Atari Deep Q Network Model을 이용한 장애물 회피에 특화된 실내 자율주행 적용에 관한 연구

백지훈*, 오현택*, 이승진*, 김상훈*
*한경대학교 전기전자제어공학과
e-mail: wind1104@hanmail.net

A Study about Application of Indoor Autonomous Driving for Obstacle Avoidance Using Atari Deep Q Network Model

Ji-Hoon Baek*, Hyeon-Tack Oh* Seung-Jin Lee*, Sang-Hoon Kim*
*Dept of Electrical, Electronic and Control, Hankyong National University

요 약

최근 다층의 인공신경망 모델이 수많은 분야에 대한 해결 방안으로 제시되고 있으며 2015년 Mnih이 고안한 DQN(Deep Q Network)는 Atari game에서 인간 수준의 성능을 보여주며 많은 이들에게 놀라움을 자아냈다. 본 논문에서는 Atari DQN Model을 실내 자율주행 모바일 로봇에 적용하여 신경망 모델이 최단 경로를 추종하며 장애물 회피를 위한 행동을 학습시키기 위해 로봇이 가지는 상태 정보들을 84*84 Mat로 가공하였고 15가지의 행동을 정의하였다. 또한 Virtual world에서 신경망 모델이 실제와 유사한 현재 상태를 입력받아 가장 최적의 정책을 학습하고 Real World에 적용하는 방법을 연구하였다.

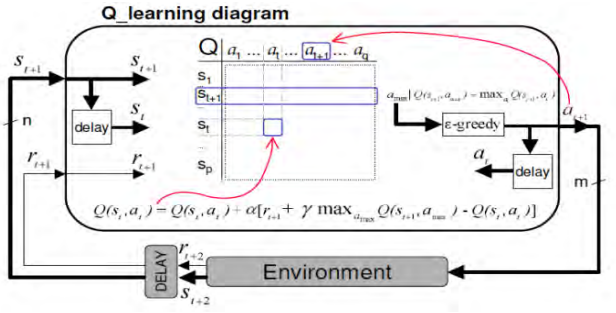
1. 서론

최근 뇌의 생물학적인 신경연결망을 모방한 다층의 인공세포와 연결망으로 구성된 심층 신경망과 연결망의 강도(Weight)를 학습시키는 딥 러닝 방법을 이용한 심층 신경망 모델이 수많은 분야에 대한 해결방안으로 제시되고 있다. 그 가운데 Mnih et al (2015)이 고안한 강화학습의 한 방법인 Q-Learning을 심층신경망에 적용시켜 학습시킨 DQN(Deep Q Network)는 Atari의 49개의 게임에 대한 결과 타 알고리즘보다 43개의 게임에서 우위를 차지하였고 29개의 게임에서는 human tester 점수의 75% 이상을 얻어 인간 수준의 제어 능력을 보여주었다[1]. 본 논문은 이 기술을 모바일 로봇에 접목시켜 심층 신경망이 학습을 통해 실내 환경에서 장애물 회피에 특화될 수 있도록 유도시키기 위한 연구의 과정이다. 이 연구에서 심층 신경망의 입력 데이터는 Lidar의 2D 스캔을 통한 주변 장애물 거리 정보, ROS의 Gmapping 패키지를 이용하여 미리 작성된 실내 지도에서 로봇의 위치 및 자세 정보, Dijkstra algorithm을 이용한 지도상의 출발지와 목적지에 대한 최단거리 정보를 사용하였다. 이를 통해 모델은 학습으로 업데이트된 연결망의 강도에 따라 입력 데이터에 대해 장애물 회피를 위한 최적의 행동을 취할 수 있도록 신경망 모델링 및 학습 방법에 대해 연구를 진행하였다.

2. 본론

2.1 강화학습

지도 학습(Supervised learning)의 경우 학습 단계에서 많은 입력 데이터와 그에 대한 정답(label)을 필요로 하지만 모델이 상태에 따른 라벨이 붙은 학습 데이터를 만든다는 것은 의미 없는 반복적인 작업이 요구되고 연구자의 성향에 의해 Labeling된 정답이 옳다고 단정 지을 수 없기 때문에 적합하지 않다. 하지만 강화 학습(Reinforcement learning)은 학습자가 능동적으로 환경에 대한 행동을 하고, 이 행동에 대한 평가를 받아 학습이 이루어 진다[4]. 따라서 연구자는 입력 데이터와 몇 가지의 선택지, 그리고 규칙을 통한 보상을 정의하여 데이터 셋을 만들 필요가 없고 연구자의 개입이 지도 학습 방법에 대해 비교적 적은 강화 학습 방법을 이용하여 심층 신경망 모델의 연결망의 강도를 학습시키는 방법을 선택하였다. 본 논문에서는 강화 학습 방법 중 Watkins and Dayan,(1992)에 의해 제안된 TD기법 기반 알고리즘으로 행동가치함수 Q^{π}를 이용하여 최적 정책π^{*}를 찾는 기법인 Q-Learning[2]이 심층 신경망에 적용된 Deep Q-Learning 방법을 통해 모델을 학습시키고자 한다.

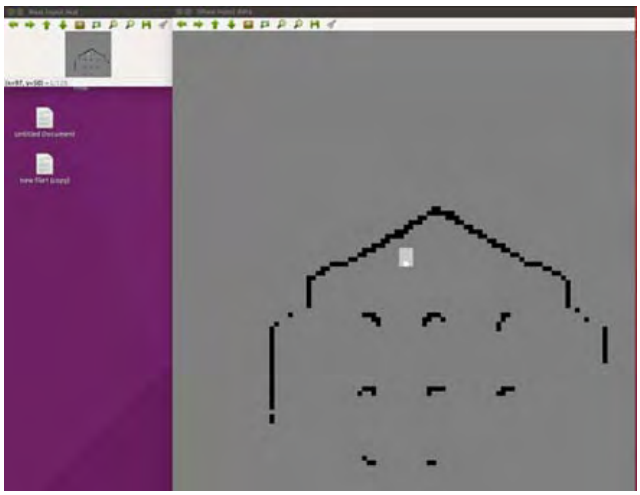


(그림 1) Diagram of tabular Q learning algorithm. [3]

2.2 Neural network architecture

인공 신경망의 구조는 Mnih et al (2015)이 고안한 Q-Learning에 신경망이 결합된 DQN구조를 참고하여 사용하였다[1].

2.2.1 Input layer



(그림 2) 이미지 형태로 표현된 입력데이터

입력 데이터는 앞서 언급한 장애물, 위치, 경로 데이터를 토대로 하여 그림 1의 오른쪽과 같이 데이터들을 동시에 나타내는 픽셀당 1byte, 84*84 배열로 가공하였고 OpenCV Library를 이용하여 시각화 하였다. 픽셀 한칸당 8.333cm의 거리를 의미하고 로봇의 Lidar는 중앙에 고정되어 있어 심층 신경망 모델은 항상 주변 3.5m까지의 장애물 정보를 인식할 수 있다. Lidar 센서의 성능에 따라 더 먼 거리까지 인식이 가능하지만 입력 데이터의 크기에 따라 연산의 속도가 저하될 수 있고, 거리 정보의 분해능에 따라 심층 신경망의 인식 정밀도가 떨어질 수 있으므로 신경망이 부분적인 관찰을 할 수 있도록 하였다. 그리고 모델에게 시간적 의존관계를 학습시키기 위하여 현재 시점(t)으로 부터 2번째 전 시점(t-2)까지 3장의 프레임을 입력 데이터로 하여 84*84*3 형태의 Input layer를 설계하였다.

2.2.2 Output layer

모바일 로봇은 전 방향 이동이 가능한 메카넘 휠을 이용하여 설계하였고 그에 따라 아래와 같이 15개의 행동을 정의하였다. 따라서 Output layer로 15개의 노드를 설계하여 15개의 동작에 대한 $Q(s,a)$값(상태s에서 행동a를 취하였을 때 얻는 가치)에 대해 Softmax function을 이용해 출력 값들을 0~1사이의 값으로 normalize하여 가장 높은 값을 갖는 액션 노드를 선택하도록 하였다.

(그림 3) 자체 제작한 전방향 이동 모바일 로봇

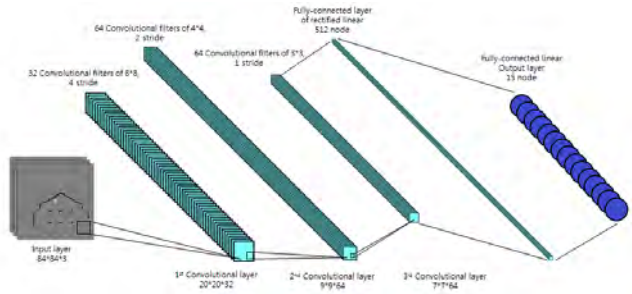
전방 좌회전	좌측 대각 전진	전진	우측 대각 전진	전방 우회전
제자리 좌회전	좌측 이동	정지	우측 이동	제자리 우회전
후방 좌회전	좌측 대각 후진	후진	우측 대각 후진	후방 우회전

<표 1> 모바일 로봇 액션 선택지

2.2.3 Hidden layer

DQN은 입력단에 합성곱층이 포함되어 있는데, 이는 이미지 픽셀에 대한 합성곱 연산을 수행하여 이미지 각 영역의 관계를 파악하여 영상정보를 처리하기 위함이다. Hidden layer는 다음과 같이 설계되었다.

1st layer : 32 convolutional filter of 8*8, 4 stride, 20*20*32
2nd layer : 64 convolutional filter of 4*4, 2 stride 9*9*64
3rd layer : 64 convolutional filter of 8*8, 1 stride 7*7*64
4th layer : Fully-connected layer of rectified linear 512 node
5th layer : Fully-connected linear Output layer 15 node



(그림 4) 네트워크 구조

2.3 Reward 정의

Reinforcement learning 방법으로 신경망의 연결망 강도를 업데이트 하는 것은 supervised learning에서 정답에 대한 Label을 이용하는 것과 달리 행동에 따른 보상을 이용하여 역 전파를 통해 각각의 가중치를 Update시킨다. 따라서 Reward를 어떻게 정의하느냐에 따라 신경망 모델의 학습 결과가 달라질 것이다. 따라서 본 논문에서는 다음의 3가지에 대한 리워드를 정의하였다.

$$Reward_{tot} = Reward_1 + Reward_2 + Reward_3$$

2.3.1 경로 추종에 따른 양의 보상

먼저 Dijkstra algorithm을 통한 최단 거리를 추종하도록 모델을 학습시키기 위하여 현재 로봇의 위치(R_t)와 이전 프레임에서의 로봇의 위치(R_{t-1}), 현재 로봇이 바라보는 방향(θ_{R_t}), 전체 경로 중 1m 거리의 경로 포인트(P_{1m}), 로봇에서 P_{1m} 까지의 방향($\theta_{P_{1m}}$)에 대한 보상을 다음과 같이 정의하였다.

$$Reward_1 = \frac{R_t - R_{t-1}}{P_{1m} - R_{t-1}} \times \frac{\theta_{R_t}}{\theta_{P_{1m}}}$$

2.3.2 장애물 충돌에 따른 음의 보상

로봇이 주행을 하며 장애물과 충돌하는 상황은 절대적으로 피하기 위하여 주변 스캔 거리의 값 가운데 가장 작은 값($Scan_{\min}$)이 0.3m 이하일 경우 아주 큰 패널티를 부여하고 모든 상태를 초기화 하도록 설계하였다.

$$Reward_2 = -100 \quad (Scan_{\min} \le 0.3)$$

2.3.3 시간에 대한 음의 보상

시간에 따라 패널티가 존재하지 않는다면 신경망 모델은 계속해서 정지 행동을 취하여 상태를 유지하려 경향을 보인다. 따라서 매 입력마다 계속해서 패널티를 부여하여 이를 방지하기 위해 다음과 같이 음의 리워드를 정의하였다.

$$Reward_3 = -0.1$$

2.4 Replay Memory, Target Q-Network

첫 번째로 강화학습 방법에서 에이전트는 시간의 흐름에 따라 순차적으로 학습 데이터를 수집하게 된다. 이 때 데이터가 순차적으로 수집되어 높은 correlation에 의해 학습이 불안정해 질 것이다. Nature에 등재된 GoogleDeepmind의 논문에 따르면 Atari 게임 환경에서 학습을 시킬 때 입력 데이터간의 Correlation을 줄이기 위해 아래와 같이 Agent의 경험들을 Replay memory에 저장한다.

$$e_t = (s_t, a_t, r_t, s_{t+1}),\ s: \text{상태},\ a: \text{액션},\ r: \text{리워드}$$

그 다음 학습을 할 때 Replay memory로부터 Random sampling을 통해 미니뱃치를 구성해서 학습을 진행한다. 이 방법을 통해 학습을 할 때 데이터가 순차적이지 않게 구성할 수 있어 입력 데이터 간의 Correlation을 줄일 수 있다.

두 번째로 Non-stationary targets 문제를 완화하기 위하여 Q-network와 같은 구조이지만 별도의 파라미터를 가진 Target network를 만들어 일정 스텝마다 Target network의 파라미터를 Q-Network의 파라미터로 업데이트 시킨다. 이 방법을 통해 Q-Network 업데이트 타겟이 움직이는 현상을 방지할 수 있다. 아래 그림5는 GoogleDeepmind에서 Replay memory size를 1,000,000으로 놓고 Target network update step을 10,000으로 설정한 성능 비교 실험이다[1].

Extended Data Table 3 | The effects of replay and separating the target Q-network

Game	With replay, with target Q	With replay, without target Q	Without replay, with target Q	Without replay, without target Q
Breakout	316.8	240.7	10.2	3.2
Enduro	1006.3	831.4	141.9	29.1
River Raid	7446.6	4102.8	2867.7	1453.0
Seaquest	2894.4	822.6	1003.0	275.8
Space Invaders	1088.9	826.3	373.2	302.0

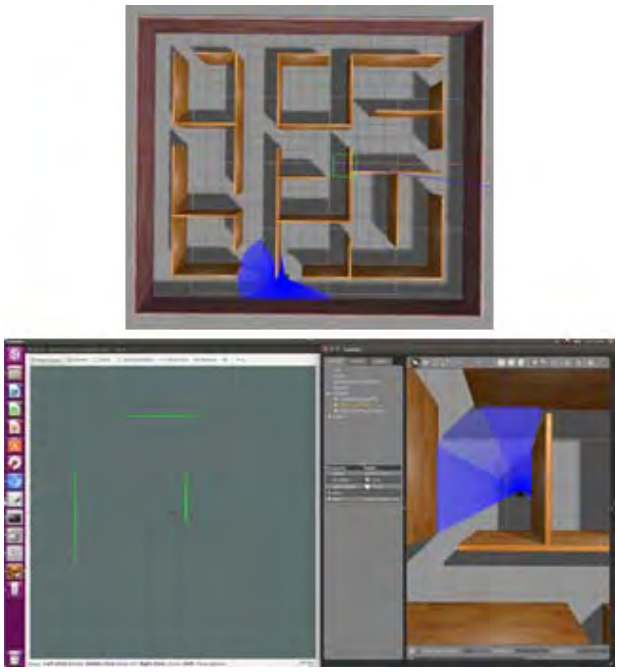
DQN agents were trained for 10 million frames using standard hyperparameters for all possible combinations of turning replay on or off, using or not using a separate target Q-network, and three different learning rates. Each agent was evaluated every 250,000 training frames for 135,000 validation frames and the highest average episode score is reported. Note that these evaluation episodes were not truncated at 5 min leading to higher scores on Enduro than the ones reported in Extended Data Table 2. Note also that the number of training frames was shorter (10 million frames) as compared to the main results presented in Extended Data Table 2 (50 million frames).

<표2> Replay memory, Target Q-Network 유무에 따른 퍼포먼스 차이 비교[1]

위 실험에서 Replay memory, Target Network 사용 유무에 따라 각각의 게임에서 에피소드에서 얻는 점수가 크게 차이가 나는 모습을 볼 수 있었다. 따라서 본 논문에서는 로봇의 임베디드 보드 메모리 용량을 고려하여 Replay memory size를 1/10 가량 줄여 사용하였다.

2.5 가상 학습 환경 조성

신경망 모델을 강화학습 방법으로 Real world에서 학습시킨다는 것은 엄청나게 고된 반복적인 노동을 필요로 하여 사실상 불가능에 가깝다. 따라서 본 논문에서는 가상 환경 시뮬레이터 gazebo를 이용하여 가상의 환경 속에서 신경망 모델이 학습을 할 수 있도록 환경을 조성하여 학습하도록 하였다. 또한 단계적인 학습을 통해 안정적인 학습을 할 수 있도록 신경망 모델의 학습 정도에 따라 환경의 난이도를 높여가며 학습을 진행할 수 있도록 단계를 나눠 학습이 진행되도록 하였다.



(그림 5) Gazebo상의 가상 학습 환경

3. 결론

로봇의 Lidar를 통한 2D Scan값, 관성센서를 이용한 자세 정보, 지도로부터 출발지에서 목적지까지의 Dijkstra algorithm 정보를 토대로 Atari DQN 모델에 사용된 입력 레이어의 크기와 같은 84*84 크기의 입력 이미지를 만들었고, 다양한 행동이 가능하도록 15개의 출력 층을 정의하였다. 그리고 보상을 정의하여 로봇이 경로를 추종하며 장애물에는 부딪히지 않고, 시간의 경과에 따른 음의 보상을 줘서 프리징 상태에 빠지는 것을 방지하였다. 이후 DQN에서 행동에 대한 보상에 대해 행동가치함수를 이용하여 최적의 정책을 찾을 수 있도록 수많은 학습을 반복한다면 로봇이 주어진 상태에 대해 장애물을 회피하며 목적지에 도착하기 위한 최적의 행동을 보일 수 있을 것이다.

참 고 문 헌

[1] V Mnih. "Human-level control through deep reinforcementlearning" Nature volume 518, pages 529 - 533 , 26 February 2015

[2] Christopher JCH Watkins and Peter Dayan. "Q-learning. Machine learning" 8(3-4):279 - 292, 1992.

[3] Marc Carreras P´erez et al. "A proposal of a behavior-based control architecture with reinforcement learning for an autonomous underwater robot" Universitat de Girona, 2003.

[4] 우주현 “심층강화학습을 이용한 무인수상선의 충돌회피” 서울대학교 대학원 [국내박사], 2018

[5] 김기서, 문정환, 김동규, 조승범, 이장명 “모바일 로봇의 자율 네비게이션을 위한 심층 강화 학습 기반 자연행동 학습” (제어·로봇·시스템학회 논문지, Vol.24 No.3, [2018])[KCI등재,SCOPUS]