임베디드 강화학습을 이용한 험지 자율주행

문재영○, 전현창○, 김경중*
광주과학기술원 융합기술원 융합기술학제학부
supermoon@gm.gist.ac.kr, kevinjeon119@gm.gist.ac.kr, kjkim@gist.ac.kr

Embedded Reinforcement Learning for Offroad Self-Driving

Jae-Young Moon○, Hyeon-Chang Jeon○, Kyung-Joong Kim*
School of Integrated Technology, Gwangju Institute of Science and Technology

요 약

대부분의 자율주행 연구는 도로와 비도로의 구분이 명확한 환경에서 진행한다. 하지만 실제 환경에서는 자동차가 비포장도로를 주행해야 하는 경우도 많다. 본 논문에서는 실제 모형 차를 이용하여, 도로의 구분이 확실하지 않은 야지에서의 강화학습 자율주행 실험 연구를 기술한다. 모든 학습은 차체 내에서 이루어지며, 주행 학습 도중, 사람은 상호작용을 통해 차량이 받는 보상에 영향을 줄 수 있다. 약 10m 거리의 도로 구분이 없는 야지 환경을 막힘없이 주행하는 것을 목표로, DDPG, SAC, PPO 세 가지 강화학습 알고리즘을 실제 모형차에서 학습하여 각 알고리즘의 성능 및 학습 패턴을 분석하였다. 실험분석은 각 알고리즘의 소요 학습 시간과 학습 결과(총 이동 거리)를 기준으로 진행하였으며, 알고리즘의 특이 학습 패턴 및 장단점을 함께 기술하였다.

1. 서 론

도로 위에서의 자율주행에 대한 연구는 이제 상용화 단계에 접어들었다. 도로를 정복하고 나면 다음은 어디일까? 연구실을 떠나 실제 사람들이 운전하는 장소를 보면, 흔히들 '오프로드'라고 부르는, 도로와 비도로의 경계가 불분명한 길을 운전하는 경우가 많다. 이를 본 논문에서는 야지 또는 험지라고 부른다. 이러한 험지 환경은 차선 추적(Lane Tracing)이 어렵고, GPS 정보를 얻기 어려운 경우가 많으며, 지도 상에 나타나지 않은 길일 가능성이 크다.

험지 환경에서의 자율주행을 위해서는 자율주행 에이전트가 주행 가능한 길을 탐색하고, 주변 사물을 피하며 주행할 수 있도록 학습해야 한다. 본 논문에서는 야지 환경에서 최대한 빠른 속도로 주행하는 자율주행 자동차를 학습하는 것을 목표로 실험을 진행했다. 이를 위해 전통적인 이미지 프로세싱 및 센서 제어 방법부터 모방학습 등의 다양한 딥러닝 방법들이 적용될 수 있겠지만, 본 논문은 강화학습 적용에 초점을 두었다. 강화학습 적용 이유는 다음과 같다.

- 전통적인 방법들은 개발 난이도 및 복잡도에 비해 높은 성능을 기대하기 어렵다.
- 모방학습은 학습 안정성이 크지만, 학습을 위해 수집한 데이터 이상의 성능을 기대하기 어렵다.
- 강화학습은 학습 목적에 따라 보상 함수를 설계하여 더욱 폭넓고 다양한 문제를 학습 할 수 있으며, 보상 함수를 적절히 설계할 경우 성능 향상을 기대할 수 있다.

본 논문에서는, 실험을 위해 시뮬레이션을 사용하는 대신, 실제 모형차에 학습 모델을 올려 실시간으로 학습을 하는 임베디드 강화학습을 진행하였다. 임베디드 강화학습 방식을 적용한 데에는 두 가지 이유가 있다. 첫째, 아직까지 자율주행 연구를 위한 시뮬레이션 프로그램들에는 험지 환경의 맵이 없고, 커스텀으로 만든다고 하더라도 실제 험지 환경의 복잡도를 구현해내기 어렵다. 둘째, 명확하게 길이라고 표현된 도로가 없기 때문에 임의의 경로를 벗어났을 때, 경로 이탈에 대한 시그널을 주기가 어렵다.

본 논문에서는 DDPG[1], SAC[2], PPO[3] 총 세 가지 강화학습 모델을 모형차에서 실시간으로 학습하며 약 10m 거리의 야지 환경을 주행하는 동안 기록된 로그를 바탕으로 각 모델 별 특징들을 비교 분석한다.

2. 관련 연구

2.1 임베디드 강화학습

강화학습은 데이터를 모아 한번에 학습하는 지도학습 모델들과 달리 인공지능 에이전트가 환경과 소통하며 시행착오(trial and error)를 겪으면서 얻은 보상을 통해 학습을 진행한다. 일반적으로 간단한 강화학습 에이전트라도 목표한 성능까지 학습하기 위해서는 최소 수십만 스텝의 학습이 필요하다. 따라서 강화학습에는 학습 시나리오를 빠르게 시뮬레이션하는 것이 중요하다.

하지만 시뮬레이션을 활용하기 어렵거나, 현실과의 소통이 불가피한 환경일 경우, 실제 환경 속에서 학습하는 임베디드 강화학습을 진행해야 한다. WAYVE 사에서는 사람이 직접 자율주행 자동차에 타서 위급 상황 시 긴급 정지와 함께 사람이 직접 운전을 조작하는 식의 방법으로 자율주행 강화학습을 성공시켰다[4]. 조지아 공대에서 발간한 논문[5]에서는 직접 임베디드 시스템에서의 강화학습을 위한 별도의 하드웨어를 직접 제작하여 임베디드 강화학습을 진행했다.

오프로드 자율주행은 특성상 학습 과정에서 인간의 개입이 필요하기 때문에 상기된 빠른 시뮬레이션에 대한 필요성이 낮다. 본 논문에서는 실제 환경에서 학습을 진

* 교신저자

행하는 임베디드 강화학습을 진행했으며, 별도의 하드웨어를 구성하지 않고 라즈베리파이 4 보드 위에서 모델을 학습했다.

3. 험지 자율주행을 위한 임베디드 강화학습

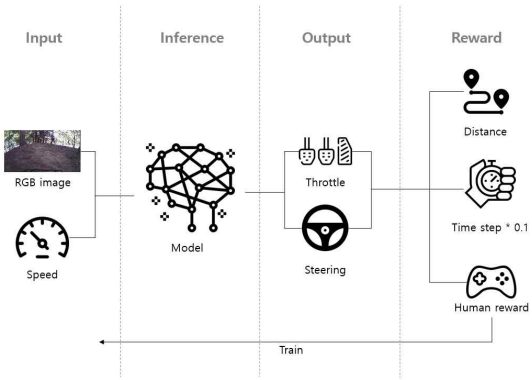

그림 1 학습 모델 프로세스

3.1 자율주행용 강화학습 실험 모델

본 연구에서는 총 3가지(DDPG, PPO, SAC) 모델들을 실험했다. 먼저 DDPG(Deep Deterministic Policy Gradient)[1]는 대표적인 Off-policy 강화학습 알고리즘으로, 기존 DQN(Deep Q-Network)에서 반영하지 못했던 연속적인 행동 공간에서 학습이 가능하다. DDPG를 사용하여 물리엔진 Mujoco와 자율주행 시뮬레이터 Torcs에서 학습이 성공적으로 이루어졌었기에 실제 환경에도 적용을 시도했다. SAC(Soft Actor-Critic)[2]는 DDPG와 마찬가지로 Off-policy 계열의 알고리즘이다. SAC는 DDPG와 달리 확률적인 행동모델을 가지고 있으며 고차원의 문제를 풀기 더 적합하다는 강점을 가지고 있다. 마지막으로 PPO(Proximal Policy Optimization)는 앞선 알고리즘들과는 달리 On-policy 계열의 알고리즘이며 구현하기 쉽고 실제 적용시 가장 성능이 잘 나오는 실용적인 알고리즘이기 때문에 본 실험을 위한 알고리즘으로 추가하였다.

3.2 보상함수 설계

학습을 위해, 모든 모델에는 동일한 입/출력과 보상함수를 적용하였다(그림 1). 모델의 입력으로는 웹캠과 로터리 엔코더로부터 받아온 160×120 RGB이미지와 현재 속력이 사용되었으며, 스로틀(Throttle)과 핸들(Steer)값이 출력으로 나온다. 보상함수는 아래 수식과 같다.

$$Reward = Distance - TimeStep \times 0.1 + HumanReward$$

이동거리를 meter 단위로 양의 보상을 주며, 각 타임스텝 별로 최소 10cm 이상 전진하도록 하기 위해 타임스텝에 0.1을 곱한 값을 음의 보상으로 주었으며, 추가적으로 사람의 개입을 보상으로 넣었다. 사람의 개입이란, 사람이 모형차의 주행을 관찰하면서 예상경로를 벗어날 경우 버튼을 눌러 -100의 음의 보상을 주거나, 목적지에 잘 도착했을 경우 성공 버튼을 눌러 +100의 양의 보상을 주는 것을 말한다.

4. 실험 결과 및 분석

4.1 자율주행 모형차

실험에 사용할 모형차(그림 2-가)의 프레임, 12T DC모터, servo모터, ESC, PWM 컨트롤러 등은 Traxxas사의 Slash VR46 모델의 부품을 사용했다. 주행 연산은 모형차에 라즈베리파이 4 보드를 연결하여 자율주행을 위한 계산을 진행했다. 주행 연산 및 보상함수에 필요한 현재 속도, 이동 거리 등을 측정하기 위해 자동차 후미에 KY-040 로터리 엔코더를 장착하였으며, 카메라는 일반 웹캠을 사용하였다. 모형차 제어를 위해서는 Donkey car 시스템을 사용했다. Donkey Car[6] 시스템은 모형차의 입출력 제어와 주행을 위한 알고리즘 실행을 연결시켜주는 인터페이스이다. Donkey Car에서 제공하는 AI 주행 라이브러리가 있지만, 지도학습 방법만 제공하고 있어서 강화학습을 위한 gym 환경 및 학습 알고리즘은 직접 제작하였다.

Donkey Car 시스템은 자동차 객체에 여러 가지 part 객체들을 추가하여 각 part들이 병렬적으로 동작하도록 자동차 객체가 관리하는 구조이다. 강화학습은 주행 도중 환경과 소통하면서 출력을 추론하는 동시에 출력을 바탕으로 학습하는 구조를 갖고 있다. 주행 중 학습이 이루어지면 이전 타임스텝에 들어간 출력이 계속 Throttle part 내에 남아 있어 오작동을 일으킬 수 있다. 따라서 이러한 병렬 프로세싱을 고려하여 gym환경 및 학습 프로세스를 구축하였다.


(가) (나)

그림 2 험지 환경 실험을 위한 Donkey Car 모형차(가)와 실제 야지 환경(나)

4.2 실험 모델 구조

전체적인 모델들의 기본 구조는 이미지를 받아서 특징벡터 형태로 만들어주는 인식 모델(Perception), 이 후, 에이전트의 행동 가치를 평가해주는 비평가 모델(Critic), 마지막으로 실제 행동을 결정하는 배우 모델(Actor)이 있으며 들어온 이미지를 인식 모듈이 전처리 한 후, 각각 비평가 모델, 배우 모델에 들어가 학습을 진행하는 구조다. DDPG는 인식 모델, 비평가 모델, 배우 모델 하나씩 가지고 있는 기본적인 네트워크 구조 형태를 취하고 있다. PPO 역시 모델은 DDPG와 같이 하나씩 가지고 있지만, 학습을 하는 과정 자체가 주행을 진행하는 도중에 멈추고 학습을 한다는 점에서 차이가 있다. 마지막으로 SAC는 두 개의 비평가 모델을 두어 둘 중 더 작은 손실값을 주는 모델의 손실 값을 바탕으로 학습을 하는 구조로 되어 있다.

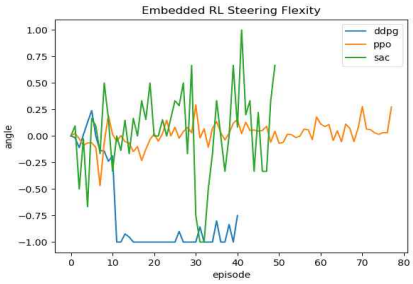

그림 3-가

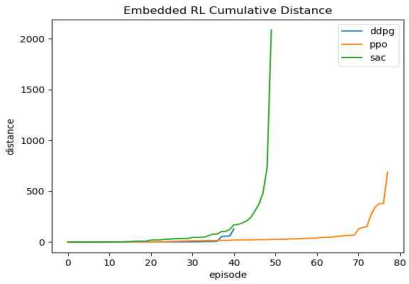

그림 3-나

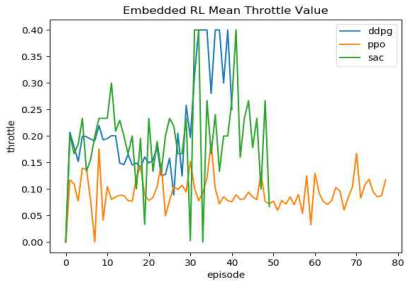

그림 3-다

그림 3 DDPG, PPO, SAC 알고리즘 에피소드 진행에 따른 실험 결과. 왼쪽부터 평균 핸들값(가), 주행 거리(나), 평균 가속도 값(다)

4.3 험지 강화학습 실험 및 분석

본 절은 각 모델을 적용하여 실제 험지에서 온라인으로 임베디드 강화학습을 진행한 결과를 분석한다.

첫 번째로, 도로 등의 객체 인식이 불가능한 상황에서의 핸들(Steer)의 편향성을 분석했다(그림 3-가). 실험했던 실제 험지 환경에서의 길은 직진이기 때문에 핸들에 대한 평균 값은 0에 수렴할수록 학습이 잘 되었다고 판단할 수 있다. PPO는 On-policy 방법이기 때문에, 에피소드 중간에 학습을 진행하는 방식으로 인해서 다른 알고리즘보다 학습이 빠르게 수렴되며 안정적으로 주행하는 양상을 보인다. SAC는 핸들값을 평균하면 직진 경로로 주행하는 것을 알 수 있지만, PPO에 비해 분산이 매우 큰 모습을 보였다. 마지막으로 DDPG는 학습 중 핸들값이 편향되어 한쪽으로 치우쳐 충돌을 일어나는 경우를 확인했으며, 이는 출력 값이 TD(Temporal Difference) 에러가 누적됨에 따라서 잘못된 방향으로 핸들이 학습되었기 때문으로 보인다.

두 번째로, 실험 모델들에 대하여 부착된 로터리 엔코더 센서를 통해 기록한 누적 거리를 분석했다(그림 3-나). PPO는 핸들과 달리 스로틀(Throttle) 학습 속도는 다소 느린 것을 확인할 수 있었다. SAC는 핸들이 다소 불안정함에도 불구하고 학습이 진행됨에 따라서 장애물을 피하며 주행을 하는 모습을 확인하고 이에 따라서 주행 거리도 늘어남을 확인했다. DDPG는 핸들의 편향성으로 인해 속도와 별개로 계속 충돌이 진행되었으며, 이로 인해 주행 거리 역시 짧았다.

마지막으로, 학습 에피소드별 알고리즘들의 평균적 속도를 비교했다(그림 3-다). PPO는 안정적인 속도를 유지하며 지속적으로 주행을 했지만, 장애물이 다소 높을 경우, 가속을 주지 못해 넘지 못하는 현상이 발생하기도 했다. SAC는 에피소드에 따라서 평균적인 가속도 값은 다소 불안정하지만, 전반적으로 PPO보다 높은 상태를 유지하며 주행을 하는 모습을 확인 할 수 있다. DDPG는 학습이 진행됨에 따라서 속도가 많이 올라감을 확인할 수 있었지만, 앞서 언급했듯이 핸들값의 편향성 문제를 가지고 있었다.

5. 결론 및 향후 연구

본 연구에서는 기존 시뮬레이터 기반 강화학습이 아닌 차체에서 직접 학습을 진행하는 임베디드 강화학습에 대한 알고리즘 실험을 진행했다. PPO와 SAC는 각각 안정성, 학습 속도라는 측면에서 장점을 갖고 있었으며 DDPG는 핸들의 과측정 문제로 인한 과제가 남았다.

향후 연구에서는 DDPG의 과측정 문제를 해결하기 위해 제안된 TD3 알고리즘을 적용할 예정이다. 이 밖에도 사람이 자체적으로 판단하며 주는 보상 체계 자체는 시간적 비용이 너무 많이들고, 비효율적임을 파악하였으며 실험 중 시간의 흐름에 따른 조광의 차이로 인하여 입력 이미지의 피처가 심각하게 손상되는 것을 확인했다. 이러한 문제들을 해결하기 위해 강력한(Robust) 입력 처리 및 Sim2Real에 대한 연구를 진행하여야 한다.

감사의 글

본 연구는 국방과학연구소의 지원(UD180026RD)으로 '최신의 강화학습 기반 경로 계획 기술' 위탁 연구에서 수행되었습니다.

이 논문은 2020년도 광주과학기술원의 재원으로 글로벌 선도대학 육성 사업의 지원을 받아 수행된 연구임

참고문헌

[1] Lillicrap, Timothy P., *et al.* Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971, 2015.

[2] Harrnoja Tuomas, *et al.* Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. arXiv preprint arXiv:1801.01290, 2018.

[3] Schulman, John, *et al.* Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347, 2017.

[4] Alex Kendall, *et al.* Learning to Drive in a Day. arXiv preprint arXiv:1807.00412, 2018.

[5] Anvesha Amravati, *et al.* A55nm Time-Domain Mixed-Sigmal Neuromorphic Accelerator with Stochastic Synapses and Embedded Reinforcement Learning for Autonomous Micro-Robots, ISSCC, 7, 2018.

[6] Donkey Car, http://docs.donkeycar.com/