

모방 학습과 계층적 구조를 이용한 시뮬레이션 기반

자율주행 자동차 학습

박동혁⁰, 주호택, 김경중*

광주과학기술원 융합기술원 융합기술학제학부

skypia0906@gist.ac.kr, ureca87@gmail.com, kjkim@gist.ac.kr

Simulation based Autonomous Driving Car Learning using Imitation Learning and Hierarchical Architecture

Dong Hyeok Park⁰, Ho-Taek Joo, Kyung-Joong Kim*

School of Integrated Technology, Gwangju Institute of Science and Technology

요 약

최근 다양한 딥러닝 모델이 공개되면서 이를 자율 주행 자동차 문제에 적용하려는 연구가 활발히 이뤄지고 있다. 대표적으로 환경과의 상호작용과 보상 함수를 통해 딥러닝 모델을 학습하는 강화학습과 전문가 데이터를 기반으로 딥러닝 모델을 학습하는 모방 학습이 있다. 모방 학습은 딥러닝 모델의 구조가 비교적 간단하고 학습이 빠르게 수렴한다는 장점이 있지만 높은 성능을 기대하기 위해선 방대한 양의 데이터가 필요하다. 또한, 다양한 상황과 행동이 존재하는 자율주행 자동차 문제에서 단일 모방 학습 모델로는 복잡한 환경의 현실 주행 문제를 해결하는데 한계가 있다. 본 연구는 이러한 문제를 해결하면서 자율주행이 가능하도록 자동차의 직진, 좌회전, 우회전과 같은 기본적인 주행 동작을 각각 독립된 딥러닝 모델로 학습시키며 상황에 따라 주행 동작을 결정하는 상위 컨트롤러 모델을 학습시켜 계층적 모방 학습 모델 구조를 구현하였다. 또한, AirSim 시뮬레이터에서 자체 제작한 맵에서 실험을 진행하며 계층적 모방 학습으로 안정적인 주행이 가능함을 확인하였다.

1. 서 론

최근 다양한 기법을 활용한 딥러닝 모델이 등장하고 이를 여러 산업에 적용하려는 시도를 볼 수 있으며, 자율주행 자동차 또한 이러한 시도들 중 하나로 많은 연구가 진행되고 있다[1]. 자율주행 자동차에 적용되는 다양한 딥러닝 모델에서는 환경과 상호작용을 통해 학습하는 강화학습 모델과 데이터를 기반으로 학습하는 모방 학습 모델이 존재한다. 강화학습 모델은 환경과 에이전트 간의 상호작용을 통해 에이전트가 환경으로부터 얻을 수 있는 보상의 합이 가장 큰 방향으로 학습이 진행된다. 하지만 정교한 보상 함수의 설계가 요구되며 학습이 어렵다는 단점이 있다. 반면에 모방 학습 모델은 전문가 데이터를 기반으로 학습을 진행하기 때문에 강화학습 모델에 비해 모델 학습이 빠르게 수렴하고 안정적인 성능을 보이지만 높은 성능을 기대하기 위해선 방대한 양의 데이터를 요구하며, 단일 모델로 다양한 상황과 이에 알맞은 행동을 학습하는데 한계가 있다. 본 연구에서는 모방 학습을 사용하며 자율주행 자동차가 가지는 다양한 행동들을 하나의 모델에서 전부 학습하지 않고 행동별로 나누어 학습하며, 상황에 알맞은 행동을 선택하는 상위 컨트롤러를 두는 계층적 구조의 모델을 제안한다.

2. 관련 연구

2.1 모방 학습(Imitation Learning)

모방 학습은 에이전트가 행동하기 위한 정책을 수립하기 위해 전문가 데이터를 사용하는 학습 방법을 의미한다. 이는 딥러닝 모델에서 주로 볼 수 있는 지도 학습(Supervised Learning)의 일환으로 지도 학습에서는 출력이 분류 등을 위한 레이블 값이지만, 모방 학습에서는 행동을 위한 액션 값이 된다. 자율주행 자동차 문제에 모방 학습을 적용하기 위해 자동차에 부착된 카메라와 다양한 센서, 행동 값들을 기록하고 딥러닝 모델에 입력으로 주어 주행에 대한 학습을 진행할 수 있다[2]. 본 연구에서는 자율주행 자동차가 주행하는데 있어 가장 기본적인 행동인 직진, 좌회전, 우회전을 나누어 각각 독립된 모델로 모방 학습을 진행하였다.

2.2 AirSim 시뮬레이터



그림 1 AirSim 시뮬레이터

* 본 연구는 UD180026RD 위탁연구의 일환으로 방위사업청과 국방과학연구소의 지원으로 수행되었음 *교신저자

AirSim 시뮬레이터[3]는 자동차 및 드론을 운용하기 위한 시뮬레이터로 자율주행 자동차 분야에서 딥러닝 모델의 학습 및 테스트 용도로 사용하는 시뮬레이터 중 하나이다. AirSim 시뮬레이터는 에이전트에 부착된 카메라 이미지, LiDAR, GPS, 속도 등 다양한 상태 정보를 에이전트에게 제공한다. 이미지는 흑백, 세그멘테이션 등 다양한 형태로 출력할 수 있으며, 에이전트에 부착할 수 있는 카메라의 위치를 자유롭게 설정할 수 있다. 또한, 언리얼 엔진을 기반으로 개발되어 사용자의 편의에 따라 기능을 추가하거나 필요한 맵을 제작하여 활용할 수 있다. 이외에도 날씨, 조도 등 기상 상황을 설정하여 다양한 상황에서 자율주행 자동차를 실험할 수 있다.

3. 연구 내용 및 방법

3.1 사용자 정의 맵 생성

AirSim 시뮬레이터는 언리얼 엔진으로 구현되어 있으며, 여러 플러그인을 추가하여 커스터마이징할 수 있다. 본 연구에서는 연구의 원활한 진행을 위해 AirSim 시뮬레이터에 내장된 지형을 사용하지 않고 RoadRunner[4]라는 플러그인을 통해 새로운 지형을 생성하였다. 생성한 지형은 다른 곡률을 가진 커브 구간과 직진, 유턴 등 학습을 위한 다양한 구간을 포함하고 있으며, 모델 학습이 원활하도록 도로 이외에 건물이나 구조물 등 다른 요소를 추가하지 않았다. 그림 2는 연구에서 사용하기 위해 제작한 맵을 2차원으로 나타낸 그림이다.

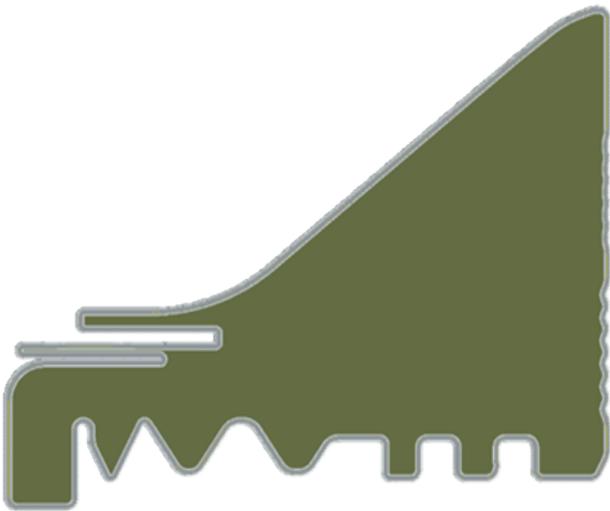


그림 2 실험에 사용한 맵

3.2 데이터 수집

모방 학습에 필요한 데이터를 수집하기 위해 본 연구에서는 드라이빙 장비인 Logitech G920을 사용하였으며, 해당 장비를 AirSim 시뮬레이터와 연동하여 데이터를 수집할 수 있도록 Pygame 패키지를 사용하였다. 데이터는 제작한 맵에서 직진 구간 6개, 좌회전 구간 24개, 우회전 구간 24개를 포함해 54개의 시작 지점을 설정하였으며, 각 시작 지점마다 10회씩 주행하여 약 12만개의 데이터를 수집하였다. 표 1은 실험에 사용한 데이터의 정보를 나타낸다.

표 1 수집된 데이터 정보

데이터	범위 및 크기	설명
Steer	[-1, 1]	에이전트의 조향 각도
Accel	[0, 1]	에이전트의 가속 페달
Brake	[0, 1]	에이전트의 브레이크 페달
Segmentation Image	(144, 256) RGB	에이전트 카메라에서 읽은 세그멘테이션 이미지

3.3 제안하는 모델

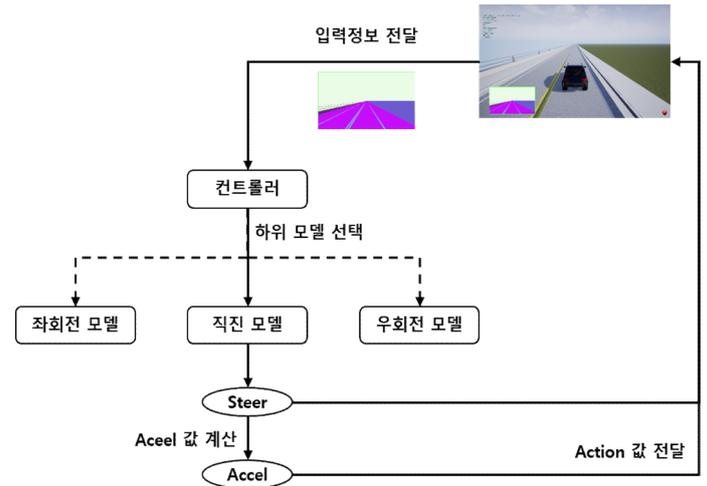
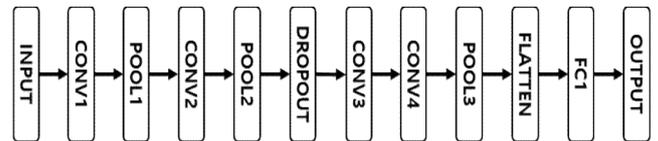
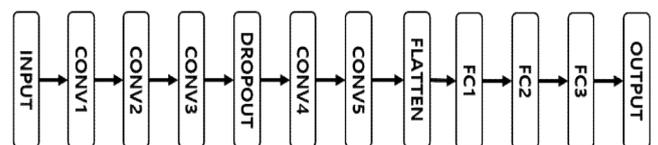


그림 3 제안하는 모델 구조

본 연구에서 제안한 모델을 그림 3과 같다. 직진, 좌회전, 우회전에 대한 행동을 담당하는 하위 행동 모델 3개와 3개의 하위 모델 중 어떤 모델을 선택할지 결정하는 상위 컨트롤러 모델 1개로 구성되어 있다. 하위 행동 모델은 수집된 데이터의 세그멘테이션 이미지를 입력으로 받아 여러 합성곱 신경망을 거쳐 출력으로 Steer 값을 출력한다. 상위 컨트롤러 모델은 하위 모델과 마찬가지로 세그멘테이션 이미지를 입력으로 받아 합성곱 신경망을 거쳐 직진, 좌회전, 우회전 중 가장 확률이 높은 한 가지로 분류된 결과를 출력하며, 하위 모델과 상위 모델 모두 손실 함수로 Mean Squared Error를 사용한다. 그림 4는 실험에서 사용된 상위 컨트롤러 모델과 행동을 담당하는 하위 행동 모델의 네트워크 구조이다.



상위 컨트롤러 모델 구조



하위 행동 모델 구조

그림 4 상위 및 하위 모델 네트워크 구조

상위 컨트롤러 모델과 하위 행동 모델을 학습하기 위해 사전에 제작한 맵을 주행하며 수집한 세그멘테이션 이미지 및 행동 값들을 사용하였다. 이미지를 처리하기 위한 합성곱 신경망에 이미지를 입력으로 넣기 전에 RGB 픽셀 값을 0 ~ 1 값으로 조정하는 정규화 작업을 진행하였으며, 이미지 전체를 사용하지 않고 이미지에서 필요한 부분을 잘라내어 사용하였다. 또한, 상위 컨트롤러 모델 학습에서 직진, 좌회전, 우회전에 해당하는 세그멘테이션 이미지 데이터 중 무작위로 각각 500개씩 선택하여 상위 컨트롤러 모델의 학습을 진행하였다.

모델	데이터 종류	데이터 사용 수
상위 컨트롤러 모델	이미지	1,500 개
직진 행동 모델	이미지, 행동 값	20,100 개
좌회전 행동 모델	이미지, 행동 값	44,539 개
우회전 행동 모델	이미지, 행동 값	55,008 개

표 2 모델 학습에 사용된 데이터

평가 단계에서는 학습이 완료된 상위 컨트롤러 모델에서 출력한 결과에 따라 직진, 좌회전, 우회전 하위 행동 모델이 선택되며, 선택된 하위 행동 모델은 입력으로 받아오는 이미지에 따라 Steer 값을 출력한다. Accel 값은 하위 행동 모델에서 출력된 Steer 값에 따라 아래 수식에 의해 결정되어 에이전트를 동작시킨다.

$$accel = (0.8 - 0.4 * |steer|) * 0.85$$

4. 실험 결과 및 향후 연구

실험에서는 제작한 맵에서 학습이 완료된 계층적 모방 학습 모델이 전문가와 얼마나 유사한 행동을 보이는지 측정하였다. 그림 5와 그림 6은 제작한 맵의 90도 커브 구간을 비롯해 다양한 곡률의 커브 구간에서 AirSim 시뮬레이터 상에서 학습된 제안 모델과 전문가가 주행하며 발생한 Steer 값의 분포를 비교한 것이다. 그림 5와 그림 6에서 볼 수 있듯이 전문가 데이터와 계층적 모방 학습 모델에서 나온 행동 패턴이 유사한 것을 확인할 수 있다. 본 연구의 시뮬레이션 동영상은 유튜브에서 확인할 수 있다[5].

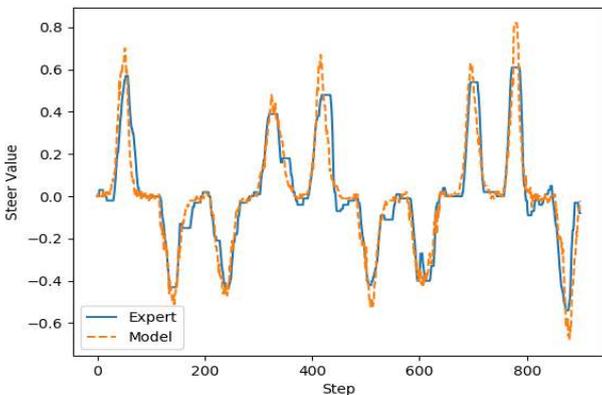


그림 5 90도 커브 구간에서의 Steer 값 비교

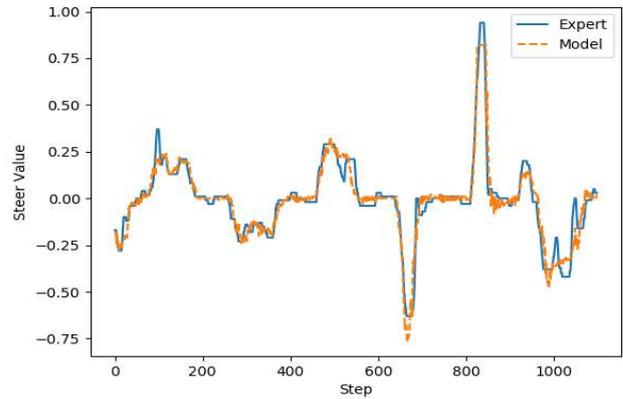


그림 6 다른 커브 구간에서의 Steer 값 비교

향후 연구로는 현재 행동을 담당하는 하위 모델에서 Steer 값만 출력하며, Steer 값에 따라 Accel 값이 결정되기 때문에 속도가 느린 단점이 있다. 따라서 이러한 문제를 해결하기 위해 딥러닝 모델에서 Accel과 Brake 값을 함께 출력하도록 하여 상황에 따라 속도를 유연하게 조절할 수 있도록 딥러닝 모델을 개선할 예정이다. 또한, 현재 실험을 위해 제작한 맵은 1차선 도로이며 다른 차량이나 신호등, 보행자 등 다른 요소를 고려하지 않아 현실 문제에 적용하는데 한계가 있다. 향후 연구에는 보행자와 차량, 교통 시스템 등 다양한 요소를 추가하여 연구를 진행할 예정이다.

참고문헌

[1] 이병윤. “국내외 자율주행자동차 기술개발 동향과 전망.” 한국통신학회지 (정보와통신) 33.4 (2016) 10-16.
 [2] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba. End to End Learning for Self-Driving Cars, Technical Report, .http://arxiv.org/abs/1604.07316, 2016
 [3] Arisim, https://microsoft.github.io/AirSim/
 [4] RoadRunner, https://www.vectorzero.io/
 [5] DongHyeok Park, “Hierarchical IL - AirSim”, https://youtu.be/0FmKCV0dEng