

Reinforcement Learning-Based Path Planning for Mobile Robot in a Digital Twin Environment

Minkyung Jun, Hoeryong Jung†

Department of Mechanical Engineering, Konkuk university

Introduction

- 현재 로봇 분야에서는 장애물 회피, 경로 계획, 목적지 도착 등 다양한 영역에서 강화학습을 활용한 주행 연구가 활발히 진행되고 있음.
- 실제 환경에서는 복잡한 상황을 재현하여 실험하는 데 한계가 있어, 디지털 트윈 환경이 필요함.
- 본 연구에서는 Unity ML-Agents를 활용해 모바일 로봇 주행을 위한 강화학습 시스템을 설계하고, 다양한 보상 함수 및 종료 조건을 비교함. 또한, 보다 정밀한 디지털 환경 구축을 위해 NVIDIA Isaac Sim을 활용함.

Method & Result

1. 강화학습 기반 모바일 로봇 주행 설계

1) Unity ML-Agents를 활용한 강화학습 환경 구축

- Fig.1과 같이 Unity에서 간단한 미로 환경을 제작하였으며, 태스크는 정적 장애물을 회피하여 목적지에 도달하는 것으로 설정함.
- 로봇은 Clearpath Husky A200이며, 센서는 Velodyne VLP-16 LiDAR를 장착함.
- 강화학습 알고리즘으로 PPO를 적용하였으며, ROS와 연동하여 Unity와 데이터를 주고 받도록 구성함.
- State는 식 (1)과 같이 로봇의 위치, 속도, 목표 지점의 위치, LiDAR 거리 데이터로 구성됨.
- Action는 식 (2)와 같이 로봇의 선속도와 각속도이며, 보상 함수는 식 (3)에 따라 목적지 도착 시 보상을 부여하고, 충돌 시 에피소드를 종료하도록 설정함.

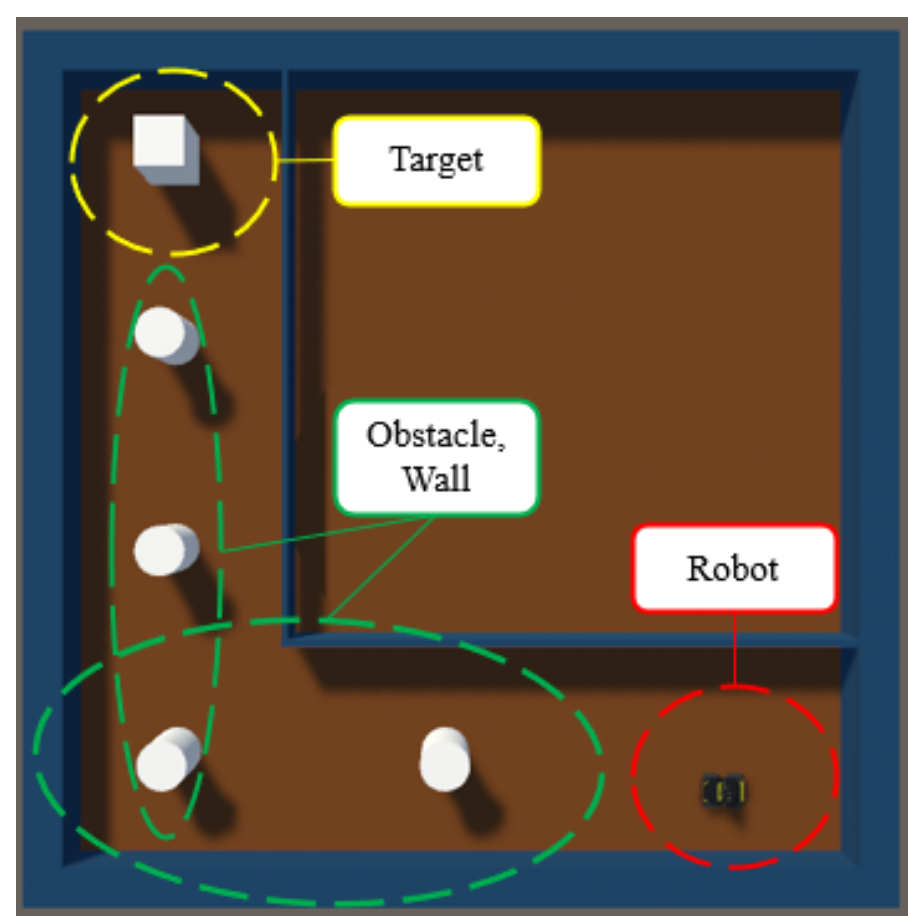


Fig 1. Simple maze environment

$$S_t = [S_t^{husky}, S_t^{target}, LiDAR_measurement] \quad (1)$$

$$S_t^{husky} = [P_x, P_y, V_x, V_z]$$

$$S_t^{target} = [P_x, P_y]$$

$$LiDAR_measurement = ranges[897]$$

$$A_t = [v, \omega] \quad (2)$$

$$R = R_{goal} = \begin{cases} 1, & \text{if } d_{husky \text{ to target}} \leq 1.42 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

2) 다양한 보상 함수 설계 및 최적화 실험

- 다양한 학습 양상을 분석하기 위해, 기본 설정 외에도 [Table 1]과 같이 여러 보상 함수와 종료 조건을 실험함.
- 조건에 따라 학습 시간 단축, 의도적인 충돌, 신중한 장애물 회피 등 로봇의 행동 양상이 달라지는 것을 확인할 수 있었음.
- 복합적인 보상 구조를 적용한 학습 결과는 Fig.2와 같으며, 보상 값은 약 11에 수렴하고 학습에는 약 3시간이 소요됨.

Time	$R_{step} = -0.01$ (per step) episode_length = 60 seconds
Distance	$R_{dist} = Dist_{previous} - Dist_{current}$
Collision	$R_{collision} = -1.0$ if collides

Table1. RMSE of each model

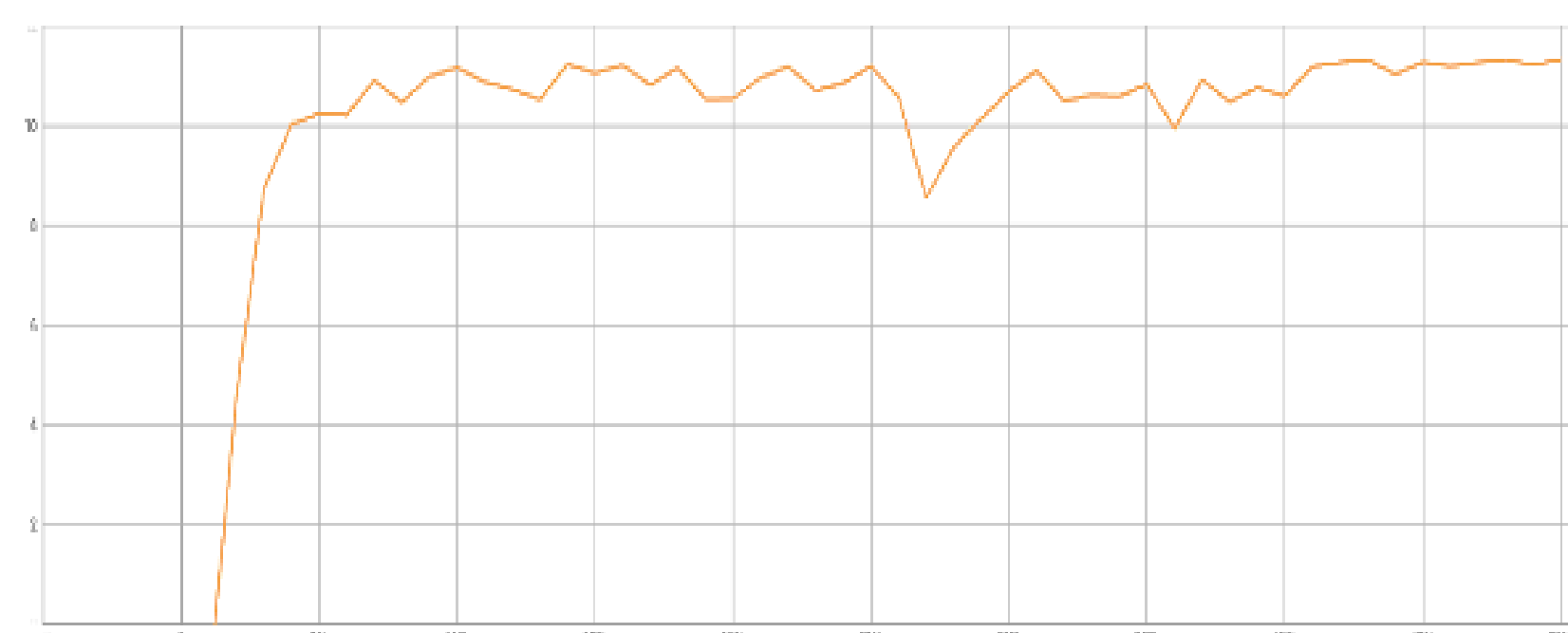


Fig 2. First graph of cumulative reward over step

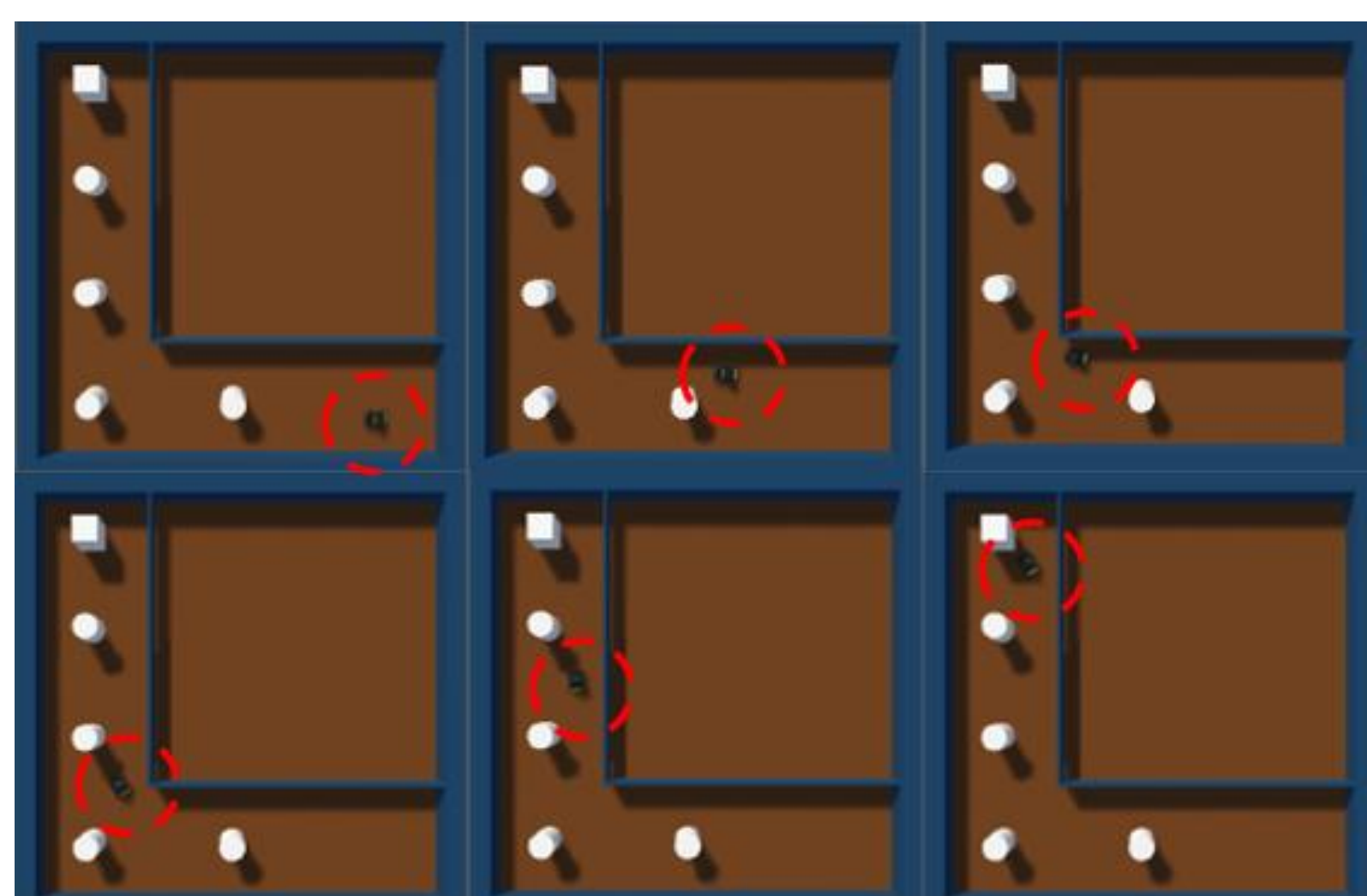


Fig 3. First trajectory of the trained robot
(from top-left to right)

2. 디지털 트윈 환경 구축 및 검증

1) Unity 기반 디지털 트윈 환경 구축

- 실제 환경 검증을 위해 Fig.4와 같이 연구실 복도를 Unity로 모델링하여 실험을 진행함.
- 학습 결과, Fig.5에서 보듯이 누적 보상은 약 14.9에 수렴하였으며, 학습 시간은 3시간이 소요됨.
- Fig.6과 같이 로봇이 최적의 코너링을 하며 경로를 따라 주행하는 모습을 보임.

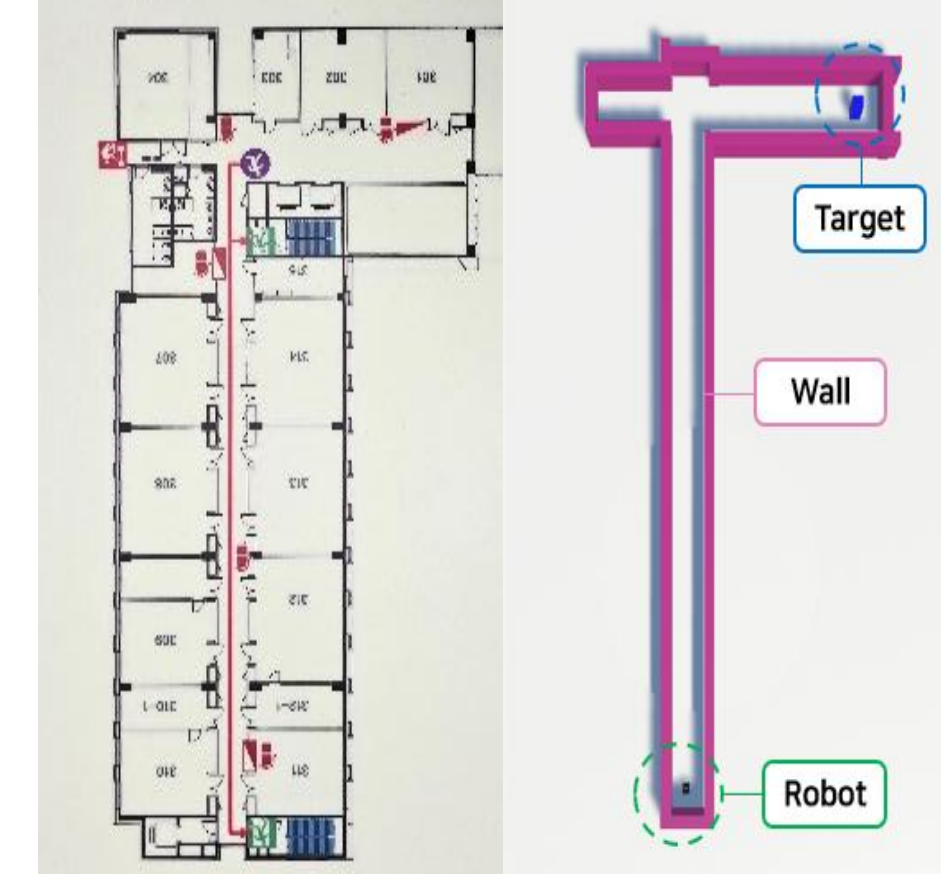


Fig 4. Real environment, Unity
(from left to right)

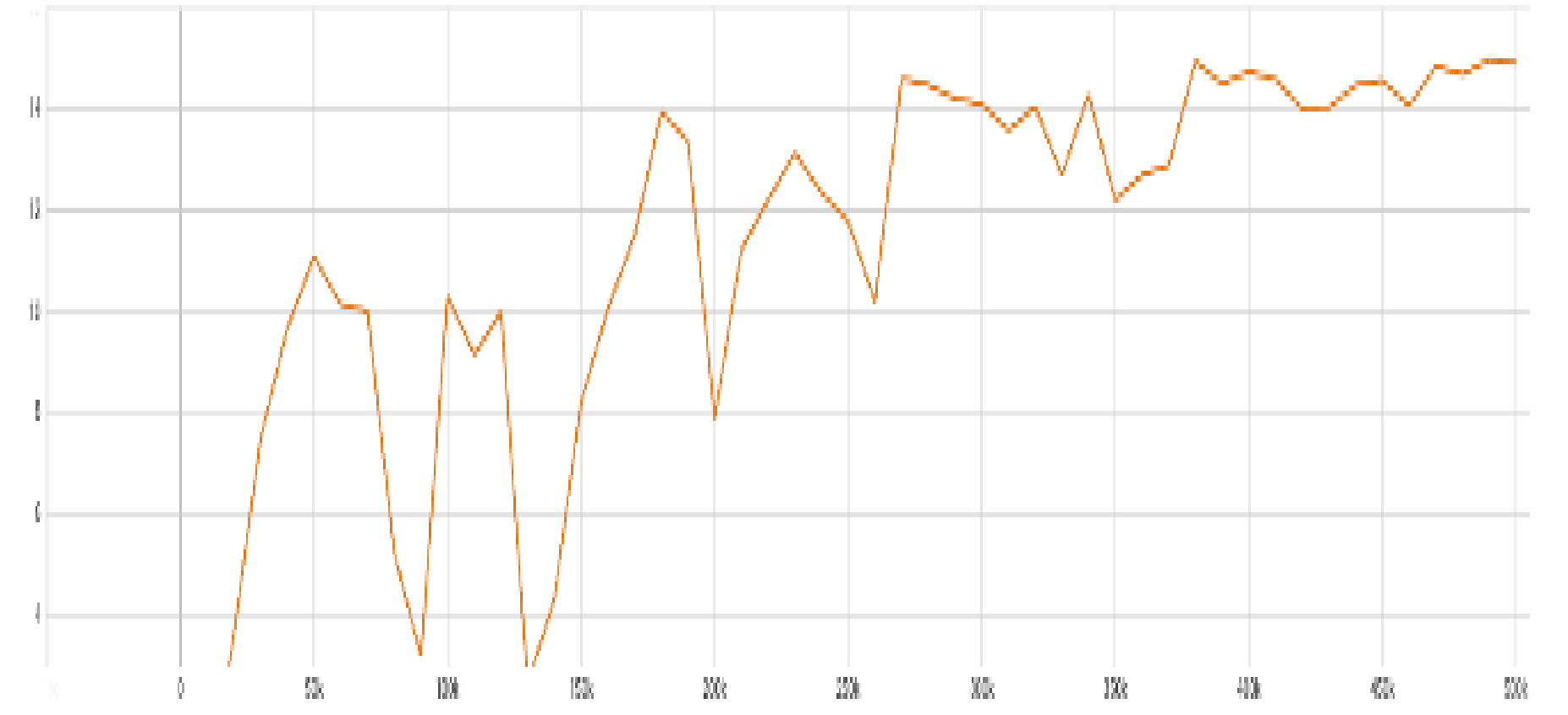


Fig 5. Second graph of cumulative reward over step

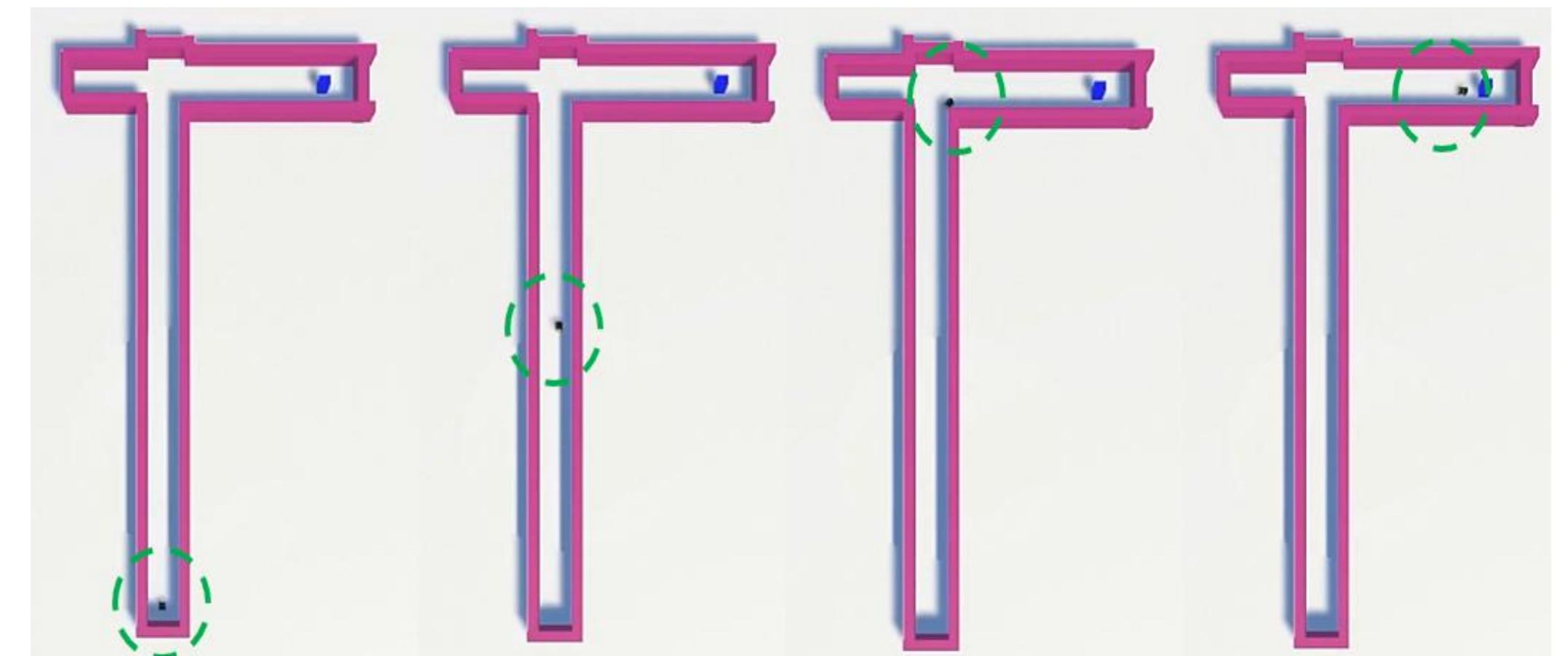


Fig 6. Second trajectory of the trained robot (from left to right)

2) 학습된 모델을 실제 환경에 적용 및 성능 검증

- 실제 환경에서는 LiDAR 데이터 차이로 인해 로봇이 벽에 충돌하는 문제가 발생함.
- 해결책으로 LiDAR 데이터에 Gaussian Noise를 추가하고, Fig.7과 같이 직진 및 교차로 경로를 별도로 학습하도록 설정함.
- 학습 결과, Fig.8에서 보듯이 누적 보상은 약 5에 수렴하며, 학습 시간은 3시간이 소요됨.
- 이전보다 실제 환경에서 직진 경로를 안정적으로 주행하는 모습을 확인함.

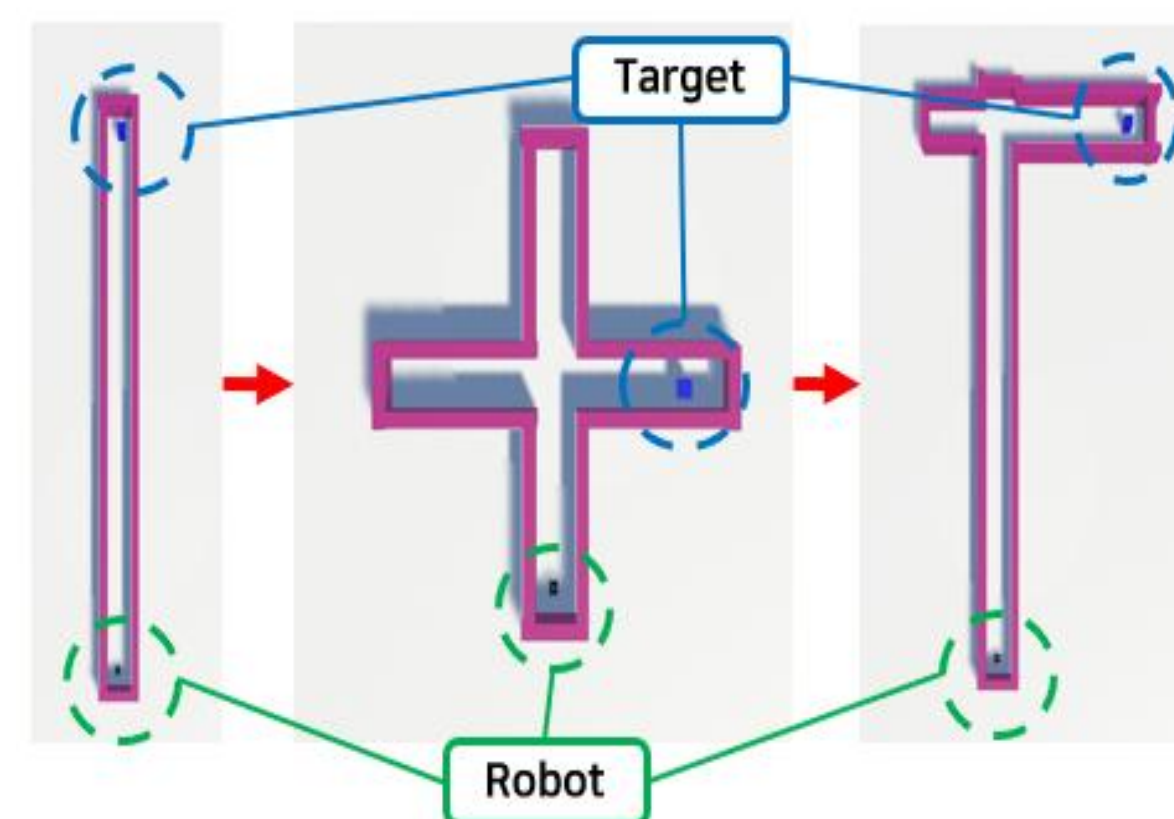


Fig 7. Straight, Intersection, Combine
(from left to right)

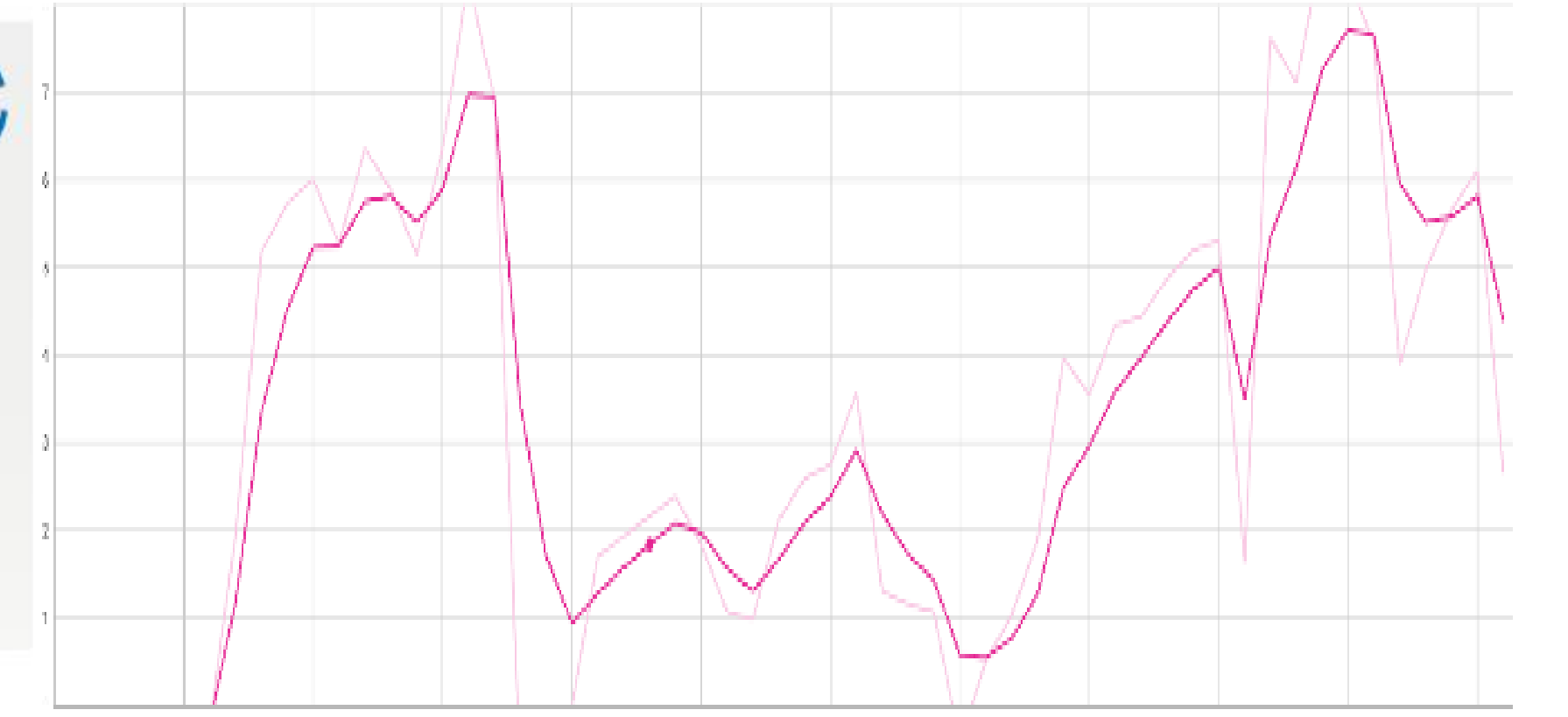


Fig 8. Third graph of cumulative reward over step

3. 고정밀 디지털 트윈 전환 (Isaac Sim)

- 실제와 시뮬레이션 간의 차이를 최소화하기 위해 Isaac Sim을 활용하여 학습 환경을 이전함.
- 3D 스캐너로 실제 공간을 스캔한 후 조명과 재질을 추가하여 Fig.10과 같이 정밀한 가상 환경을 구축함.
- 실제 로봇에 장착된 Velodyne VLP-16과 동일한 설정의 RTX LiDAR Config를 제작하여 시뮬레이션과 실제 환경의 유사성을 더욱 향상함.



Fig 9. Real environment, Digital Twin in Isaac Sim (from left to right)

Conclusion

- 본 연구에서는 강화학습과 디지털 트윈 환경을 활용하여 모바일 로봇의 정적 장애물 회피 및 목적지 도착 태스크를 구현함.
- 다양한 보상 함수와 종료 조건을 비교하여 최적의 학습 파라미터를 도출하고, 디지털 트윈 환경에서 학습을 진행한 후 실제 환경에서 성능을 검증함.
- 정밀한 디지털 트윈 환경을 구축하기 위해 3D 스캐너와 Isaac Sim 플랫폼을 활용하여 실제와 유사한 가상 공간을 조성함.
- 향후 연구에서는 동적 장애물 회피 및 실외 환경 등 더욱 복잡한 시나리오를 학습하고, 실제 환경에서 검증하고자 함.