# Unity3D 가상 환경에서 강화학습으로 만들어진 모델의 효율적인 실세계 적용

임은아*, 김나영*, 이종락**, 원일용*
*서울호서전문학교 사이버해킹보안과
**영남이공대학교 사이버보안과
e-mail:forestea@naver.com,skdud990306@naver.com,jllee@ync.ac.kr,
clccclcc@shoseo.ac.kr

# Applying Model to Real World through Robot Reinforcement Learning in Unity3D

En-A Lim*, Na-Young Kim*, Jong-lark Lee**, Ill-yong Weon*
*Dept of Cyber Security, Seoul Hose Techincal College
**Dept of Cyber Security, YeungNam University College

요 약

실 환경 로봇에 강화학습을 적용하기 위해서는 가상 환경 시뮬레이션이 필요하다. 그러나 가상 환경을 구축하는 플랫폼은 모두 다르고, 학습 알고리즘의 구현에 따른 성능 편차가 크다는 문제점이 있다. 또한 학습을 적용하고자 하는 대상이 실세계의 하드웨어 사양이 낮은 스마트 로봇인 경우, 계산량이 많은 학습 알고리즘을 적용하기는 쉽지 않다. 본 연구는 해당 문제를 해결하기 위해 Unity3D에서 제공하는 강화학습 프레임인 ML-Agents 모듈을 사용하여 실 환경의 저사양 스마트 로봇에 장애물을 회피하고 탐색하는 모델의 강화학습을 적용해본다. 본 연구의 유의점은 가상 환경과 실 환경의 유사함과 일정량의 노이즈 발생 처리이다. 로봇의 간단한 행동은 원만하게 학습 및 적용가능함을 확인할 수 있었다.

## 1. 서론

강화학습은 기계 학습의 한 영역으로 주어진 상황에서 어떠한 행동을 취할지를 학습하는 것을 의미한다[1]. 강화학습을 적용하기 위한 전제조건은 무한 반복이 보장되어야 하며 환경의 초기화가 용이해야 한다. 이러한 강화학습의 특성으로 인해 실세계의 로봇에 강화학습을 적용하는 것은 어려운 일이다. 또한, 실세계에서 강화학습이 가능한 환경을 구축했다 하더라도 물리적 시간 흐름에 제어를 받기 때문에 비교적 시간 제어에 제약이 덜한 가상 환경에서의 학습보다 비효율적이다.

강화학습을 실제 문제에 적용하고자 시도한 여러 연구가 존재한다. 자율 주행 자동차[2], 보행 로봇[3], 무인수상선[4], 무인항공기[5] 등이 사례들이다. 이러한 연구들은 해당 문제를 극복하기 위해 각각의 실험에 적합한 가상 환경을 구축하여 시뮬레이션을 진행함으로써 문제를 해결한다. 그러나 가상 환경을 구축하는 플랫폼은 모두 다르고, 학습 알고리즘의 구현에 성능 편차가 크다는 문제점이 있다. 그리고 학습을 적용하고자 하는 대상이 실세계의 하드웨어 사양이 낮은 스마트 로봇인 경우, 계산량이 많은 학습 알고리즘을 적용하기는 쉽지 않다.

본 연구는 하드웨어 사양이 좋지 못한 스마트 로봇에 강화학습을 효과적으로 적용하는 것에 대한 것이다. 우리는 실제 환경을 가상 환경으로 시뮬레이션하여 효과적으로 모델을 학습한 후 해당 모델을 저사양 스마트 로봇에 적용하였다. 특히 가상 환경은 3D 모델링 표준 플랫폼으로 널리 사용되는 Unity3D를 사용하였으며, 학습 알고리즘의 안정성을 위해 동일한 플랫폼에서 제공하는 ML-Agents 모듈을 사용하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를, 3장에서는 제안하는 시스템의 개념 및 구성을 언급하였다. 4장에서는 실험 및 결과를 서술했으며, 5장에서는 결론 및 향후 과제를 기술하였다.

## 2. 관련 연구

### 2.1 Unity3D ML-Agents

ML-Agents는 개발자 및 AI 연구자를 위해 Unity3D에서 개발한 플랫폼으로 지능형 에이전트를 개발하고 학습시킬 수 있는 환경을 제공하는 시뮬레이션 플러그인이다.

ML-Agents는 Reinforcement Learning(강화학습), Imitation Learning(모방학습), neuroevolution, 그 외의 다른 알고리즘을 이용하여 크게 4가지 방식으로 에이전트를

학습시킬 수 있는 환경을 제공한다.

본 연구는 ML-Agents 환경 중 강화학습 환경을 이용하여 진행된다.

2.2 로봇의 행동(원더링)

원더링이란 특정 장애물을 로봇이 감지하여 피해 가는 움직임을 뜻한다[6].

전통적인 로봇의 개념과 달리 스스로 상황을 판단하고 자율적으로 움직이는 로봇(자율이동로봇)[7]은 실생활에 응용되기 위해 불확정적으로 변화하는 환경에서의 적응 능력이 필요하다. 이를 위하여 환경 특정적인 지식이나 외부의 명시적인 제어 없이 주어진 환경에 적응할 수 있는 능력을 갖춘 로봇을 개발하려는 연구가 진행되어 왔다[8].

본 연구는 강화학습을 이용하여 특정 환경에 대한 지식이나 제어 없이 로봇이 원더링 할 수 있음을 보이고자 한다.

2.3 Reinforcement Learning

'Learning, improving, and generalizing motor skills for autonomous robot manipulation : an integration of imitation learning, reinforcement learning, and deep learning = 자율 로봇 매니퓰레이션을 위한 로봇 운동 솜씨 학습, 개선 및 일반화(한양대학교 대학원, 조남준)'의 논문에도 나와 있듯이, 강화학습은 주변 상태에 따라 행동을 할지 판단을 내리는 주체인 agent와 해당 agent가 속한 환경으로 구성된다[7].

agent가 행동하면 그에 따라 상태가 바뀌고 보상이 결정된다. 즉, 주어진 환경에서 보상을 최대한 많이 받도록 agent가 학습하는 것이다. 정확하게 정의를 해보면 아래의 <표 1> 와 같다.

<표 1> 강화학습 용어

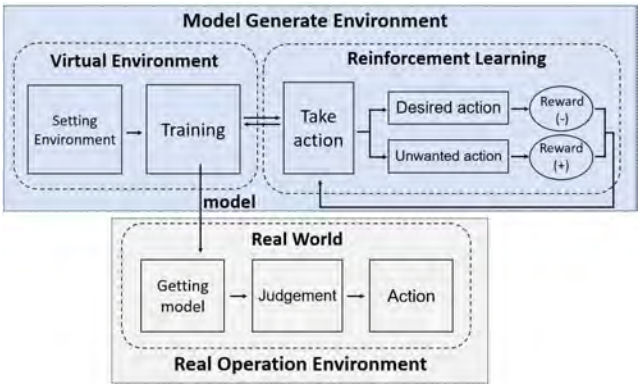
| | 정의 |
|---|---|
| State | Agent가 인식하는 자신의 상태 |
| Action | environment에서 특정 state에 갔을 때 action을 지시하는 것 |
| Reward | Agent가 action을 취하면 그에 따른 reward를 environment가 Agent에게 알려주는 것 |

따라서 s라는 state에 있을 때, a라는 action을 취했을 경우 얻을 수 있는 값이 reward이다.

$$R_s^a = E[R_{t+1} | S_t = s, A_t = a]$$

## 3. 시스템 구성

시스템의 전체적 구성은 (그림 1)과 같다. 실제 환경을 시뮬레이션 할 수 있는 가상 환경 모듈과 가상 환경과 상호작용을 통하여 모델을 생성하는 학습 모듈로 나누어진다. 이러한 2개의 환경을 통해 실제 환경에서 사용할 모델을 생성하고 해당 모델을 실제 환경에서 사용하는 구성이다.




(그림 1) 전체 시스템 구성도

시스템이 효율적으로 작동하기 위해서 가장 중요한 것은 가상 환경이 얼마나 실제 환경을 반영하는가이다. 학습 부분에서 가장 중요한 것은 실제 환경과 유사한 로봇의 움직임과 이 움직임에 의해 발생하는 행동의 결과다. 특히 실제 환경과 가상 환경에서 사용하는 모든 수치 단위가 상대적으로 일정한 비율을 반영하는 것이 중요하다. 즉, 로봇의 출력(운동)과 입력(센싱)의 수치 단위가 실제 환경의 n배를 유지하도록 가상 환경을 고려해야 한다.

전체적인 작업 흐름은 아래와 같다.

1. Configure real environment and check the measured values
2. Configure virtual environment reflected the real environment
3. Training in the virtual environment to make model
4. Convert the model to using in the real environment
5. Apply the model to the real environment

(그림 2) 전체 절차

## 4. 실험 및 결과

우리의 실험 구성은 다음과 같다. 먼저 Unity3D로 실제 스마트 로봇의 환경을 만든다. 이 로봇은 거리를 측정할 수 있는 3개의 센서를 가지고 있는데, 해당 센서로부터 환경을 인지하여 장애물을 회피 및 무작위 탐색(원더링)을 수행한다. 구축한 가상 환경에서는 ML-Agents 모듈을 통해 강화학습을 수행한다. 가상 환경에서 로봇이 장애물을 회피하며 원활하게 탐색을 진행하면 학습을 종료시킨다. 이후 실제 환경에 이 모델을 적용하여, 로봇이 장애물을 잘 회피하며 환경을 탐색하도록 하는 것이 목적이다.

4.1 실제 환경

실제 환경에서 구동되는 로봇은 전면에 3개의 거리 측정 센서(HC-SR04)를 달고, 움직이는 자동차 로봇이다. 제

어 컴퓨터는 라즈베리파이(4B)를 사용하였다. 또한 강화학습을 원만하게 사용하기 위해 edge TPU(Coral TPU)를 사용했다.


(그림 3) 실세계 로봇

로봇의 거리 센서는 최대 약 450cm까지 측정하며, 1초에 약 48cm를 이동한다. 구현한 로봇의 기본적인 움직임은 아래 5개의 명령으로 정의하고 구현하였다.

<표 2> 로봇의 움직임 명령어

| | Agent's action |
|---|---|
| go() | 앞으로 이동 |
| back() | 뒤로 이동 |
| turnLeft() | 왼쪽으로 회전 |
| turnRight() | 오른쪽으로 회전 |
| stop() | 멈춤 |

4.2 가상 환경

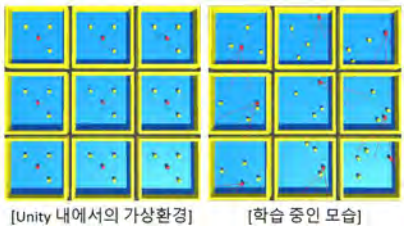
가상 환경에서의 로봇은 사각 박스로 대응시켰는데, 중요한 것은 외형이 아니고 실제 환경과 유사한 운동 및 센싱이기 때문이다. 장애물 역시 4각 박스로 표현했으며 거리 측정을 위한 가상의 센서는 물리적 센서의 한계와 유사하게 시뮬레이션하였다. 초음파 센서 3개가 위치한 각도는 Quaternion 모듈의 Euler() 함수를 통해 가상 환경과 유사하게 적용하였다. 로봇의 속도도 실제 환경과 유사하게 반영하였다. 가상의 로봇도 실제 로봇과 동일하게 5개의 움직임을 가지고 있다.

(그림 4) 가상 시뮬레이션 환경

4.3 강화학습을 통한 모델 생성

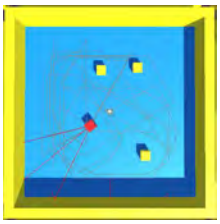
학습 모듈은 ML-Agent 0.15.1 버전을 사용하였다. 장애물은 실세계가 실시간으로 환경이 바뀐다는 가정하에 한 에피소드가 끝날 때마다 랜덤한 위치에 생성이 되어 로봇이 고정된 환경을 과적합되는 것을 방지하고 장애물에 즉각적인 반응을 보이도록 설정하였다. 사용한 강화학습 알고리즘은 PPO(Proximal Policy Optimization)이다.

강화학습을 진행하기 위해서는 로봇의 action과 reward가 필요한데, 우리의 구현은 다음과 같다. 로봇이 장애물을 만났을 경우, agent의 동작은 학습된 모델이 state를 판단하여 선택된다. 로봇이 장애물에 부딪히거나 back action을 취할 경우, reward에 (-)값을 주었으며 agent가 go action을 취할 경우, (+)값을 부여하였다. (-)값은 장애물에 부딪혔을 경우, back action을 취했을 경우의 순으로 큰 값에서 작은 값(값의 크기는 절댓값을 기준으로 비교)을 부여하였다.



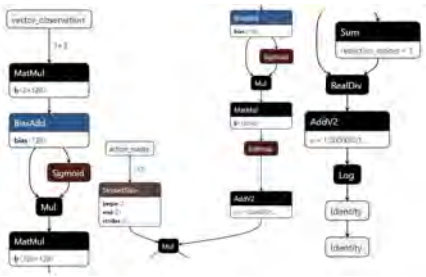

(그림 5) 가상 환경에서 시뮬레이션

가상 환경에서 로봇이 원만한 원더링을 보일 때 학습을 종료하였으며, 아래 그림은 가상 환경에서 로봇의 원더링 궤적의 예시를 보여 준다.



(그림 6) 로봇의 원더링 궤적

만들어진 모델의 신경망 구조는 아래 그림과 같다.



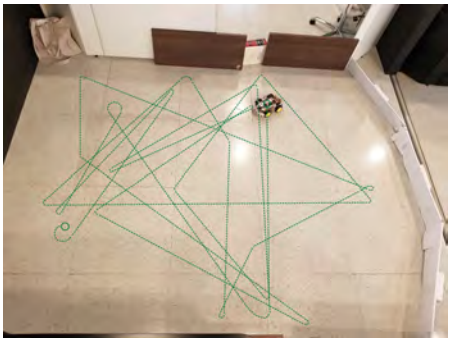

(그림 7) 생성된 모델의 신경망 구조

4.4 실세계에서 적용

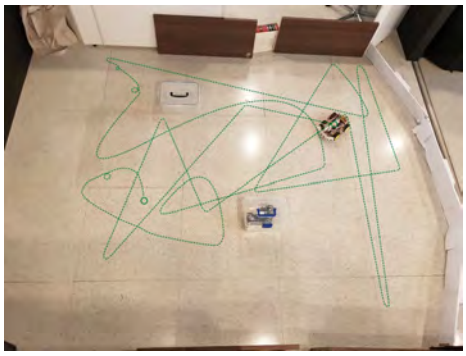
실제 환경에서 로봇은 tensorflow light 버전을 사용하기 때문에 해당 모델을 tflite 파일로 변환하는 과정이 필요하다. Edge TPU는 이렇게 변환된 모델에서 작동하게 되어 있다.

실험에서는 Unity3D의 agent와 로봇의 시간당 충돌 횟수를 비교해보아 학습이 원활히 진행되었는가와 원더링이 무사히 이뤄지고 있는가를 확인해 볼 것이다. 시간은 1분을 기준으로 한다.

해당 로봇은 가상 환경의 모델에 비하여 1분간 충돌 횟수가 평균적으로 1-2회 많은 것을 확인할 수 있었다. 따라서 실세계에서의 적용은 가상 환경보다 정확도는 떨어지지만 학습된 모델대로 동작한다는 것을 알 수 있다. 아래의 도면은 실세계에서의 로봇의 궤적을 나타낸 것이다.



(그림 8) 장애물이 없는 로봇의 궤적



(그림 9) 장애물이 있는 로봇의 궤적

## 5. 결 론

실 환경 로봇에 강화학습을 적용하기 위해서는 가상 환경 시뮬레이션을 사용한다. 우리는 Unity3D에서 제공하는 강화학습 프레임을 이용하여 실 환경에서의 로봇에 장애물을 회피하여 탐색하는 모델의 강화학습을 적용하였다.

로봇의 간단한 행동에 대한 가상 환경에서의 모델이 실제 환경에서도 원만하게 작동함을 실험으로 보였다. 이러한 시스템의 성능은 실 환경을 얼마나 유사하게 가상 환경에 모델링 하는가가 중요하며, 특히 입력되는 센서의 값이 실제 환경과 비례해야 한다. 그리고 가상 환경에서는 데이터의 노이즈가 없지만, 실제 환경에서는 일정 수준의 노이즈가 발생한다는 점을 고려해야 한다.

추후 연구 과제는 좀 더 복잡한 로봇의 문제를 이러한 플랫폼에서 쉽고 효과적으로 적용하여 학습할 수 있는지 확인하는 추가 실험이 필요하다.

## References


[1] Richard Sutton and Andrew Barto, “Reinforcement Learning: An Introduction,” 2th ed., 2017.

[2] Ahmad EL Sallab, Mohammed Abdou, Etienne Perot, and Senthil Yogamani, “Deep reinforcement learning framework for autonomous driving,” 2017.

[3] Xue Bin Peng, Glen Berseth, KangKang Yin, and Michiel Van De Panne, “Deeploco: Dynamic locomotion skills using hierarchical deep reinforcement learning,” 2017.

[4] 우주현, “Collision Avoidance for an Unmanned Surface Vehicle Using Deep Reinforcement Learning,” Ph.d. Dissertation, 서울대학교 대학원, 2018.

[5] Adam Coates, Pieter Abbeel, and Andrew Y Ng, “Apprenticeship learning for helicopter control,” 2009.

[6] 박순용, “Object-spatial layout-route-based hybrid nap and its application to mobile robot navigation,” B.S. Dissertation, 연세대학교.

[7] 조남준, “Learning, improving, and generalizing motor skills for autonomous robot manipulation : an integration of imitation learning, reinforcement learning, and deep learning,” 한양대학교 대학원.

[8] 안병규, “An Adaptive Motion Learning Architecture for Mobile Robots” 성균관대학교, 2006