광운대학교 KwangWoon University

출발드론팀

LIG Nex1

유니티 시뮬레이션 환경 내 강화학습을 통한 무인기 자율운항

김경범 · 김도연 · 임태헌 · 최형석 · 황지웅

I. 연구의 배경

현재 드론은 많은 소비자, 산업, 정부 및 군사 애플리케이션 에 사용된다. 여기에는 항공, 사진/비디오, 화물 운송, 경주, 수색 및 측량 등을 포함한다. 특수한 목적을 가진 임무용 드론, 레저/완구용 드론, 산업용 드론 등 다양한 분야에서 드론의 활용 사례가 늘고 있다. 그러나 현재 소형 드론에 적용된 센서의 정밀도 및 오차범위가 크고 외부 환경적 요인에 영향을 많이 받고 있어 임무용 드론의 경우 전문 조종사의 투입이 필수적이다. 현재 비행자동화 수준은 40%정도이고 조종사의 역할이 60%이다. 또한 조종사의 임무 성공 유무는 임무 작업 환경에 대한 다수의 경험에 큰 영향을 받는다.

II. 연구의 목적

본 프로젝트에서는 Unity 시뮬레이션 환경 내에서 산악 지형 환경을 만들어 강화학습이 적용된 드론의 자율 비행을 진행한다.
Unity를 이용하여 드론이 자율 주행할 수 있는 환경을 실제 지형과 가깝게 제작을 하고, PPO 알고리즘과 보상함수를 설계하여 다양한 상황에서 드론이 장애물 충돌 없이 목적지까지 비행할 수 있도록 한다. 그리고 자율 비행의 과정 및 결과를 Unity 환경 내에서 확인하며 Tensorboard로 Performance를 분석한다.

III. 연구 내용

산악환경 조성 및 시뮬레이터 UI 설계

Fig 1. Drone Simulator 메인 화면

Fig 2. Camera1 화면

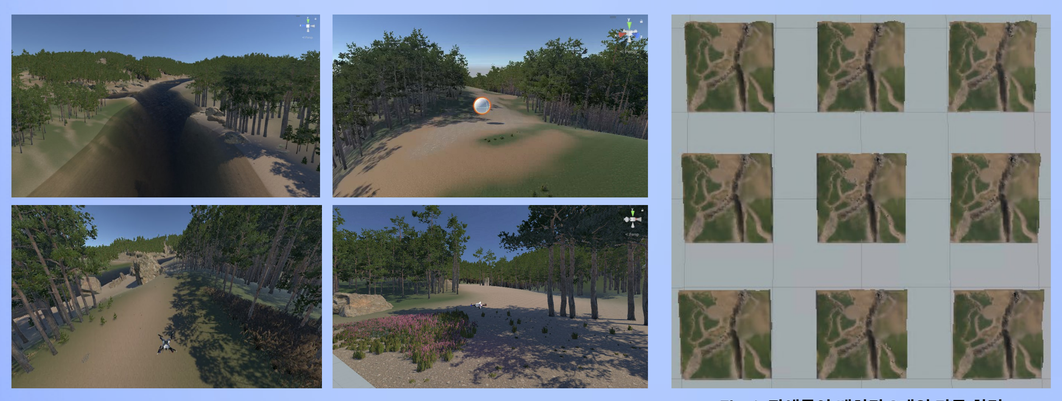

Fig 4. 장애물이 배치된 9개의 다른 환경

시뮬레이션 환경으로는 산악 환경을 선정하였다. Fig1은 드론 시뮬레이터의 메인 화면, Fig2는 카메라 시점을 위에서 볼 수 있게 하여 게임처럼 드론의 자율비행을 시뮬레이션해볼 수 있게 만들었다.
Fig3은 환경의 세부 디자인으로 나무, 바위와 같은 환경 요소를 이용하여 드론이 피해야 하는 정적 장애물을 배치하였고 새 떼와 같은 동적 장애물을 배치하였다.
두 종류의 장애물을 회피하여 출발지로부터 목적지까지 드론이 이동하는 환경을 구성하였다. Fig4는 환경에서 동적 장애물 및 정적 장애물의 위치를 바꾼 9개의 다른 환경이다. 이러한 9개의 다른 환경에서 강화학습을 진행하였다.
드론의 경우 실제와 유사하게 하기 위해 속도는 최대 8m/s로 설정했으며 초기 환경에서 목표의 높이는 약 180m, 드론과 목표지점 사이의 거리는 약 900m로 설정했다.

보상함수 설계 및 파라미터 설정

```
behaviors:
  Drone:
    trainer_type: ppo
    hyperparameters:
      batch_size: 2024
      buffer_size: 20240
      learning_rate: 0.0003
      beta: 0.005
      epsilon: 0.2
      lambd: 0.95
      num_epoch: 3
      learning_rate_schedule:
linear
    network_settings:
      normalize: true
      hidden_units: 512
      num_layers: 3
      vis_encode_type:
simple
    reward_signals:
      extrinsic:
        gamma: 0.99
        strength: 1.0
    keep_checkpoints: 5
    max_steps: 12000000
    time_horizon: 500
    summary_freq: 20000
    threaded: true
```

Fig 5. PPO 알고리즘의 하이퍼파라미터

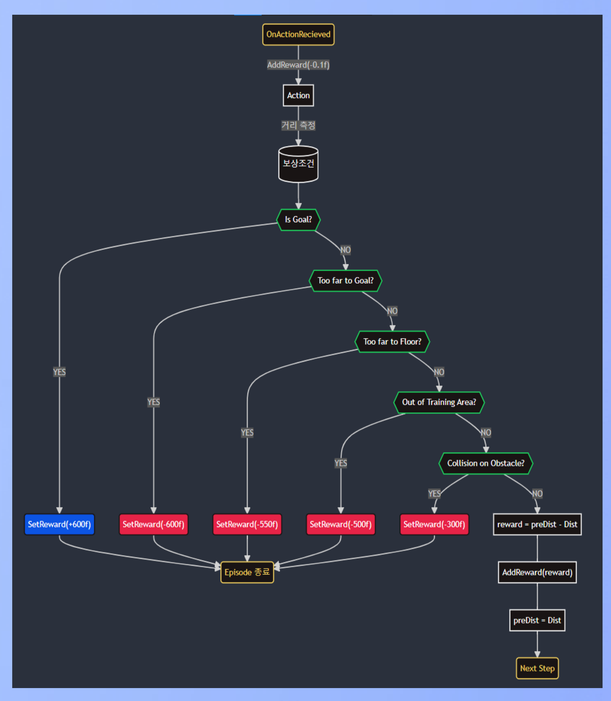

Fig 6. 보상함수 Flow Chart

Fig5는 PPO 알고리즘의 하이퍼파라미터이다. ML-Agents에서 제공하는 PPO 알고리즘의 하이퍼파라미터를 수정하며 실험해서 드론이 장애물과의 충돌을 피해 목적지까지 도달할 수 있는 최적의 하이퍼파라미터를 찾았다.

Fig 7. 센서를 통해 장애물을 감지하는 드론

Fig6는 보상함수에 대한 설계이다. 목표지점에 도달하면 양의 보상을 주고 Episode를 종료한다. 목표지점과 너무 많이 멀어지거나, Training 지역을 벗어나거나, 바닥과의 거리가 너무 멀어지거나, 장애물을 감지하여 가깝다고 판단하면, 음의 보상을 주고 Episode를 종료한다. 그 외에는 이전 step에서 측정한 목표지점과의 거리의 차이만큼 보상을 누적한다. 이는 목표지점과 가까워지면 양의 보상, 멀어지면 음의 보상을 누적한다. 그리고 다음 step을 진행한다.

IV. 연구 결과

Fig8과 Fig9는 순서대로 보상과 손실 값에 대한 결과 그래프이다.
x축은 학습 스텝 수, y축은 각각 평균 보상 값, 손실 값을 의미 한다.

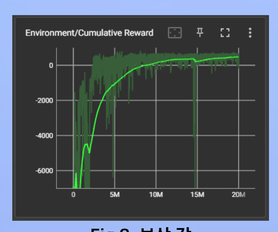

Fig 8. 보상 값

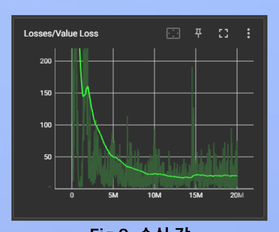

Fig 9. 손실 값

학습 초기에는 보상 값이 낮고 손실 값이 높은 반면 점차 학습이 진행됨에 따라 보상 값이 높아지고 손실 값이 낮아지는 것을 확인할 수 있다. 이를 통해 학습이 점진적이고 안정적으로 됨을 알 수 있다.

V. 결론 및 향후계획

Unity를 활용하여 무인기 강화학습 시뮬레이터 환경을 구성하고 PPO 알고리즘과 보상함수 설계를 활용하여 효과적인 자율비행 강화학습을 진행했다. 결과적으로 다양한 산악환경에서 안정적인 자율비행능력을 확인했고 TensorBoard로 학습이 안정적으로 이루어졌음을 확인했다.
향후에는 PPO 알고리즘을 이용한 멀티 에이전트 강화학습을 적용하여 여러 대의 드론이 편대를 이루어 목표지점에 도달하는 연구를 수행할 예정이다.

KWANGWOON UNIVERSITY