한국컴퓨터그래픽스학회
Korea Computer Graphics Society Vol. 30, No. 3, P. 109~123

자율주행 차량 시뮬레이션에서의 강화학습을 위한 상태표현 성능 비교

안지환O 권태수*

한양대학교 컴퓨터소프트웨어학과

(k125tw, taesoo)@hanyang.ac.kr

Comparing State Representation Techniques for Reinforcement Learning in Autonomous Driving

Jihwan AhnO Taesoo Kwon*

Hanyang University

요 약

딥러닝과 강화학습을 활용한 비전 기반 엔드투엔드 자율주행 시스템 관련 연구가 지속적으로 증가하고 있다. 일반적으로 이러한 시스템은 위치, 속도, 방향, 센서 데이터 등 연속적이고 고차원적인 차량의 상태를 잠재 특징 벡터로 인코딩하고, 이를 차량의 주행 정책으로 디코딩하는 두 단계로 구성된다. 도심 주행과 같이 다양하고 복잡한 환경에서는 Variational Autoencoder(VAE)나 Convolutional Neural Network(CNN)과 같은 네트워크를 이용한 효율적인 상태 표현 방법의 필요성이 더욱 부각된다. 본 논문은 차량의 이미지 상태 표현이 강화학습 성능에 미치는 영향을 분석하였다. CARLA 시뮬레이터 환경에서 실험을 수행하였고, 차량의 전방 카메라 센서로부터 취득한 RGB 이미지 및 Semantic Segmented 이미지를 각각 VAE와 Vision Transformer(ViT) 네트워크로 특징 추출하여 상태 표현 학습에 활용하였다. 이러한 방법론이 강화학습에 미치는 영향을 실험하여, 데이터 유형과 상태 표현 기법이 자율주행의 학습 효율성과 결정 능력 향상에 어떤 역할을 하는지를 실험하였다.

Abstract

Research into vision-based end-to-end autonomous driving systems utilizing deep learning and reinforcement learning has been steadily increasing. These systems typically encode continuous and high-dimensional vehicle states, such as location, velocity, orientation, and sensor data, into latent features, which are then decoded into a vehicular control policy. The complexity of urban driving environments necessitates the use of state representation learning through networks like Variational Autoencoders (VAEs) or Convolutional Neural Networks (CNNs). This paper analyzes the impact of different image state encoding methods on reinforcement learning performance in autonomous driving. Experiments were conducted in the CARLA simulator using RGB images and semantically segmented images captured by the vehicle's front camera. These images were encoded using VAE and Vision Transformer (ViT) networks. The study examines how these networks influence the agents' learning outcomes and experimentally demonstrates the role of each state representation technique in enhancing the learning efficiency and decision- making capabilities of autonomous driving systems.

키워드: 자율주행, 강화학습, 상태 표현, 시뮬레이션, 가상환경

Keywords: Autonomous driving, Reinforcement learning, State representation, Simulation, Virtual environment

*corresponding author: Taesoo Kwon/Hanyang University(taesoo@hanyang.ac.kr)

Received : 2024.06.15./ Review completed : 1st 2024.07.02. / Accepted : 2024.07.05.
DOI : 10.15701/kcgs.2024.30.3.109
ISSN : 1975-7883(Print)/2383-529X(Online)

1. 서론

인공지능 기술의 급속한 발전에도 불구하고, 자율주행 자동차의 상용화는 여전히 많은 어려움이 있다. 특히 차량의 다양하고 연속적인 입력을 처리하는 어려움에 효과적으로 대응하기 위해 주로 모듈식과 엔드투엔드 방법론이 활용된다. 모듈식 방법론은 주행 시스템을 인식[1], 계획[2], 제어[3] 등 여러 독립적인 모듈로 분리하여 개발하는 반면, 엔드투엔드 방법론은 모방학습[4] 또는 강화학습[5]을 통해 환경 관찰에서 차량 동작으로 직접 매핑되는 주행 정책을 학습한다.

강화학습은 에이전트가 매 스텝마다 환경으로부터 상태와 보상을 제공받고 시행착오를 통해 최적의 행동을 선택하는 과정으로 학습한다 [6]. 자율주행에서 강화학습의 목표는 입력 상태에 맞는 적절한 행동을 수행하여 가장 높은 누적 보상을 얻을 수 있는 주행 정책을 개발하는 것이다. 자율주행 차량의 상태는 일반 적으로 차량의 위치, 속도, 방향, 주변 환경 인식 데이터 등을 포함한다. 행동은 가속, 제동, 조향각 조정 등의 주행 제어 명령으로 구성되며, 보상은 경로 추종 정확도, 충돌 회피 등의 기준에 따라 설정된다. 그러나 강화학습 알고리즘을 훈련하려면 다양한 주행 시나리오를 포괄하기 위해 위험할 수 있는 행동을 포함한 방대한 양의 데이터와 환경과의 상호작용이 필요하다. 상태의 효율적이 고 정확한 표현은 학습 성능과 효율에 직접적인 영향을 미친다. 선행 연구에서는 RGB 이미지로부터 VAE[7] 등의 네트워크를 통 해 특징을 추출하여 상태 표현 학습을 하였고, 이 방법이 학습을 향상시킬 수 있다는 것을 보였다[8, 9]. 그러나 비전 기반 엔드투엔드 시스템에서 우리와 같이 전면 카메라의 RGB와 Semantic Segmented 이미지 데이터를 VAE, ViT[10]로 사전 인코딩하여 잠 재 벡터로 사용했을 때 에이전트의 학습 성능에 미치는 영향에 대해 비교한 연구는 없었다. 따라서 본 논문에서는 도심 주행 시뮬레이터 CARLA[11]를 활용하여, 입력 데이터의 유형과 상태 표현 학습 네트워크에 따른 강화학습 에이전트의 성능 차이를 정 량 지표와 정성 지표를 통해 체계적으로 평가하였다. 이를 통해 비전 기반 자율주행 차량 제어 문제에서 이미지 상태를 ViT로 사 전 인코딩하는 것이 VAE를 사용하는 기존 방법보다 에이전트의 학습 효율을 더 향상시킬 수 있음을 증명하였다.

본 논문의 기여는 다음과 같다. 첫째, 본 연구는 기존 연구들

과 달리 VAE와 ViT 두 가지 상태 표현 기법을 비교 분석하였다. 이를 통해 각 기법이 자율주행 차량의 강화학습 성능에 미치는 영 향을 체계적으로 평가하여, 상태 표현 방법의 효과를 명확히 밝혔다. 둘째, 차량의 전방 카메라로부터 입력받은 RGB 이미지와 Semantic Segmented 이미지를 각각 사용하여 상태를 정의하고, 이러한 다양한 입력 데이터 유형이 학습 성능에 미치는 영향을 분석하였다. 이를 통해 입력 데이터의 유형이 자율주행 시스템의 성능에 미치는 중요한 역할을 확인하였다. 셋째, CARLA 시뮬레이터 환경에서 다양한 실험을 통해 각 상태 표현 기법과 데이터 유형이 자율주행 에이전트의 학습 효율성과 결정 능력 향상에 어떻게 기여하는지 실험적으로 입증하였다. 특히, ViT 인코더를 사용한 모델이 VAE 인코더를 사용한 모델보다 학습 초기부터 빠르게 수렴하고, 더 안정적인 성능을 유지함을 확인하였다. 이러 한 기여를 통해 본 연구는 비전 기반 자율주행 차량의 상태 표현 방법에 대한 새로운 인사이트를 제공하며, 향후 자율주행 시스템 개발에 참조 자료가 될 수 있다.

2. 관련 연구

2.1 모방학습

Pomerleau의 ALVINN[12] 연구를 시작으로 모방학습은 비전 기반 자율주행 연구에 성공적으로 적용되어 왔으며, 최근에는 그 적용 범위를 복잡한 도심 환경으로 확장하고 있다. 모방학습은 전문 운전자가 생성한 경로 데이터[4, 12, 13] 혹은 다양한 센서 데이터[14, 15]를 학습에 활용한다. 기존의 모듈식 접근법과 달리, 모방학습은 엔드투엔드 방식으로 자율주행 모델을 차량의 주행 목표에 맞게 최적화할 수 있어 개별 요소를 별도로 튜닝하는 과정을 줄일 수 있다[5, 12].

하지만 모방학습에는 몇 가지 한계점이 있다. 첫째, 에이전트가 직면할 수 있는 모든 잠재적 상황에 대한 전문가 데이터를 확 보하는 것이 현실적으로 불가능하다. 둘째, 대규모 데이터를 취 득하더라도 다양한 실제 도로 환경과 운전자 행동 패턴을 모두 포함하기 어려워 데이터 분포가 편향될 수 있다. 이러한 한계로 인해 모방학습 에이전트는 주행 중 예기치 못한 상황에 적응하는 데 어려움을 겪을 수 있다.

2.2 강화학습

모방학습과 더불어, 강화학습 또한 자율주행 연구에서 중요한 역할을 한다[16, 17]. 강화학습 에이전트는 환경과 상호작용하며 시행착오로부터 학습하기 때문에, 모방학습의 편향된 분포 문제에 대한 대안이 될 수 있다. 에이전트는 상태 입력에 따라 행동을 취하고, 그에 대한 보상을 얻으며 누적 보상을 최대화하는 것을 목표로 한다.

Dosovitskiy 등[11]은 A3C[18]를 사용하여 CARLA

시뮬레이 터에 대한 첫 강화학습 모델을 제안했지만, 모듈식 접근법과 모 방학습에 비해 낮은 성능을 보였다. 이는 강화학습의 샘플 비효율성 문제를 드러냈고, Liang 등[19]은 이를 해결하기 위해 사전 학습된 행동 복제 액터 네트워크를 갖춘 DDPG[20] 방식을 제안했다.

시각적 입력 처리 측면에서, Kendall 등[9]은 단일 전방 카메라와 DDPG를 사용하여 자율주행 차량이 하루 만에 도로를 따라가도록 학습시켰다. 그들은 VAE를 사용해 이미지 상태를 인 코딩하여 모델의 성능을 향상시켰으나, 본 연구에서와 같이 Se- mantic Segmented Image는 사용하지 않았다. 한편, Chen 등[21] 은 VAE로 사전 학습한 버즈 아이 뷰 RGB 이미지를 DDQN[22],TD3[23], SAC[24]의 입력으로 사용하여 비전 기반 자율주행 에이전트의 시각적 복잡도를 낮추려 했다.

유사하게, Kargar 등[25]은 CNN, VAE, ViT로 사전 인코딩한 버즈 아이 뷰 RGB 이미지를 DQN[26] 에이전트의 입력으로 사 용하여 비교 실험한 결과, ViT가 다른 네트워크보다 우수한 성능을 보임을 밝혔다. 한편, Xu 등[27]은 가상환경에서 학습한 강화 학습 에이전트를 실제 작업에 적용하는 자율주행 모델을 제안했 다. 그들은 가상환경과 현실 간 격차를 해소하기 위해 Semantic Segmented Image를 사용했지만, VAE나 ViT 등으로 사전 인코딩 하지는 않았다.

본 연구는 두 가지 측면에서 위 연구들과 다르다. 첫째, 우리

는 차량의 전방 카메라로부터 입력 받은 RGB와 Semantic Segmented Image의 두 가지 유형의 입력 데이터를 사용하여 에이전트의 상태를 정의하였다. 둘째, 기존 연구들이 주로 VAE를 이미 지 인코더로 사용한 반면, 우리는 VAE뿐만 아니라 ViT도 이미지 사전 인코더로 사용하였고, 다양한 조합에 대해 비교 분석을 수 행하였다.

3. 시스템 오버뷰

우리는 비전 기반 자율주행 차량이 경로를 벗어나지 않고 주행하도록 제어하는 문제를 강화학습을 이용하여 해결한다. 이 연구의 목적은 다양한 상태 표현 방식이 학습 성능에 미치는 영향을 실험하여, 자율주행 알고리즘의 성능을 최적화하는 것이다. 우 리는 이미지 상태 표현을 VAE 또는 ViT로 인코딩하여 PPO[28]의 입력 상태(state)로 사용하였으며 구조와 학습 방법은 Figure 1과 같다. 우리는 CARLA 시뮬레이터에서 Proximal Policy Optimization(PPO) 기반 강화학습을 적용하였다. PPO는 Actor-Critic 구조를 바탕으로 하여, 각 상태에 대한 행동을 결정하고 해당 상 태의 가치를 평가하는 방식이다. 이 방식은 연속적인 제어 문제에 적합하며, 효율적인 정책 최적화를 제공한다. 정책 업데이트 시 이전 정책으로부터의 편차를 제한하는 클리핑된 목표 함수를

사용함으로써 안정적인 학습과 빠른 수렴을 도모한다. 또한 자 율주행과 같은 복잡한 환경에서 높은 차원의 연속 행동 공간을 효과적으로 다룰 수 있기 때문에 PPO를 적용하였다. 강화학습의 학습 대상인 정책 네트워크 (Figure 1의 PPO Agent)는 시뮬레이션 환경에서 차량의 상태 정보를 활용하여 주 행 행동을 결정한다 (section 4). 환경은 매 스텝마다 행동이 주어진 상태에서 얼마나 효과적으로 수행되었는지를 평가한다 (section 5). 이 과정이 반복되는 강화학습의 한 에피소드동안 차량은 고정된 경로의 웨이 포인트를 따라 주행한다. 경로에서 에이전트 가 다음 웨이 포인트나 목적지에 도달하면 에피소드는 성공으로 간주하고, 충돌하거나 경로를 벗어나는 등의 경우에는 실패로 간 주하여 학습이 조기 종료되며 보상을 얻는다. 우리는 다양한 인코더를 이용하여 학습을 진행하였고, 각 인코더가 이미지 상태 인코딩을 얼마나 효과적으로 수행하는지 다양한 실험을 통해 확인하였다 (section 6).

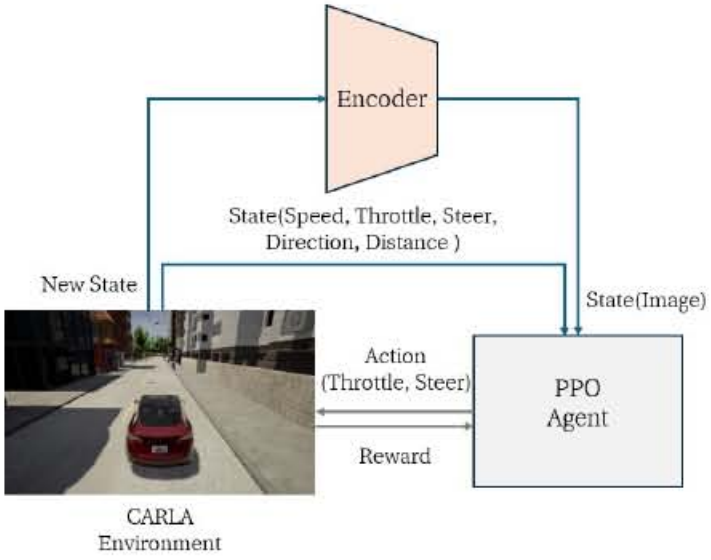

Figure 1: System Structure

4. 상태 표현

우리는 에이전트의 상태를 이미지(image), 차량의 속도(Speed), 가속(Throttle), 이전 스티어링(Previous Steering), 주행 방향(Direction), 도로 중심과의 거리(Distance from center)로 정 의하였다. 에이전트의 상태를 나타내는 세부 구성 요소는 Table 1 과 같다. 이미지 상태는 차량의 전면 카메라 센서로부터 입력 받은 160x80 크기의 3채널 RGB 또는 Semantic Segmented Image를 VAE 또는 ViT 인코더를 거쳐 95차원의 잠재 벡터로 인코딩하였 다. 이미지 이외의 상태들은 모두 1차원 벡터이다.

State	Description
Image	Encoded latent vector from car's image camera
Speed	Speed (km/h)
Throttle	Acceleration

Steer	Previous timestep steering
Direction	Angle between vehicle and waypoint
Distance	Signed perpendicular distance from waypoint

Table 1: Agent's state

4.1 이미지 상태 인코딩 기법 선택 이유

본 연구에서는 다양한 네트워크 모델 중에서 VAE와 ViT를 선택하여 이미지 상태 인코딩을 수행하였다. CNN 등 다른 네트워크 모델들도 이미지 인코딩에 효과적일 수 있으나, 본 연구는 VAE와 ViT 두 가지 모델의 성능 차이를 중점적으로 분석하기 위해 이들을 선택하였다. VAE는 이미지 데이터를 저차원 잠재 공간으로 효과적으로 인코딩할 수 있는 생성 모델로, 기존 연구 에서도 이미지 상태 표현의 효과가 입증되었기 때문에 본 연구에 포함되었다. ViT는 최근 컴퓨터 비전 분야에서 주목받고 있는 모 델로, 언어 모델 등의 다양한 분야에서 성공적인 효과를 보이고 있는 트랜스포머 아키텍처를 기반으로 하고 있다. ViT는 이미지 패치 간의 관계를 학습하며, 이는 고차원 잠재 공간에서 유의미한 특성을 추출하는 데 강점을 지닌다. 이러한 강점 덕분에 ViT 가 도심 환경과 같은 복잡한 시각적 정보를 처리하는 데 뛰어난 성능을 보일 것이라는 가정하에 선택되었다. 우리는 제한된 연구 범위 내에서 두 모델의 비교 분석을 통해 더 명확한 결론을 도출하고자 하였다.

4.2 이미지 상태 인코딩

우리는 자율주행 차량의 시각적 인식을 향상시키기 위해, CARLA 시뮬레이터의 Town2 환경을 주행하며 160x80 사이즈의 RGB와 Semantic Segmented image 데이터 각각 10,000장을 수집하였다 (Figure 2 참조). 이 데이터 수집 과정은 다양한 주행 시나리오와 환경에서 차량이 마주할 수 있는 시각적인 요소들을 포괄하기 위함이다. Semantic segmented을 적용했을 때 RGB보 다 주요 도로 표지에 대한 상태를 정확하게 찾아낼 수 있으며, 이미지의 픽셀은 높은 확률로 이웃하는 픽셀과 동일한 Semantic label을 갖기 때문에[29] 에이전트의 학습 효율을 높일 수 있고, 어떤 표현의 이미지로 학습시켰는지에 따라 상태값이 달라 에이 전트의 학습 효율에 차이가 있다는 가정하에 실험하였다. 취득한 이미지들은 상태 표현 학습에 이용하기 위해 VAE와 ViT 인코더를 거쳐 특징 추출을 한 RGB-VAE, SS-VAE, RGB-ViT, SS-ViT의 네 가지 사전학습 인코더 모델을 생성하였다. 각 인코더들은 입 력 데이터를 잘 인코딩하였는지 확인하기 위해 디코딩하여 학습 데이터셋의 랜덤한 이미지를 Figure 3와 같이 복원하는 과정을 거쳤다.

Figure 2: Camera Image

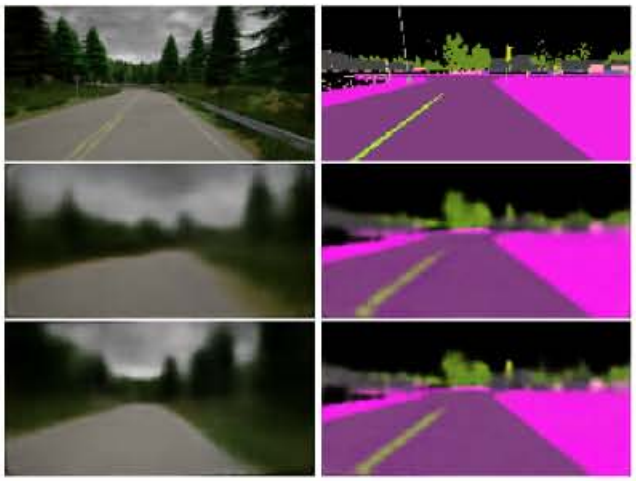

Figure 3: Top row: (Left) Original RGB Image, (Right) Original SS Image. Middle row: (Left) RGB Image Reconstructed by VAE, (Right) SS Image Reconstructed by VAE. Bottom row: (Left) RGB Image Reconstructed by ViT, (Right) SS Image Reconstructed by ViT.

4.3 인코더의 활용

본 논문에서는 자율주행 차량의 주행 결정 과정을 최적화하기 위해, 차량 전면 카메라에서 취득한 이미지 데이터를 VAE 또는 ViT 인코더를 사용하여 효과적으로 잠재 공간에 인코딩하였다. 이 잠재 벡터는 Figure 4와 같이 PPO의 입력 상태로 이용하였다. 이미지 데이터의 인코딩 방식은 실험에 따라 다르게 설정되었다. Semantic Segmented (SS) 이미지를 사용하는 경우에는 SS-VAE와 SS-ViT 인코더를 활용하였으며, 차량 전면 카메라도 Semantic Segmented 카메라로 설정하였다. 반면, RGB 이미지를 사용하는 경우에는 RGB-VAE와 RGB-ViT 인코더를 사용하였다. 이러한 인코딩 접근 방식은 PPO 강화학습의 학습 효율을 크게 향상시킨다[9].

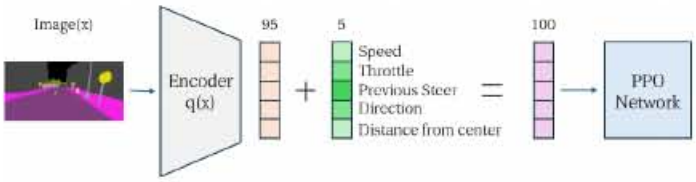

Figure 4: Pretrained Encoder + PPO

5. 보상 설계

보상 함수는 에이전트의 행동이 주어진 상태에서 얼마나 효과적으로 수행되었는지를 평가하는 지표로 활용되며, 환경으로부터 행동에 따라 결정된다. 에이전트는 보상의 결과를 기반으로 학습을 진행하여 더 높은 보상을 얻을 수 있는 방향으로 행동 전략을 수정한다. 본 연구는 Kendall 등[9]의 연구를 참고하여 보상 체계를 구축하였다. Kendall 등은 목표 속도 유지를 주된 보상 기준으로 설정하고, 주행의 정확성 및 충돌 방지는 조기 종료 조건으로 적용하였으나, 본 연구에서는 속도 유지 보상을 좀 더 세분화하고 Kendall 등의 조기 종료 조건들을 보상 체계에 통합하여 실험하였다. 최종 보상 함수는 다음과 같이 설계하였다.

$$R(s,a,s') = R_{\text{path}}(s,a) \cdot R_{\text{speed}}(s,a) + R_{\text{safety}}(s,a).$$

보상 함수 $R(s,a,s')$는 주어진 상태 s, 행동 a, 그리고 그

결과로 이동한 새로운 상태 s'에 기초하여 계산된다. 이 함수는 경로 추종, 속도 유지, 안전 운전 세 가지 주요 구성 요소로 이루어져 있다.

경로 추종 보상. 경로 추종 보상 $R_{\text{path}}(s)$은 차량이 도로의 중심선을 얼마나 잘 따르는지에 따라 계산된다. 이는 중심으로부터의 거리 보상 R_{center}과 각도 보상 R_{angle} 두 가지 주요 요소로 구성된다. 먼저, 중심으로부터의 거리 보상은 차량이 도로의 중심선에 가까울수록 높은 값을 갖는다.

$$R_{\text{center}}(s,a) = \max\left(1.0 - \frac{d_{\text{center}}}{d_{\max}}, 0.0\right).$$

여기서 d_{center}는 차량의 현재 위치와 도로 중심선 간의 거리이며, $d_{\max}$는 최대 허용 거리로 2m로 설정하였다. 각도 보상은 차량의 진행 방향이 도로의 진행 방향과 얼마나 잘 일치하는지를 나타낸다.

$$R_{\text{angle}}(s,a) = \max\left(1.0 - \left|\frac{\theta_{\text{vehicle}} - \theta_{\text{road}}}{\theta_{\max}}\right|, 0.0\right).$$

여기서 θ_{vehicle}는 차량의 현재 진행 방향 각도, θ_{road}는 도로의 진행 방향 각도, $\theta_{\max}$는 최대 허용 각도 차이로 20도를 넘어가면 경로를 이탈한 것으로 설정하였다. 따라서 최종 경로 추종 보상은 다음과 같이 정의된다.

$$R_{\text{path}}(s,a) = R_{\text{center}} \cdot R_{\text{angle}}.$$

속도 유지 보상. $R_{\text{speed}}(s,a)$은 차량의 속도가 목표 속도 범위 내에 있을 때 더 높은 보상을 받도록 설계하였다. 구체적으로, 차량의 속도가 최소 속도 15km/h 미만일 경우 현재 속도를 최소 속도로 나눈 값을 보상으로 설정하였다. 차량의 속도가 목표 속도 25km/h보다 높고 최대 속도 40km/h 미만일 경우에는 (1.0 - (현재속도 - 목표속도) / (최대속도 - 목표속도)) 값을 보상으로 설정하였다. 만약 차량의 속도가 목표 속도 25km/h를 잘 유지한다면 최대 보상인 1.0으로 설정하였다. 이는 다음과 같이 정의할 수 있다: 먼저, 차량의 속도가 최소 속도 v_{min}보다 낮을 때의 보상은 다음과 같다:

$$R_{\text{speed}}(s,a) = \frac{\text{velocity}}{\text{v}_{\text{min}}}.$$

차량의 속도가 목표 속도 v_{target}보다 높지만 최대 속도 $\text{v}_{\max}$보다 낮을 때의 보상은 다음과 같다:

$$R_{\text{speed}}(s,a) = 1.0 - \frac{\text{velocity} - \text{v}_{\text{target}}}{\text{v}_{\max} - \text{v}_{\text{target}}}.$$

차량의 속도가 목표 속도 v_{target} 내에 있을 때의 보상은 다음과 같다:

$$R_{\text{speed}}(s,a) = 1.0.$$

따라서 최종 속도 유지 보상은 다음과 같이 정의된다:

$$R_{\text{speed}}(s,a) = \begin{cases} \frac{\text{velocity}}{\text{v}_{\text{min}}}, & \text{if velocity} < \text{v}_{\text{min}}, \\ 1.0 - \frac{\text{velocity} - \text{v}_{\text{target}}}{\text{v}_{\max} - \text{v}_{\text{target}}}, & \text{if } \text{v}_{\text{min}} \le \text{velocity} < \text{v}_{\max}, \\ 1.0, & \text{if velocity} = \text{v}_{\text{target}}. \end{cases}$$

안전 운전 보상. 안전 운전 보상 $R_{\text{safety}}(s,a)$은 차량이 안전한 운전을 하는지를 평가하며, 장애물 회피와 충돌 방지 능력을 중점적으로 고려한다. 위험한 상황을 피하고 안전한 행동을

할 때 높은 보상을 주었다.

$$R_{\text{safety}}(s,a) = \begin{cases} w_1, & \text{if collision occurs,} \\ w_2, & \text{if distance from center} > \text{max distance from center,} \\ w_3, & \text{if episode start time} + 10 < \text{current time and velocity} < 1.0, \\ w_4, & \text{if velocity} > \text{max speed.} \end{cases}$$

이와 같이, 안전 운전 보상은 충돌 발생, 도로 중심으로부터의 거리 초과, 에피소드 시작 후 10초 경과 시 속도가 1.0km/h 이하, 최대 속도 초과 등의 상황을 반영하여 가중치 w_1, w_2, w_3, w_4를 통해 정의하였고 모두 보상을 -10으로 설정하였다. 이를 통해 에이전트가 위험한 상황을 회피하고 안전한 주행을 유지하도록 유도하였다.

실험

6.1 주행 시뮬레이션 환경

이상에서 설명한 상태 기반 자율주행의 강화학습은 오픈소스 도심 주행 시뮬레이터 CARLA를 통해 실험하였다. CARLA는 언리얼 엔진4 기반의 자율주행 연구용 프로그램이며, 이를 이용하여 Figure 5와 같은 도심 주행 맵과 다양한 센서들을 현실적으로 시뮬레이션할 수 있다. 우리는 CARLA의 맵들 중 도심 환경인 Town2와 복잡한 도로 환경이 반영되어 있는 Town7에서 실험을 하였다.

Figure 5: CARLA Simulation Map

한 에피소드는 최대 7,500 타임스텝 동안 진행되고, 각 훈련 epoch마다 총 3,000,000 타임스텝의 학습을 진행하였다. 이를 활용해 대리손실함수를 구성하고, 수집된 데이터는 Mini-Batch Stochastic Gradient Descent(SGD) 방법을 사용하여 여러 Epoch 동안 학습을 반복하면서 정책을 점진적으로 개선하였다. 10번의 에피소드마다 현재 정책 π_θ 과 이전 정책 $\pi_{\theta_{\text{old}}}$ 와의 편차에 대해 최적화를 진행하고 상태를 저장하여 학습 버퍼를 생성하였다.

6.2 상태 인코딩 네트워크의 학습 성능 비교

본 논문에서는 VAE와 ViT를 이용한 이미지 상태 인코딩의 효과가 에이전트의 학습에 중요한 역할을 했다. 각 인코더의 훈련 및 검증 손실의 변화를 분석하여 VAE와 ViT가 이미지 상태 인코딩을 어떻게 효과적으로 수행하는지 확인하였다.

.2.1 손실 함수 차이 및 정규화 이유

VAE와 ViT의 손실 함수는 모델의 구조적 차이와 학습 목표에 따라 다르다. VAE는 생성 모델로서 입력 데이터를 저차원 잠재 공간으로 매핑한 후 재구성하는 것을 목표로 하며, 이에 따라

재 구성 손실과 Kullback-Leibler 발산(KL divergence)을 포함한 복 합 손실 함수를 사용한다. 반면, ViT는 이미지 패치 간의 관계를 학습하고 고차원 잠재 공간에서 유의미한 특성을 추출하는 모델 로서 주로 재구성 손실만을 사용한다. 본 논문에서는 VAE와 ViT 의 모델 및 손실 함수 등을 본래의 목표에 맞게 유지하였고 이러 한 차이로 인해 VAE와 ViT의 손실 값의 직접적인 비교는 어렵다. 따라서 본 논문에서는 손실값을 정규화하여 수렴 값이 1이 되도 록 하였으며, 이를 통해 두 모델의 학습 성능을 보다 정확하게 시각적으로 비교할 수 있었다. Figure 6와 Figure 7에 제시된 그래프는 정규화된 손실 값을 사용하여 VAE와 ViT의 학습 및 검증 손실을 비교한 결과이다.

6.2.2 학습 손실 비교

Figure 6와 같이, ViT와 VAE의 훈련 손실 변화를 비교하였다. VAE는 훈련 초기에 급격한 손실 감소를 보이며, 훈련이 끝날 때까지 안정적인 감소를 유지하였다. 이는 VAE가 복잡한 이미지 데이터에서 특징을 효과적으로 추출하고 잘 일반화함을 보여준다. 반면, ViT의 훈련 손실은 높은 초기 값에서 시작하여 점진적으로 감소하였으며, 이는 ViT가 복잡한 이미지 인코딩 문제에 상대적으로 천천히 학습하고 있음을 나타낸다.

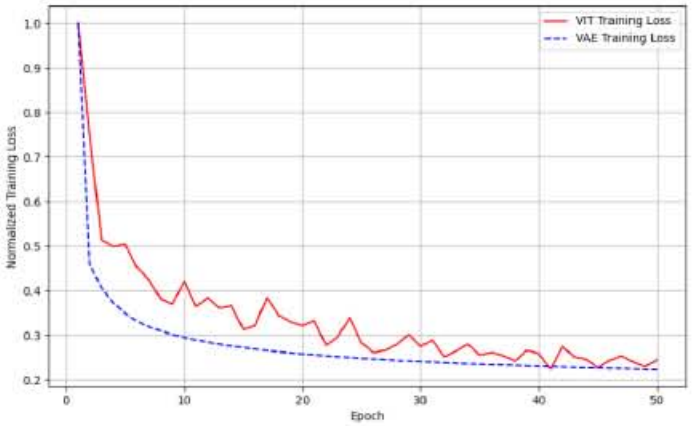

Figure 6: Training Loss/Epoch Comparison

6.2.3 검증 손실 비교

Figure 7 와 같이, ViT와 VAE의 검증 손실 변화를 비교하였다. ViT의 검증 손실은 초기 급격한 감소 후 안정적인 수준을 유지하여 VAE보다 검증 데이터셋에 대해 높은 일반화 성능을 보여준다. VAE의 검증 손실 또한 일관된 감소를 보여, 데이터 처리 과정에서 과적합 없이 잘 학습하고 있음을 나타낸다.

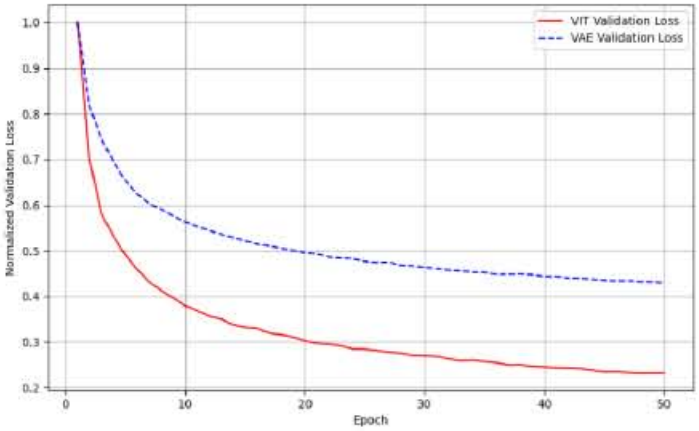

Figure 7: Validation Loss/Epoch Comparison

결과들을 통해 VAE가 ViT에 비해 초기 학습 속도와 안정성 면에서 우수한 성능을 보여주는 것을 알 수 있다. VAE의 빠른 수 렴 속도와 낮은 검증 손실은 복잡한 시각적 데이터에 대한

높은 적응력과 일반화 능력을 보여 준다. 반면, ViT는 점진적인 학습 곡선을 통해 도심 환경과 같은 다양한 특징을 갖는 이미지 데이터에 점차적으로 적응하는 것으로 나타난다. 이러한 결과는 향후 비전 기반의 자율주행 시스템의 상태 표현을 위한 인코더 선택에 중요한 정보를 제공하며, 특히 복잡하고 다양한 정보를 담고 있는 도심 환경에서의 주행 정책 개발에 있어 VAE뿐만 아니라 ViT 인코더도 충분히 활용 가치가 있음을 보여준다.

6.3 상태 인코딩에 따른 차량 주행 결과

본 논문에서는 VAE와 ViT를 이용한 이미지 상태 인코딩의 효과가 에이전트의

6.3.1 경로 중심으로부터의 편차

VAE 모델을 사용한 경우, Figure 8과 같이 차량이 경로 중심에 보다 가까이 유지되는 경향을 보였다. 반면, ViT 모델을 사용했을 때는 상대적으로 더 큰 편차를 경험했다. 이는 VAE 모델이 이미지 데이터의 공간적 구조를 더 잘 인식하고 이를 기반으로 보다 정확한 주행 경로 결정을 내리는 능력을 보여준다. 또한, SS 데이터에서 학습한 모델이 RGB 데이터를 사용한 모델에 비해 일반적으로 더 낮은 편차를 보여줌으로써, 사전 처리된 데이터 형식이 차량의 주행 정밀도를 개선하는데 기여함을 확인할 수 있다.

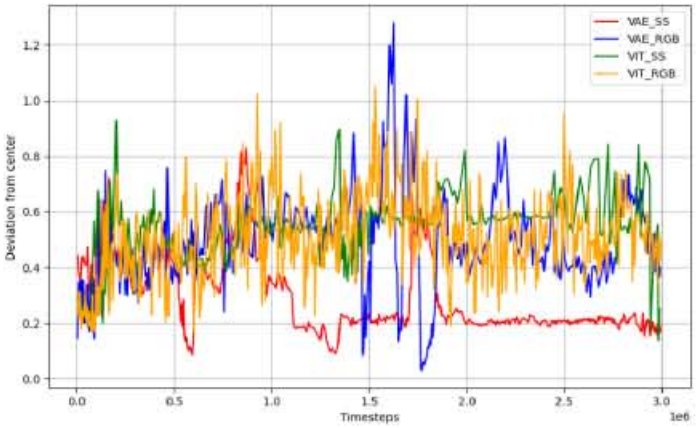

Figure 8: Deviation from Center

6.3.2 평균 보상

모든 모델에서 평균 보상은 학습을 거듭할수록 점진적으로 개선되었다 (Figure 9). 특히, SS-ViT 모델이 사용된 실험에서 다른 모델들보다 더 빠르고 지속적인 보상 증가가 관찰되었다. 이는 ViT가 주어진 주행 환경에서 더 적합한 행동을 선택하고, 이에 따라 더 높은 보상을 획득함을 시사한다. SS 데이터를 사용한 모델들이 RGB 데이터보다 더 일관된 보상 증가를 보여, 환경에 대한 더 명확한 이해가 보상 증가에 기여한 것으로 보인다.

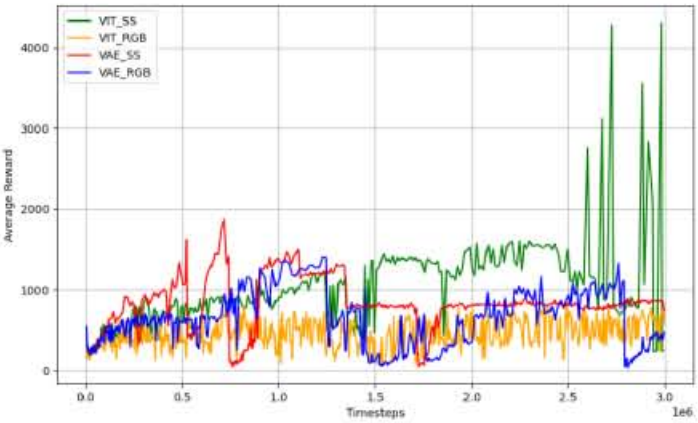

Figure 9: Average Reward

6.3.3 누적 보상

누적 보상에 있어서도 SS-ViT 모델이 VAE나 RGB 모델에 비해 더 높은 성능을 나타냈다 (Figure 10). 초기부터 누적 보상의 지속적인 증가가 관찰되며, 학습이 진행됨에 따라 이 격차는 더욱 확대되었다. 이 결과는 ViT의 높은 학습 효율과 더 나은 일반화 능력을 반영한다. SS 데이터를 사용한 실험들은 RGB 데이터보다 전반적으로 더 높은 누적 보상을 보여, 처리된 데이터가 복잡한 환경에서 보다 효과적으로 학습될 수 있음을 강조한다.

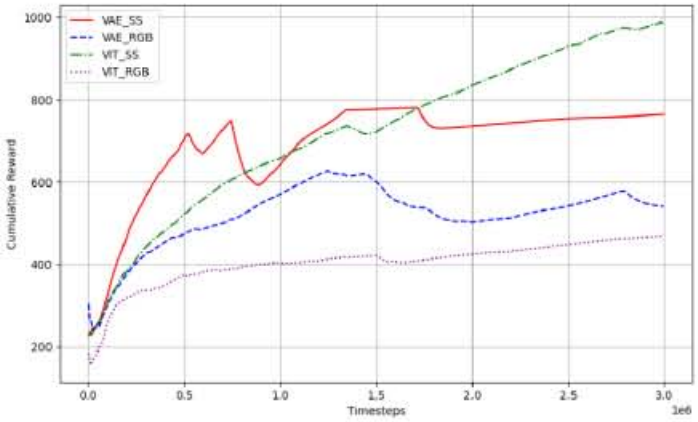

Figure 10: Cumulative Reward

이 결과들을 통해 ViT가 VAE에 비해 초기 학습 속도와 안정성 면에서 우수한 성능을 보여주는 것을 알 수 있다. ViT의 빠른 수렴 속도와 낮은 검증 손실은 복잡한 시각적 데이터에 대한 높은 적응력과 일반화 능력을 보여 준다. 반면, VAE는 점진적인 학습 곡선을 통해 다양한 특징을 가진 이미지 데이터에 점차적으로 적응하는 것으로 나타난다. 이러한 결과는 향후 비전 기반의 자율주행 시스템의 상태 표현을 위한 인코더 선택에 중요한 정보를 제공하며, 특히 복잡하고 다양한 정보를 담고 있는 도심 환경에서의 주행 정책 개발에 있어 VAE뿐만 아니라 ViT 인코더도 충분히 활용 가치가 있음을 보여준다. 또한, SS 데이터를 사용한 모델들이 RGB 데이터를 사용한 모델들보다 전반적으로 더 높은 성능을 보이는 것을 확인할 수 있었다. 이러한 결과는 복잡한 도심 환경에서의 자율주행 성능을 극대화하기 위해 사전 처리된 SS 데이터의 사용이 효과적임을 시사한다.

7. 결론

연구는 강화학습을 이용하여 자율주행 차량의 시각 기반 상태 인코딩 방식을 체계적으로 비교하였다. 실험 결과, Vision Transformer(ViT)는 Variational Autoencoder(VAE)에 비해 학습 초기부터 빠른 수렴 속도를 보이며, 안정적인 성능을 유지하는 반면, VAE는 점진적인 학습 곡선을 통해 보다 복잡한 이미지 패턴을 처리하는 능력을 개발하였다. 또한, Semantic Segmented(SS) 데이터는 RGB 데이터에 비해 환경 인식에 있어 더욱 정밀하게 작동함을 입증하였으며, 이는 자율주행 시스템의 학습 효율을 높이는 중요한 요소로 작용한다. 이러한 발견은 도심 환경에서의 자율주행 차량의 주행 정책을 개발함에 있어 ViT와 VAE의 적절한 활용 방안을 제시한다.

그러나 본 연구에는 몇 가지 한계점이 존재한다. 첫째,

연구에서 사용된 시뮬레이션 환경은 실제 도로 환경의 동적 요소들, 예를 들어 보행자나 다른 차량과 같은 움직이는 객체들을 포함하지 않고 있다. 이는 본 연구의 인코더가 실제 도로 환경의 예측 불가능한 변수들에 대응하는 능력을 평가하는 데 한계가 있음을 의미한다. 둘째, 본 연구에서는 단일 이미지 카메라 센서만을 사용하여 실험을 진행하였으나, 실제 자율주행 차량은 여러 카메라와 센서를 조합하여 환경을 인식한다. 이에 따라, 더 많은 센서 입력을 통합할 수 있는 시스템의 개발이 필요하다. 셋째, 본 연구는 주로 ViT와 VAE의 성능 차이에 초점을 맞추어 인코더와 차량의 이미지 상태에서 RGB나 SS로 동일한 데이터를 사용했지만, 추가적인 인코더의 조합과 다양한 네트워크 아키텍처의 실험을 통해 더욱 향상된 결과를 도출할 가능성이 있다. 예를 들어, ViT 인코더는 SS 데이터로 사전학습을 하고 차량의 전면 카메라 센서로부터의 입력은 RGB로 설정하여 에이전트를 학습시켜볼 수도 있다. 따라서 향후 연구에서는 다양한 동적 객체와 센서, 여러 조합의 이미지 상태 인코더 등의 실험을 계획하고 있다.

References

[1] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2722–2730.

[2] J. Chen, W. Zhan, and M. Tomizuka, "Autonomous driving motion planning with constrained iterative lqr," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 2, pp. 244–254, 2019.

[3] B. Paden, M. Cˇ a´p, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self- driving urban vehicles," IEEE Transactions on intelligent vehicles, vol. 1, no. 1, pp. 33–55, 2016.

[4] F. Codevilla, M. Müller, A. López, V. Koltun, and A. Dosovitskiy, "End-to-end driving via conditional imitation learn- ing," in 2018 IEEE international conference on robotics and automation (ICRA). IEEE, 2018, pp. 4693–4700.

[5] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al., "End to end learning for self-driving cars," arXiv preprint arXiv:1604.07316, 2016.

[6] R. S. Sutton and A. G. Barto, Reinforcement learning: An introduction. MIT press, 2018.

[7] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," arXiv preprint arXiv:1312.6114, 2013.

[8] I. Higgins, A. Pal, A. Rusu, L. Matthey, C. Burgess, A. Pritzel, M. Botvinick, C. Blundell, and A. Lerchner, "Darla: Improving zero-shot transfer in reinforcement learn- ing," in International Conference on Machine Learning. PMLR, 2017, pp. 1480–1490.

[9] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J.-M. Allen, V.-D. Lam, A. Bewley, and A. Shah, "Learning to drive in a day," in 2019 international conference on robotics and automation (ICRA). IEEE, 2019, pp. 8248–8254.

[10] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.

[11] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in Con- ference on robot learning. PMLR, 2017, pp. 1–16.

[12] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," Advances in neural information processing systems, vol. 1, 1988.

[13] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," arXiv preprint arXiv:1912.01603, 2019.

[14] D. Chen, B. Zhou, V. Koltun, and P. Kra¨henbu¨hl, "Learning by cheating," in Conference on Robot Learning. PMLR, 2020, pp. 66–75.

[15] Y. Pan, C.-A. Cheng, K. Saigol, K. Lee, X. Yan, E. Theodorou, and B. Boots, "Agile autonomous driving using end-to-end deep imitation learning," arXiv preprint arXiv:1709.07174, 2017.

[16] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. Al Sallab, S. Yogamani, and P. Pe´rez, "Deep reinforcement learning for autonomous driving: A survey," IEEE Transactions on Intel- ligent Transportation Systems, vol. 23, no. 6, pp. 4909–4926, 2021.

[17] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "End-to-end autonomous driving: Challenges and frontiers," arXiv preprint arXiv:2306.16927, 2023.

[18] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in International conference on machine learning. PMLR, 2016, pp. 1928– 1937.

[19] X. Liang, T. Wang, L. Yang, and E. Xing, "Cirl: Control- lable imitative reinforcement learning for vision-based self- driving," in Proceedings of the European conference on com- puter vision (ECCV), 2018, pp. 584–599.

[20] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous con-trol with deep reinforcement learning," arXiv preprint arXiv:1509.02971, 2015.

[21] J. Chen, B. Yuan, and M. Tomizuka, "Model-free deep reinforcement learning for urban autonomous driving," in 2019 IEEE intelligent transportation systems conference (ITSC). IEEE, 2019, pp. 2765–2771.

[22] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in Proceedings of the AAAI conference on artificial intelligence, vol. 30, no. 1, 2016.

[23] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in International conference on machine learning. PMLR, 2018, pp. 1587– 1596.

[24] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforce- ment learning with a stochastic actor," in International conference on machine learning. PMLR, 2018, pp. 1861–1870.

[25] E. Kargar and V. Kyrki, "Vision transformer for learning driving policies in complex multi-agent environments," arXiv preprint arXiv:2109.06514, 2021.

[26] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., "Human-level control through deep reinforcement learning," nature, vol. 518, no. 7540, pp. 529–533, 2015.

[27] N. Xu, B. Tan, and B. Kong, "Autonomous driving in real- ity with reinforcement learning and image translation," arXiv preprint arXiv:1801.05299, 2018.

[28] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," arXiv preprint arXiv:1707.06347, 2017.

[29] Q. Khan, T. Schön, and P. Wenzel, "Latent space reinforcement learning for steering angle prediction," arXiv preprint arXiv:1902.03765, 2019.

〈저자소개〉

안 지 환

- 2008-2014 홍익대학교 경영학부 학사
- 2023-현재 한양대학교 컴퓨터소프트웨어학과 석사과정
- https://orcid.org/0009-0002-4079-3538

권 태 수

- 1996-2000 서울대학교 전기컴퓨터공학부 학사
- 2000-2002 서울대학교 전기컴퓨터공학부 석사
- 2002-2007 한국과학기술원 전산학전공 박사
- https://orcid.org/0000-0002-9253-2156