으로 학습을 진행하며 전체적인 구조는 그림 3과 같다.

경험 재생(experience replay)은 환경으로부터 얻은 경험 튜플(experience tuple)을 재생 버퍼(replay buffer)에 저장하여 배치 사이즈만큼 무작위 샘플링하여 Q-Network를 업데이트하는 방법이다. 우선 순위 경험 재생은 식(1)과 동일한 계산으로 얻은 TD-error

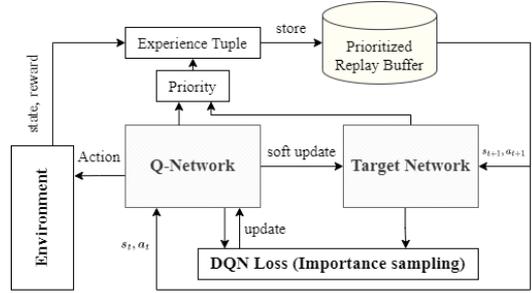


그림 3. 자율주행 모델의 강화학습 구조  
Fig. 3. Reinforcement learning architecture

(Temporal Difference error)  $\delta_t$ 로 아래 식(2)에 따라 각 경험에 대한 우선순위  $p_t$ 를 결정한다.

$$p_t = |\delta_t| + \epsilon \quad (2)$$

위 식에서 TD-error의 절댓값에 매우 작은 실수  $\epsilon$ 를 더해 우선순위를 결정하여 모든 경험의 취할 확률이 0이 아님을 보장한다. 하지만, 이러한 방식은 항상 우선순위가 큰 경험만을 훈련시켜 과적합이 발생하기 때문에 식(3)로 계산한 확률론적 우선순위와 식(4)의 중요도 샘플링 가중치(importance sampling weights)를 활용한다.<sup>[12]</sup>

$$P(i) = \frac{p_i^a}{\sum_k p_k^a} \quad (3)$$

$$w_i = \left(\frac{1}{N} \cdot \frac{1}{P(i)}\right)^b \quad (4)$$

식(3)의  $a$ 는 0에 가까울수록 무작위 샘플링에 가까워지며, 1에 가까울수록 우선순위가 높은 경험 선택에 가까워진다. 또한 자주 보이는 경험의 가중치를 줄여 업데이트를 조정하는 식(4)의  $w_i$ 를 사용한다.  $N$ 은 재생 버퍼의 크기이고,  $b$ 는 이러한 중요도 샘플링 가중치가 학습에 미치는 영향을 제어하는 변수이다. 이는 초기에 작은 값부터 시작해 학습 후반부에는 1까지 증가하면서 모델이 안정적으로 수렴하도록 한다. 이후 학습을 위해 배치된 경험은 다시 한번 TD-error를 계산 하여 우선순위를 업데이트 한다.

제안된 DQN기반 심층 강화학습 신경망은 그림 4와 같이 CNN(Convolutional Neural Network)와 MLP(Multi-Layer Perceptron)을 통과하여 얻은 특징 및 패턴 데이터를 dueling 구조에 입력하여 특정 상태에 대한 가치인 상태 가치(state value)와 주어진 상태에서 특정 행동이 다른 행동에 비해 가지는 상대적 가치를 의미하

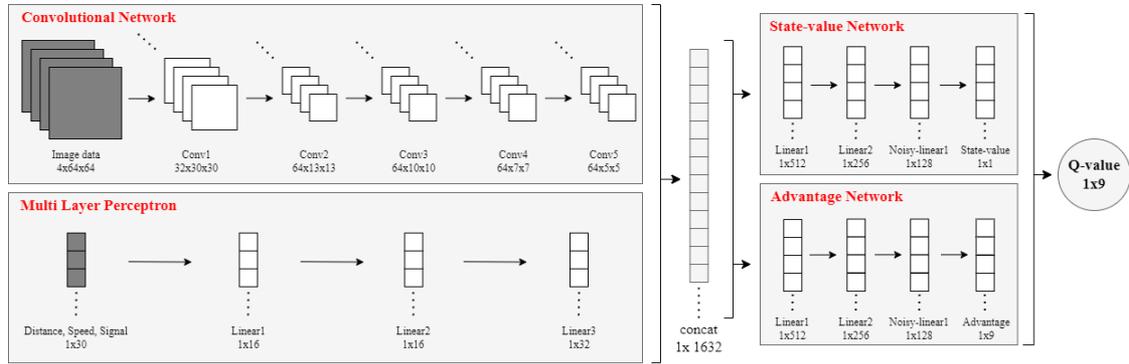


그림 4. 자율주행 시스템의 심층 신경망 구조  
Fig. 4. Deep neural network architecture

는 어드밴티지(advantage)를 출력하여 최종적으로 행동 가치를 근사한다.

CNN은 이미지 데이터에 대한 특징을 추출하며 4 step의 차량 전방 이미지를 입력한다. 5번의 합성곱(convolution)이 진행되며, 출력층을 제외한 각 합성곱 이후에는 활성화 함수인 GELU를 통과한다. MLP는 차량 에이전트와 연석/장애물 간의 거리, 속도 리스트, 신호등 one-hot encoding data 리스트를 결합하여 입력하여 패턴을 추출한다. 3번의 선형 변환(linear transform)이 진행되며 출력층의 제외한 각 변환 이후에는 GELU를 통과한다.

CNN과 MLP의 출력을 결합하여 상태 가치 네트워크(state-value network)와 어드밴티지 네트워크(advantage network)에 각각 입력한다. 두 네트워크의 구조는 동일하나 별도의 네트워크이며, 2번의 선형 변환과 2번의 잡음 선형 변환(noisy linear transform)로 진행된다. 출력층을 제외하고 각 변환 이후 GELU를 통과한다. 이를 통해 구해진 상태 가치와 어드밴티지로 다음 식과 같이 최종적인 행동 가치를 구한다.

$$Q(s, a) = V(s) + A(s, a) - \frac{1}{|N|} \sum_a A(s, a) \quad (5)$$

단순히 V(s)와 A(s, a)를 더하여 Q(s, a)를 구하게 되면, 어떤 V(s)와 A(s, a)에 얼마나 영향을 받는지 알 수 없기 때문에 식별 가능성 문제가 발생한다. 그래서 어드밴티지의 평균을 활용해 식별 가능성 문제를 해결하고 안정성을 높인다.

### 3.2 자율주행 차량 제어를 위한 상태 정의

본 논문의 자율주행이 차선 유지, 장애물 회피, 신호준수의 목표를 이루기 위해 ML-unity 기반 가상 도로

환경에서 다양한 데이터 수집이 필요하며, 이를 활용해 다양한 상황을 충분히 반영할 수 있는 상태 공간을 정의해야 한다.

차량 에이전트가 ML-unity 상에서 제공하는 카메라 센서와 레이 센서로 관측한 데이터를 활용하여 상태를 정의하여 다음과 같이 차량 전방 관촬 이미지, 에이전트와 연석/장애물 간 거리, 속도 리스트, 신호등 one-hot encoding data 리스트로 구성한다.

차량 전방 이미지는 그림 5에서와 같이 도로 상의 모든 정보를 관촬하는 차량 전방 카메라로 이미지를 캡처 하여 64x64 크기의 gray-scale로 조정하고 4 step을 쌓아서 활용한다. 이러한 이미지 스택은 FIFO(First In First Out) 방식으로 업데이트되며 매 step마다 오래된 이미지는 제거하고 새로운 이미지를 추가한다.

차량 에이전트와 연석/장애물 간의 거리는 그림 6의 ML-Unity가 제공하는 레이 센서를 활용해 수집한다. 두 개의 레이 센서를 활용하여 연석과 장애물 각각의 거리를 측정한다. 연석과의 거리 측정 센서는 전방 180°에 펼쳐진 19개 중 5개를 활용하고, 장애물 거리 센서

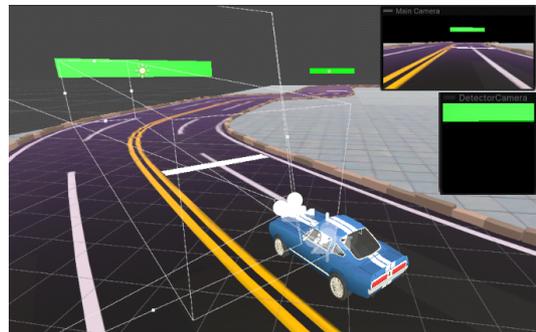


그림 5. 차량 전방 이미지, 신호등 인식 카메라  
Fig. 5. Image and traffic light recognition camera

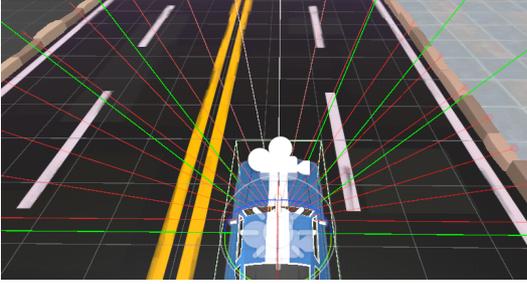


그림 6. 연석, 장애물 거리 측정 레이 센서  
Fig. 6. Curb and obstacle measurement ray sensor

또한 전방 180°에 펼쳐진 11개로 거리를 측정하고 정규화 하여 활용한다. 또한 속도 리스트는 매 step의 정규화된 속도 값을 FIFO 방식의 리스트에 저장하여 4 step의 데이터를 유지한다.

차량 전방 카메라 외에도 도로 상의 신호등만을 캡처하는 카메라가 차량 에이전트에 부착되어 있기 때문에 이를 활용하여 매 step마다 pixel 기반의 one-hot encoding data를 생성한다. 이 또한 4 step의 데이터를 FIFO 방식으로 유지한다.

### 3.3 자율주행 시스템에서의 행동 정의

정의된 상태를 통해 목표로 하는 주행이 진행되도록 적절한 행동 공간을 구성해야 한다. DQN은 각 행동에 대한 행동 가치를 근사하여 행동을 결정하기 때문에 가능한 행동을 무한한 공간에서 결정하기 어렵다. 따라서 해당 모델에서는 이산적인 행동 공간을 적용한다. 이를 바탕으로 유한한 행동공간에 따라 다양한 주행 상황에서 더 큰 보상을 얻기 위한 행동을 선택할 수 있도록 효과적인 정의가 필요하다.

차량 에이전트의 행동은 조향각(steering turn)과 속도(speed)로 정의된다. 이에 따라 행동 공간은 식(6)과 같이 정의된다.

$$Action = \begin{cases} a_0: stop \\ a_1: accelerate \\ a_2: decelerate \\ a_3: turn right \\ a_4: turn left \\ a_5: accelerate with right turn \\ a_6: accelerate with left turn \\ a_7: decelerate with right turn \\ a_8: decelerate with left turn \end{cases} \quad (6)$$

위 식에서 차체대로 감속, 가속, 좌회전, 우회전 감소 행동의 조합과 정지 행동으로 총 9개의 이산적인 행동으로 이루어진다. 회전량은 시뮬레이션 상 1 step이 진

행되는 시간(0.1s)를 고려하여 결정하였다. 또한 현재 step에서 고려되는 행동은 이전 속도나 이전 조향각의 영향을 받으며, 이를 통해 ML-Unity 도로 환경에서 실제 차량과 같은 부드러운 주행을 보여줄 수 있다. 속도의 범위는 [0m/s, 12m/s], 조향각의 범위는 [-4.5°/step, 4.5°/step]이다.

### 3.4 보상 함수 설계

자율주행 시스템의 학습 목표는 차선 유지, 장애물 회피, 신호 준수를 이루며 주어진 도로 환경을 한 바퀴 주행하는 것이다. 목표를 달성하기 위해서는 각 행동에 따라 적절한 보상과 페널티가 부여되어야 한다. 이를 위해 보상 함수는 세 가지 주요 요소로 구성되며, 각 행동에 따른 즉각적인 보상 또는 페널티가 부여된다. 차량 에이전트가 주어진 상태에서 적절한 행동을 취하면 보상을 얻게 되어 올바른 주행을 유도한다. 반면, 부적절한 행동을 취하면 페널티를 얻어 에이전트가 해당 주행을 피하도록 학습한다. 또한 학습이 올바른 방향으로 나아가기 위해 보상, 페널티 스케일의 균형을 맞추는 것이 매우 중요하다.

본 논문의 차선 주행 학습을 위한 보상 함수 구조는 아래 식(7)과 같다.

$$R_{lane} = \begin{cases} +1.4, & driving\ in\ lane\ 1 \\ +0.8, & driving\ in\ lane\ 2 \\ -0.4, & close\ to\ center\ line \\ -1.3, & cross\ center\ line \\ -10.0, & hit\ a\ curb \end{cases} \quad (7)$$

식(7)에 따라 1,2 차선에서 주행 시 보상을 얻으며, 중앙선에 가까워지거나 중앙선을 넘어 주행하는 경우에는 페널티를 부여한다. 아래 식(8)은 장애물 회피를 위한 보상 구조이다.

$$R_{obstacle} = \begin{cases} -10.0, & obstacle\ collision \\ +0.5, & obstacle\ avoidance \end{cases} \quad (8)$$

정지/주행 장애물과 충돌 시에는 큰 페널티를 부여하고, 전방 장애물 거리 측정 센서를 이용해 에이전트가 장애물을 회피했다고 판단하면 약간의 양의 보상을 부여한다. 아래 식(9)는 신호 준수를 위한 보상 구조이다.

$$R_{traffic\ light} = \begin{cases} +1.2, & stop\ at\ red\ signal\ (in\ Box) \\ +1.2, & move\ at\ green\ signal\ (in\ Box) \\ -1.0, & stop\ at\ green\ signal\ (in\ Box) \\ -1.0, & move\ at\ red\ signal \\ -0.5, & stop\ at\ no\ signal \end{cases} \quad (9)$$

신호등 캡처 카메라에 의해 신호등 없음, 빨간(노란) 신호, 초록 신호가 식별된다. 이에 따라 신호등의 정지선 부근의 box collider와의 충돌 여부에 따라 보상이 부여된다. 빨간 신호 일 때 box collider 내에서 정지를 유지하거나, 초록 신호일 때 box collider를 통과하면 보상이 부여된다. 반대로 초록 신호 일 때 box collider 내에서 정지 유지, 또는 빨간 신호일 때 box collider를 통과하면 페널티가 부여된다. 마지막으로는 신호등이 존재하지 않는데 정지해도 페널티가 부여된다.

$$R_{goal} = \begin{cases} +1.0, & \text{midpoint} \\ +10.0, & \text{goal} \end{cases} \quad (10)$$

마지막으로, 위 식(10)에서 볼 수 있는 것처럼 학습 속도 향상을 위해 도로 주요 포인트마다 약간의 보상과, 도로 한 바퀴 주행을 완료 시 큰 보상을 부여한다. 최종적인 보상총합은 다음 식 (11)과 같이 정의된다.

$$Reward = R_{lane} + R_{obstacle} + R_{traffic\ light} + R_{goal} \quad (11)$$

#### IV. 구현 상 기법

본 장에서는 가상 도로 환경에서의 자율주행 시스템에 대한 안정적이고 효율적인 학습이 진행될 수 있도록 고려한 기법들을 소개한다.

##### 4.1 단계적으로 복잡성이 증가하는 도로 환경

본 논문에서 목표하는 바를 달성하기 위해 가상 도로 환경 초기에 모든 목표에 대한 학습을 진행하기에는 무리가 있다. 학습 초기에는 모델학습이 불안정하기 때문에 초반부터 복잡한 환경과 상태가 주어진다면 학습에 어려움이 존재한다. 그렇기 때문에 차선 유지, 장애물 회피, 신호 준수에 대한 학습을 단계적 진행, 즉 단계적으로 복잡성이 증가하도록 가상 도로 환경을 구현하여 적용하였다.

도로 초기에는 차량 장애물과 신호등을 배치하지 않고 차선 유지를 위한 학습에 집중한다. 이를 통해 학습 초기에는 연선에 충돌하지 않고 차선을 유지하는 것을 학습한다. 어느정도 차선 유지에 대한 학습이 진행되었다고 판단되는 구간부터 장애물 또는 신호등을 각각 배치하여, 차선 유지와 더불어 장애물 회피 또는 신호 준수에 따른 학습이 진행되도록 한다. 도로 중반부 이후에는 장애물과 신호등이 복합적으로 배치되며, 차선유지, 장애물 회피, 신호 준수를 모두 달성할 수 있도록 학습

한다. 마지막으로 복잡한 커브 구간과 장애물을 연속적으로 회피하는 구간을 추가하여 한 바퀴 주행을 종료하도록 하였다.

이와 같이 초기에는 간단한 주행을 학습하고, 시나리오가 진행됨에 따라 복잡한 상황을 학습할 수 있도록 디지털 가상 도로 환경을 구현하였다.

##### 4.2 융합된 탐험 전략

구축한 도로 환경에서의 효과적인 학습을 위해서 탐험(exploration)과 활용(exploitation)의 적절한 균형을 조절하는 것이 필수이다. 그렇기 때문에 해당 도로 환경을 효과적으로 활용하기 위한 탐험 기법으로 엡실론 그리디 코사인 감쇠(epsilon greedy cosine decaying)과 NoisyNet을 활용한다.

엡실론 그리디 코사인 감쇠는 학습 초기에는 새로운 경험 획득에 집중하고, 학습이 진행되면서 경험 활용으로 최적의 정책을 학습하도록 돕는 기법이다. 초기 엡실론(epsilon)을 1로 설정하고, 코사인 감쇠(cosine decaying)을 통해 특정 step 이후 최소 엡실론을 유지하도록 한다. 이러한 감쇠방법을 사용하는 이유는 탐험 비율을 부드럽게 감소하여 초기에 환경에 대한 다양한 데이터 수집을 충분히 하기 위함이다. 매 step마다 [0,1] 사이의 실수를 무작위로 선정하고 엡실론 값과 비교를 통해 행동을 결정한다. 무작위 값이 엡실론보다 작을 경우에는 무작위 행동을, 반대로 클 경우에는 근사된 행동 가치 중 최대 값을 가지는 행동을 수행한다.

다음으로, NoisyNet은 기존 모델 파라미터에 노이즈(Noise)를 추가함으로써 확률적인 행동 선택을 통해 탐험이 진행되도록 하는 기법이다. 본 논문의 시스템에서는 factorized gaussian noise 방식으로 노이즈를 분배한다. 이는 가우시안 분포 (gaussian distribution)에서 각 입력과 출력에 대해 독립적으로 노이즈를 생성하는 방법으로, 해당 분포에서 난수를 추출하여 노이즈를 생성하고 이를 파라미터에 더함으로써 모델의 행동을 확률적으로 조절한다. 아래 식(11)은 일반적인 선형 레이어, 식(12)는 노이즈가 추가된 선형 레이어의 구조이다.<sup>[13]</sup>

$$y = \omega x + b \quad (11)$$

$$y = (\mu^\omega + \sigma^\omega \odot \varepsilon^\omega)x + \mu^b + \sigma^b \odot \varepsilon^b \quad (12)$$

식(11)에서  $x$ 를 입력하면 가중치  $\omega$ 와 편향  $b$ 에 따라  $y$ 를 출력한다. 이에 노이즈 파라미터(noise parameter)인  $\mu^\omega, \sigma^\omega, \mu^b, \sigma^b$ 와 노이즈 변수(noise variable)인

$\epsilon^a$ ,  $\epsilon^b$ 가 추가되어 식(12)와 같게 된다. 가우시안 분포에서 입력의 개수  $p$ , 출력의 개수  $q$ 만큼 노이즈를 뽑아  $p \times q$ 로  $\epsilon^a$ ,  $q$ 로  $\epsilon^b$ 를 생성하고, 노이즈 파라미터에 의해 노이즈 변수의 분산이 결정된다. 학습이 진행됨에 따라 가중치와 편향이 조정되므로, 즉 노이즈의 크기가 조절됨을 의미한다. 학습 초기에는 노이즈 파라미터가 크게 설정되어 탐험이 활발히 일어나고, 학습이 진행됨에 따라 작아지게 됨으로 정보를 유지하며 성능 개선의 효과를 기대할 수 있다.

이러한 두 탐험 기법을 함께 활용하는 것이 본 논문의 가상 도로 환경에 적합하다고 판단하였다. 엡실론 그리디 감쇠 기법은 상태에 상관없이 무작위로 탐험이 진행되며, 이는 시간이 지남에 따라 탐험이 최소화되므로 단계적으로 복잡성이 증가하는 가상 도로 환경에서 한계가 존재한다. 그에 반면에 NoisyNet은 학습 시간에 상관없이 상태에 따른 행동이 확률적으로 결정하여 탐험한다. 즉, 학습 초기에는 두 기법을 함께 활용하여 활발한 탐험을 진행하고, 이후에는 NoisyNet에 의해 확률적 탐험이 유지되면서 환경에서 안정적이고 유연한 학습이 진행된다.

융합된 탐험 전략을 통해 본 논문의 가상 도로 환경에서 목표하고자 하는 바를 효과적으로 학습할 수 있다.

### 4.3 기존방법과의 구현 방법 비교

심층강화학습을 이용하는 기존 자율주행 학습 모델들은 자율주행의 도로환경 (다차선, 단일차선 포함), 차량 및 차량주변 도로 장치에서의 센서 종류 및 기능, 다른 차량의 존재 유무, 차선간 이동제약조건, 장애물 및 신호등 존재 유무 등의 환경조건이 매우 다르고 차량

의 동작제어 방법 역시 매우 제한적이거나 보다 복잡한 제어가 가능한 경우 등 다양하다. 또한 각 방법들이 추구하는 목적이 다르며 보상함수의 정의 및 기준 역시 상이하여 직접적인 실험결과와의 비교는 어려운 점이 있다. 본 장에서는 기존 방법들 중 제안된 방법과 가장 유사한 목적을 갖는 방법 [5]와 방법 [6]에 대해 제안된 방법과 실험환경 구성, 강화학습 방법론, 학습목적, 행동정의 및 보상함수의 범위 등에 대한 비교 설명을 표 1에 기술하였다.

## V. 시뮬레이션 환경 및 결과

본 장에서는 심층강화학습기반 자율주행 모델을 디지털 가상 도로 환경에서의 학습하고 성능 평가를 진행한다. 심층강화학습 모델은 python과 pytorch, cupy, numpy 라이브러리를 활용해 구현하였다. 도로 환경은 unity와 ML-agent toolkit을 활용한 ML-unity로 도로와 장애물, 신호등, 그리고 차량 에이전트를 구성하였다. ML-Unity 기반 환경에서 차량 에이전트가 행동으로 획득한 관찰 데이터를 활용하여 상태 공간을 구성하고 이를 python의 모델로 입력하여 차선 유지, 장애물 회피, 신호 준수를 위한 학습을 진행하였다.

차량 에이전트에 부착된 카메라를 활용하여 실시간 주행을 확인할 수 있도록 하여, 자율주행 시스템 목표에 따른 심층강화학습 모델 학습의 진행도를 정성적으로 확인하였다. 또한 학습을 진행하면서 한 에피소드에서의 보상 합과 결정된 행동의 분포, 에피소드의 첫 step 의 최대 행동 가치, 그리고 실시간 주행 영상을 확인하

표 1. 기존방법과의 심층강화학습 기반 자율주행 방법론 비교표  
Table 1. Comparison of deep reinforcement learning-based autonomous driving methodologies with existing methods

| 방법       | 학습 방법                      | 학습 환경    | 학습 목적                                    | 상태 정의   | 행동 정의                | 보상 함수                                      |
|----------|----------------------------|----------|--|---|----------------------|--|
| Proposed | Dueling DQN+ PER+ NoisyNet | ML-Unity | 장애물 회피, 차선 유지, 신호 준수, 차량 추돌회피, 시간절약      | 차량 전방 Gray-scale 이미지, 인식 거리, 장애물 거리, 신호 정보, 속도 정보 | Discrete             | 연석과의 거리, 장애물 충돌/회피, 신호 준수 여부, 중간 포인트/목표 달성 |
| [5]      | DQN                        | CARLA    | 차선 유지, 신호 준수, 앞 차량 추돌, 보행자탐지/회피, 주행속도 제어 | 주변 환경 RGB 이미지, 주변 환경의 3D정보, 현재 차량 위치, 속도 정보       | Continuous, Discrete | 목표 달성도, 장애물 충돌/회피, 주행 경로의 효율성, 신호 준수 여부    |
| [10]     | DQN + LSTM                 | CARLA    | 차선 유지, 선행 차량과의 안전거리 유지                   | 주변 환경 RGB 이미지, 차량 속도, 도로 중앙으로부터의 차량 각도            | Discrete             | 차선 유지, 선행 차량과의 안전거리 유지, 속도 제한 준수, 충돌 회피    |

면서 시각적인 분석이 가능하도록 하였다. 최종적으로 학습 종료되면 보상 합과 주행 시간에 대한 저장된 데이터를 활용하여 시스템의 성능 평가를 수행하였다.

### 5.1 ML-Unity 시뮬레이션 환경

본 논문의 심층강화학습 모델 학습과 시스템성능 평가를 위해 ML-unity 기반의 시뮬레이션 환경을 제작하여 활용한다. 아래 그림 7의 가상 도로 환경은 전체 크기 200 unity x 100 unity의 왕복 4차선 도로로 구성된다.

도로 환경에는 그림 8의 차량 장애물과 신호등을 배치하여, 장애물 회피와 신호 준수를 위한 주행 경험을 수집할 수 있도록 하였다.

차량 에이전트는 본 논문의 시스템이 목표로 하는 자율주행 학습을 위해 카메라와 레이 센서를 활용하여 필요한 관찰 데이터를 수집한다.

차량 장애물은 위치가 고정된 정지 차량과 주어린 차선을 주행하는 주행 차량으로 나뉘어진다. 에이전트인 파란 차량을 제외한 나머지 차량은 장애물이다. 해당 환경에서 표 2와 같은 하이퍼 파라미터를 설정하여 학습을 진행하였다. 본 논문의 심층강화학습 모델 학습과 시스템성능 평가를 위해 ML-unity 기반의 시뮬레이션 환경을 제작하여 활용한다. 아래 그림 7의 가상 도로 환경은 전체 크기 200 unity x 100 unity의 왕복 4차선 도로로 구성된다.

도로 환경에는 그림 8의 차량 장애물과 신호등을 배치하여, 장애물 회피와 신호 준수를 위한 주행 경험을

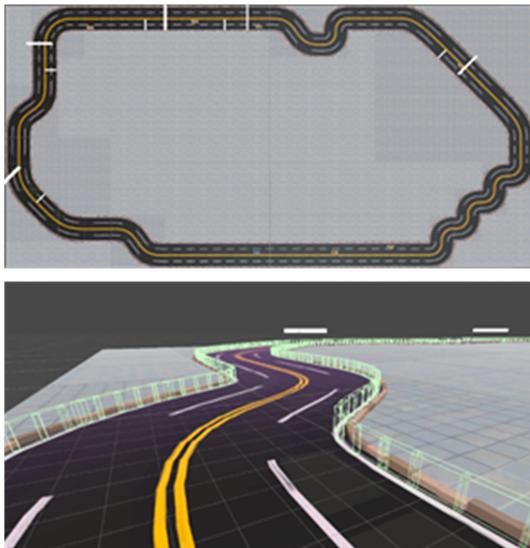


그림 7. ML-Unity 기반 가상 도로 환경  
Fig. 7. Virtual road environment based on ML-Unity

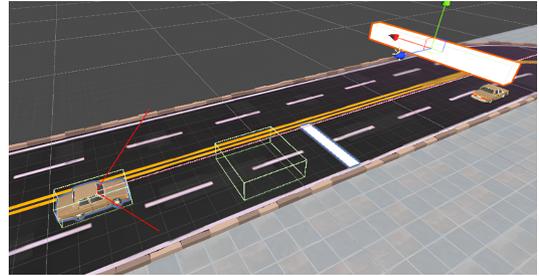


그림 8. 도로에 배치된 장애물과 신호등  
Fig. 8. Obstacles and traffic lights placed on the road

수집할 수 있도록 하였다.

차량 에이전트는 본 논문의 시스템이 목표로 하는 자율주행 학습을 위해 카메라와 레이 센서를 활용하여 필요한 관찰 데이터를 수집한다. 차량 장애물은 위치가 고정된 정지 차량과 주어린 차선을 주행하는 주행 차량으로 나뉘어진다. 에이전트인 파란 차량을 제외한 나머지 차량은 장애물이다. 해당 환경에서 표 2와 같은 하이퍼 파라미터를 설정하여 학습을 진행하였다.

표 2. 심층 신경망 하이퍼 파라미터  
Table 2. Deep neural network hyperparameters

| Hyperparameter   | Value                                  |
|--|--|
| Learning rate  | 0.00002                                |
| Batch size   | 8                                      |
| Discount factor  | 0.95                                   |
| Number of actions                                      | 9                                      |
| Epsilon-greedy decaying                                | [0.1, 1] for 200000 step               |
| Soft target update rate warmup                         | [0.005, 0.04]<br>0.005 for 150000 step |
| Prioritized experience Replay_e                        | 1e-8                                   |
| Prioritized experience Replay_α                        | 0.6                                    |
| Prioritized experience Replay_β                        | 0.3                                    |
| Prioritized experience Replay_β increment per sampling | 0.000001                               |
| Weight initializer                                     | GELU                                   |
| Optimizer  | Adam                                   |
| Episode  | 5000                                   |

### 5.2 시뮬레이션 결과

본 논문에서의 다양한 상황을 포함하는 디지털 가상 도로환경에서 심층강화학습 기반 자율주행 시스템을 구현하고 시뮬레이션을 통해 차선 유지, 장애물 회피, 신호 준수를 확인하고 성능을 평가하였다. 시스템의 성능 평가는 학습률, 엡실론, 강화학습 모델에 따른 보상

합과 주행 시간의 비교를 통해 진행하였다.

아래 그림 9는 학습률에 따른 보상 합의 추세를 보이며, 50 에피소드 보상 합을 활용한 수치이다. 그림 9에서, 학습률이 높을수록 학습 초기에는 보상 합이 빠르게 증가하지만, 학습 후반에는 보상 합의 변동이 심하거나 크게 감소하는 것을 볼 수 있다. 또한 보상 합의 최대 값은 제일 큰 학습률을 활용했을 때에 확인할 수 있다. 이에 대한 이유는 본 논문의 강화학습에서 우선순위 재생 기법 사용과 가상 도로 환경이 단계적으로 복잡성이 증가하기 때문이라고 할 수 있다. 도로 초반부에는 차선 주행 학습에 집중하도록 다른 요소들이 배치되어 있지 않기 때문에 빠르게 학습이 진행되어 더 많은 구간을 주행할 수 있지만, 신호등과 장애물이 관찰되면서 높은 학습률로 인해 최적점을 찾지 못하는 결과를 보인다. 따라서 학습률을 0.00002로 고정하고 탐험 비율을 조정하여 시스템을 평가하였다.

그림 10은 엡실론 그리디 코사인 감소 주기를 다르게 하여 학습을 진행했을 때의 보상 합이다. 엡실론에 도달하기까지 200000 step 이상 소요, 즉 해당 step 이상 탐험의 비율이 높게 되면 수집한 경험을 제대로 활용하지 못하여 수렴하지 못해 학습 진행이 어려운 것을

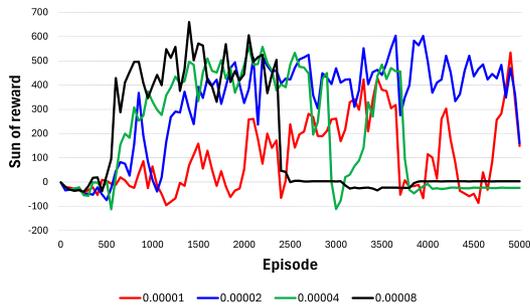


그림 9. 학습률에 따른 보상 합  
Fig. 9. Reward sum according to learning rate

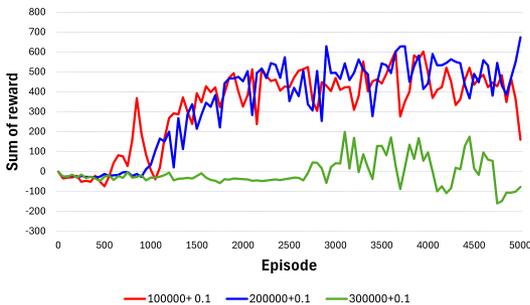


그림 10. 엡실론 감소 주기 변화에 따른 보상합  
Fig. 10. Reward sum according to the epsilon decaying period

확인하였다.

그림 11은 각 감소 주기 별 최소 엡실론에 따른 보상 합이다. 엡실론 감소 주기를 증가시킬수록 최소 엡실론을 조금씩 줄여 나가는 것이 좋은 성능을 보임을 확인할 수 있다.

그림 12는 본 논문의 심층강화학습 모델과 다른 모델의 보상 합을 비교하는 그래프다. 여기서도 복잡성이 낮은 학습 초반의 차선 주행 학습에서는 오히려 vanilla DQN과 NoisyNet의 학습이 빠르게 진행되는 것을 확인할 수 있다. 하지만 학습이 진행될수록 본 논문의 모델이 안정적임을 볼 수 있다. 이는 모델이 어느정도 수렴해 나가면서 우선순위 경험 재생과 중요도 샘플링에

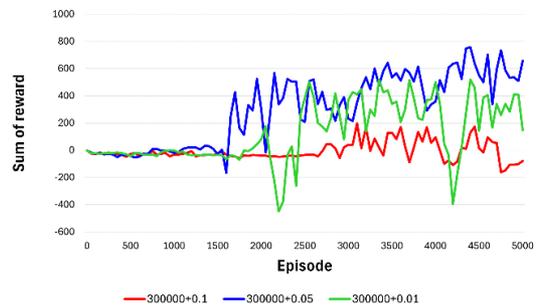
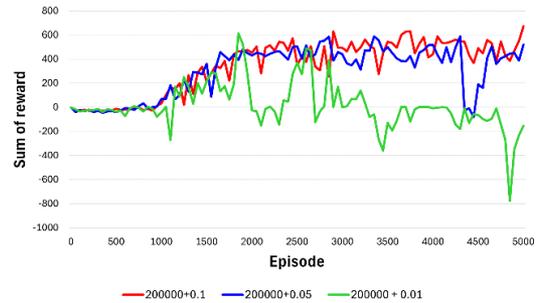


그림 11. 최소 엡실론에 따른 보상 합  
Fig. 11. Reward sum according to minimum epsilon

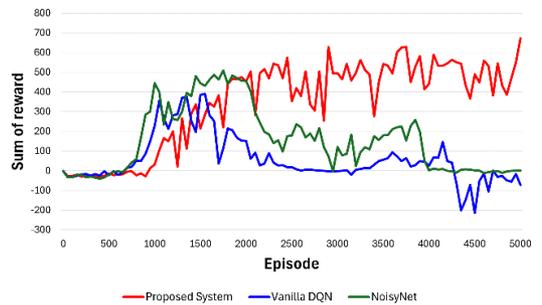


그림 12. 강화학습 모델에 따른 보상 합  
Fig. 12. Reward sum according to reinforcement learning model

의해 복잡해지는 도로 환경에서도 잘 적응해 나가는 것으로 판단된다.

그림 13은 주행 시간을 비교하는 그래프이다. 주행 시간은 step 수를 의미한다. 엡실론 감소 주기가 길고 최소 엡실론이 작으면 더 긴 주행시간을 가지지만, 보상 합 그래프와 함께 고려해 보았을 때 상황에 불필요한 정지 행동이 많이 발생하는 것을 유추해볼 수 있다. 그렇기 때문에 본 시스템의 목표에 최적화된 엡실론 감소 주기는 200000 step, 최소 엡실론은 0.1임을 도출하였다.

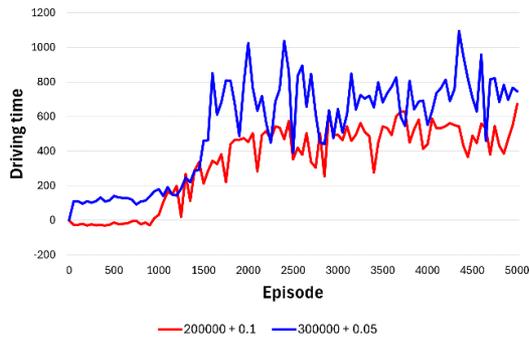


그림 13. 에피소드에 따른 주행 시간  
Fig. 13. Driving time according to episodes

## VI. 결 론

본 연구에서는 심층강화학습기반 ML-unity 디지털 가상 도로 환경에서의 자율주행 시스템을 제안하였다. 디지털 가상 환경에서 차량 에이전트는 다차선 도로에서 다양한 상황에 대한 관찰 데이터를 수집하여 자율주행 학습에 활용하였다. 복잡한 관찰 데이터로 인해 단계적으로 학습의 복잡성을 높이는 방식으로 도로 환경을 구축하였다. 이에 따라 심층강화학습 모델 역시, 우선순위 경험 재생 기법과 융합된 탐험 전략을 활용하여 구축된 환경에서 우수한 성능을 보일 수 있도록 하였다. 마지막으로, 기존 모델과의 성능 비교를 통해 본 논문의 ML-unity 환경에서 우수한 성능을 보이며, 다차선 도로에서의 차선 주행, 장애물 회피, 신호 준수에 따른 주행을 확인하였다.

## References

[1] K. S. Arikumar, A. D. Kumar, T. R. Gadekallu, S. B. Prathiba, and K. Tamilarasi, "Real-time 3D object detection and classification in autonomous driving

environment using 3D LiDAR and camera sensors," *Electronics*, vol. 11, no. 24, 4203, 2022.

(<https://doi.org/10.3390/electronics11244203>)

- [2] Y. Zhang, K. Liu, H. Bao, X. Qian, Z. Wang, S. Ye, and W. Wang, "AFTR: A robustness multi-sensor fusion model for 3D object detection based on adaptive fusion transformer," *Sensors*, vol. 23, 8400, 2023. (<https://doi.org/10.3390/s23208400>)
- [3] Y. Li, W. Yuan, S. Zhang, W. Yan, Q. Shen, C. Wang, and M. Yang, "A survey of simulators for autonomous driving: Taxonomy challenges, and evaluation metrics," *arXiv preprint arXiv:2311.11056*, 2023. (<https://doi.org/10.48550/arXiv.2311.11056>)
- [4] I. G. Daza, R. Izquierdo, L. M. Martínez, et al., "Sim-to-real transfer and reality gap modeling in model predictive control for autonomous driving," *Appl. Intell.* vol. 53, pp. 12719-12735, 2023, (<https://doi.org/10.1007/s10489-022-04148-1>)
- [5] B. Osínski, et al., "Simulation-based reinforcement learning for real-world autonomous driving," *2020 ICRA*, pp. 6411-6418, Paris, France, 2020. (<https://doi.org/10.1109/ICRA40945.2020.9196730>)
- [6] J. Kwak and K. Yi, "Development of simulation environment for autonomous driving algorithm validation based on ROS," *J. Auto-vehicle Safety Assoc.*, vol. 14, no. 1, pp. 20-25, Mar. 2022. (<https://doi.org/10.22680/kasa2022.14.1.020>)
- [7] W.-H. Kim, S.-H. Wang, D.-S. Jeon, and D.-S. Eom, "Comparison and analysis of deep reinforcement learning algorithms for indoor autonomous vehicles," in *Proc. Symp. KICS*, pp. 779-780, 2020.
- [8] B. R. Kiran, et al., "Deep reinforcement learning for autonomous driving: A survey," in *IEEE Trans. Intell. Transport. Syst.*, vol. 23, no. 6, pp. 4909-4926, Jun. 2022. (<https://doi.org/10.1109/TITS.2021.3054625>)
- [9] Ó. Pérez-Gill, et al., "Deep reinforcement

learning based control algorithms: Training and validation using the ROS Framework in CARLA simulator for self-driving applications,” *2021 IEEE Intell. Veh. Symp. (IV)*, pp. 1268-1273, Nagoya, Japan, 2021. (<https://doi.org/10.1109/IV48863.2021.9575616>)

- [10] M. Ahmed, C. P. Lim, and S. Nahavandi, “A deep q-network reinforcement learning-based model for autonomous driving,” *2021 IEEE Int. Conf. SMC*, pp. 739-744, Melbourne, Australia, 2021. (<https://doi.org/10.1109/SMC52423.2021.9658892>)
- [11] W. Zhao, J. P. Queralta, and T. Westerlund, “Sim-to-real transfer in deep reinforcement learning for robotics: A survey,” *2020 IEEE SSCI*, pp. 737-744, Canberra, ACT, Australia, 2020. (<https://doi.org/10.1109/SSCI47803.2020.9308468>)
- [12] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, “Prioritized experience replay,” *4th ICLR 2016*, pp. 4-5, San Juan, Puerto Rico, 2016. (<https://doi.org/10.48550/arXiv.1511.05952>)
- [13] M. Fortunato, M. G. Azar, B. Piot, J. Menick, I. Osband, A. Graves, V. Mnih, R. Munos, D. Hassabis, O. Pietquin, C. Blundell, and S. Legg, “Noisy networks for exploration,” *CoRR*, p. 4, 2017. (<https://doi.org/10.48550/arXiv.1706.10295>)

이재영 (Jae-yeong Lee)



2018년 3월~현재 : 인하대학교  
정보통신공학 (학사 재학)  
<관심분야> 인공지능, 머신러닝,  
알고리즘, 데이터 마이닝

유상조 (Sang-Jo Yoo)



1988년 2월 : 한양대학교 전자통신학과 (공학사)  
1990년 2월 : 한국과학기술원 전기및전자공학과 (공학석사)  
2000년 8월 : 한국과학기술원 전자전산학과 (공학박사)  
1990년 3월~2001년 2월 : KT 연구개발본부

1990년 3월~2000년 11월 : NIST(미국 표준기술연구원) 초빙연구원  
2001년 3월~현재 : 인하대학교 정보통신공학과 교수  
<관심 분야> 사물인터넷, 무선 네트워크, 강화학습, 인공지능 응용, 차량네트워킹  
[ORCID:0000-0003-1533-0814]