

# 자율주행 차량을 위한 Latent Diffusion Model 기반의 주행 시나리오 생성

## Driving Scenario Generation Based on Latent Diffusion Model for Autonomous Driving Vehicles

전 명 환<sup>1</sup>, 조 건 희<sup>1</sup>, 이 형 철<sup>1\*</sup>(Myungwhan Jeon<sup>1</sup>, Kunhee Cho<sup>1</sup>, and Hyeongcheol Lee<sup>2,\*</sup>)<sup>1</sup>Department of Electrical Engineering, Hanyang University<sup>2</sup>Division of Electrical and Biomedical Engineering, Hanyang University

**Abstract:** This study presents a novel deep learning-based method for scenario generation to validate autonomous driving systems (ADSs). Testing ADSs for every possible scenario in the real world is challenging and costly due to the vast number of potential driving conditions, which can slow down the research and development process. To address this issue, a simulation-based validation method using driving scenarios has been introduced. Current scenarios typically rely on the recorded traffic data or the rule-based traffic agents. However, these conventional method-based scenarios often lack sufficient realism and diversity, which hinders effective validation. This study proposes an approach that leverages a deep learning method for scenario generation to enhance realism and diversity. First, a variational autoencoder (VAE) was employed to encode complex traffic characteristics into a compact latent space. The latent diffusion model was then used to add controlled noise into the latent space and learn this space through iterative denoising processes. Consequently, a new latent space was generated from the learned denoising process using Gaussian noise as the input. The new latent space provided diverse traffic characteristics, which the VAE decode into the driving scenarios that maintain realism while exhibiting diverse traffic. The generated scenarios were validated by comparing them with the scenarios made by VAE, while focusing on their realism and diversity.

**Keywords:** autonomous driving, simulation, driving scenario, deep learning, Latent Diffusion Model

### I. 서론

최근 자율주행 시스템(autonomous driving system, ADS)이 활발히 개발되고 있다[1-6]. ADS는 교통체증의 감소로 인한 도심 환경의 쾌적화 및 교통사고로부터 안전한 환경을 만드는 등의 기대효과를 불러올 수 있지만, 현실에 적용되기 위해서는 인간운전자와 함께 도로에서 주행할 때 안전이 보장되어야 한다. 이를 위해선 다양한 도로상황에서의 대응 능력에 대한 검증이 요구된다.

ADS 검증을 실제 도로에서 모든 상황에 대해 하는 것은 비용과 시간면에서 매우 소모적일 뿐 아니라 위험이 수반된다[7-9]. 이에 시뮬레이션 환경에서 다양한 시나리오를 기반으로 ADS를 검증하는 방식이 대두되고 있다[10-12]. 기본적인 시나리오 구성을 위해서 유럽 신차 평가 프로그램(The European New Car Assessment Program, Euro NCAP)과 국제표준기구(International Organization for Standardization, ISO)에서 ADS의 표준을 위해 제공하는 프로토콜을 이용할 수 있다. 하지만, 특정한 상황에서 레벨 2이하의 ADS가 만족해야 하는 최소한의 요구 조건만을 제시할 뿐, 레벨 3이상의 ADS를 탑재

한 차량이 직면할 수 있는 복잡한 교통 상황을 충분히 반영하지 못하는 경우가 많기 때문에, 표준 기반의 시나리오들은 검증을 위한 시나리오로 적합하지 않다. 보편적인 검증을 위해 SUMO [13], CARLA [14] 등과 같은 시뮬레이터들은 교통 데이터를 사용하거나 규칙 기반(rule-based) 방식으로 각 차량의 동작을 구현하여 시뮬레이션 환경을 구성한다. 하지만, 교통 데이터를 이용하는 방식은 새로운 요소를 추가하기 어렵고, 규칙 기반의 방식은 인간 주행 특성이 반영되지 않아 자연스러움과 현실성이 떨어져 검증을 위한 시나리오로 적합하지 않다.

앞서 언급한 평가 방법들의 문제를 해결하기 위해 주행 시나리오의 생성에 관한 연구가 활발히 진행되고 있다. 특히, 딥러닝 기반의 생성형 모델을 이용해 주행 시나리오를 생성하는 연구가 주목받고 있다[15-18]. 대표적으로, 변분 오토인코더(variational autoencoder, VAE) [19]를 이용한 시나리오 생성 모델이 있다. VAE를 활용하면 복잡한 교통 데이터를 신경망에 학습시킴으로써 잠재 공간(latent space)에 교통 데이터를 압축시킬 수 있다. 압축된 latent space를 이용하여 시나리오를

\* Corresponding Author

Manuscript received October 27, 2024; revised November 25, 2024; accepted November 27, 2024

전명환: 한양대학교 전기공학과 대학원생(mhj9707@hanyang.ac.kr, ORCID<sup>®</sup> 0009-0002-4734-2756)

조건희: 한양대학교 전기공학과 대학원생(cgh8892@hanyang.ac.kr, ORCID<sup>®</sup> 0000-0002-8944-5035)

이형철: 한양대학교 전기생체공학부 교수(hclee@hanyang.ac.kr, ORCID<sup>®</sup> 0000-0002-7640-9801)

※ 본 연구는 산업통상자원부 산업핵심기술개발사업 “혼합현실 기반 자율주행 부품 및 시스템 평가 기술개발” 과제의 지원으로 수행되었음 (과제번호: 20014478).

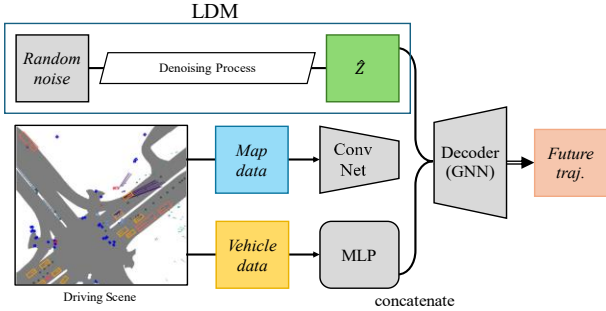


그림 1. Latent diffusion이 적용된 시나리오 생성.  
Fig. 1. Latent diffusion applied to scenario generation.

생성하면 차량들 간의 상호작용이 반영되어 현실성을 향상시킬 수 있을 뿐 아니라, 대량의 시나리오 생성이 가능하다. 하지만, 생성된 시나리오는 연속적인 latent space를 따르기 때문에 원래의 교통 데이터와 크게 다르지 않아 다양성 측면에서 한계가 생긴다.

본 논문은 잠재 확산 모델(latent diffusion model, LDM) [20] 기반의 시나리오 생성 방식을 제안하여 기존의 VAE 기반 시나리오 생성 모델의 다양성 한계를 극복하고자 한다. 기존의 VAE 기반의 시나리오 생성 모델로부터 latent space를 획득하고 LDM을 학습시킨다. 그 후, 그림 1과 같이 LDM의 노이즈 제거 과정에 랜덤으로 생성된 노이즈를 통과시킴으로써 새로운 latent space를 생성한다. 마지막으로, LDM을 통해 생성된 latent space는 VAE의 decoder 입력이 되어 도로 상황에 맞는 자연스럽고 다양한 시나리오의 생성을 가능하게 한다.

본 논문은 다음과 같이 구성된다. 2장에서는 VAE 기반 교통 모델에 대해 다루었고, 3장에서는 이를 활용한 LDM에 대해 다루었다. 4장에서는 생성된 시나리오가 자연스러운 교통 흐름을 이루는지를 확인하였고, 기존의 VAE 기반으로 생성된 시나리오와 LDM 기반으로 생성된 시나리오를 비교하여 다양성 측면과 ADS 검증의 효율성 측면에서 결과를 분석하였다.

II. 교통 모델

LDM을 학습시키기 위해 우선 교통 모델의 latent space를 얻을 수 있는 VAE 모델이 필요하다. 본 논문은 Rempe et al. [15]이 제안한 VAE 모델을 활용하였다. 이는 nuScenes dataset [21]에서 조감도로 기록된 각 차량의 경계 박스 (bounding box)와 궤적, 도로 데이터를 이용하여 VAE 모델을 학습시켰다.

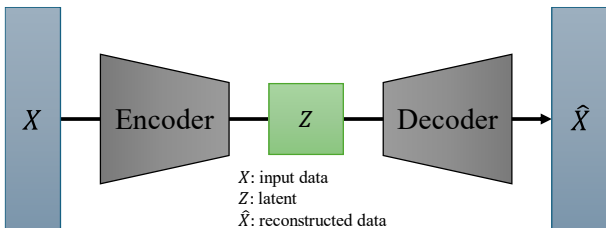


그림 2. 변분 오토인코더.  
Fig. 2. Variational autoencoder.

1. 데이터 셋

nuScenes dataset은 대규모의 실차 주행 데이터 셋이다. 6대의 카메라, 5대의 레이더, 1대의 라이다로 360도 주변이 모두 기록 되어있다. 연속하는 주행 데이터에서 주행 데이터를 20초 단위로 분할한 scene이라는 단위로 데이터를 제공한다. scene에는 차차와 주변차량들의 경로, 주행 가능 지역, 차선 등의 도로 데이터가 2Hz로 기록되어 있고, 총 1000개가 존재한다. Scene들을 각각 8초의 시나리오로 가공하여 2초의 입력으로 6초를 예측할 수 있도록 학습시켰다.

2. VAE

Latent diffusion을 위해서는 데이터를 latent space로 축소할 필요가 있고, 이를 위해서 VAE가 필요하다. VAE는 그림 2와 같이 encoder를 이용해 입력으로부터 latent space를 도출하고, decoder를 통해 입력과 유사하지만 새로운 데이터를 만들어 내는 생성형 모델의 한 종류이다.

VAE에서 입력으로부터 latent space를 도출하는 부분을 encoder라 한다(그림 3). Encoder의 입력은 각 차량의 context feature이다. 이는 각 차량의 과거 궤적 정보, 도로 정보, 차량 정보로부터 추출한 특징 정보들로 구성된다. 차량의 과거 궤적 정보는 차량의 2D 좌표( $x, y$ ), 차량 헤딩의 단위 벡터( $\theta^x, \theta^y$ ), 속도( $v$ ), 요 레이트( $\theta$ )를 포함하고 있다. 도로정보는 nuScenes에서 제공하는 drivable area, carpark area, road divider, lane divider, 네 개의 레이어 정보를 포함하고 있다. 마지막으로, 차량정보는 각 차량의 차종정보(상용차량/승용차량)와 bounding box 크기를 포함한다.

과거 궤적의 특징 정보를 얻기 위해서 과거 4스텝 동안의 궤적 정보와 차량 정보에 다층 퍼셉트론(multi-layer perceptron, MLP)을 적용하였다. MLP를 거침으로써 64의 크기를 갖는 과거 궤적의 특징 정보를 획득하였다. 도로의 특징 정보를 얻기 위해서는 합성곱 신경망(convolution neural network, CNN)과 완전 연결 계층(fully connected layer, FC layer)을 적용하였다. CNN과 FC layer의 결과로 64의 크기를 갖는 도로의 특징 정보를

표 1. Context feature의 구성 요소.

Table 1. Context feature's elements.

	Data
Past trajectory	$x, y, \theta^x, \theta^y, v, \theta$
Map	drivable area, carpark area, road divider, lane divider
Vehicle info	Semantic class(car/truck), bounding box length/width( $l, w$ )

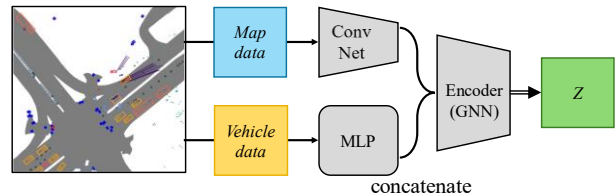


그림 3. VAE의 인코더.  
Fig. 3. Encoder of the VAE.

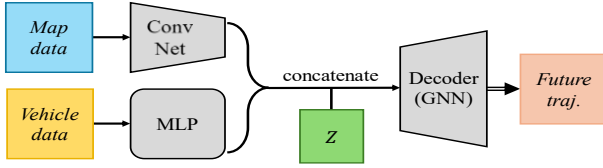


그림 4. VAE의 디코더.

Fig. 4. Decoder of the VAE.

획득하였다. 획득한 특징 정보들을 차종정보와 concatenate하여 130의 크기를 갖는 context feature를 획득하였다.

Encoder는 MLP와 그래프 신경망(graph neural network, GNN) [22]으로 구성되어 있다. GNN을 적용하면 각 차량 간의 상대 거리와 상대 헤딩 정보를 신경망에 반영할 수 있다. Context feature에 GNN을 적용함으로써 차량 간의 상대정보를 latent space에 반영하고, 최종적으로 MLP를 적용하여 32의 크기를 갖는 latent space를 획득할 수 있다.

VAE에서 latent space로부터 새로운 데이터를 생성하는 부분을 decoder라 한다(그림 4). 이 과정에서 생성된 새로운 데이터는 입력과 유사한 형태를 가진다. Decoder의 입력은 그림 4에 제시된 바와 같이 encoder의 입력에 latent space가 추가된 형태이다.

Decoder는 encoder와 동일하게 MLP와 GNN으로 구성되어 있다. Decoder를 통해 각 차량의 가속도와 요 레이트의 미분 값을 구한다. 구한 값들은 kinematic 모델의 상태 입력으로 사용되어 미래 경로를 예측하는데 활용된다. 동일한 과정을 반복적으로 수행하면서 과거의 정보를 갱신하고, 자기 회귀적으로 연속된 미래 경로를 예측한다. 최종적으로, 연속하는 미래 경로를 연결함으로써 시나리오를 생성할 수 있다.

### III. 시나리오 생성

확산 모델(diffusion model)은 정규 분포의 노이즈에서 단계적으로 노이즈를 제거하여 원래의 데이터로 복원하는 과정에서 데이터를 학습하도록 설계된 확률적 생성 모델이다. 데이터에 노이즈를 추가하는 정방향 확산 과정(forward diffusion process)과 노이즈를 제거하는 역방향 확산 과정(reverse diffusion process)으로 나뉘며 실제 노이즈와 모델이 예측한 노이즈를 최소화하는 방식으로 학습이 진행된다. LDM은 기존의 확산 과정을 데이터가 아닌 latent space에서 진행함으로써 diffusion model의 단점인 높은 계산 비용과 학습시간을 크게 감소시킨 모델이다. 또한, latent space를 학습함으로써 데이터의 주요 특징을 보존하여 기존의 모델 대비 정교한 데이터를 생성이 가능하다 장점이 있다[23-24].

VAE 기반의 시나리오 생성 모델은 기존의 차량 정보에 기반한 latent space를 생성한다. 하지만, LDM을 적용하면 주어

표 2. 생성된 시나리오의 결과.

Table 2. Results of generated scenarios.

Feature Model	Mean pairwise distance (m)	Mean Env. collision (%)	Mean vehicle collision (%)	Mean position min ADE (m)	Mean position min FDE (m)	Mean angle min ADE (deg)	Mean angle min FDE (deg)
VAE	1.17	3.28	2.14	1.74	3.82	2.60	3.86
LDM	2.69	2.54	5.57	1.90	4.29	2.78	3.83

진 기존의 차량 경로에서 벗어난, 전체 데이터의 특징이 학습된 latent space를 생성할 수 있다. LDM을 통해 가우시안 노이즈로부터 기존의 경로에서 벗어난 정교하고 다양한 시나리오의 생성이 가능하다.

#### 1. LDM 학습 과정

노이즈 스케줄을  $\text{uniform}[\beta_0, \dots, \beta_T]$ 으로 구성한다. 이때,  $\beta_0=1e-4$ ,  $\beta_T=0.02$ ,  $T=1000$ 의 값을 갖는다. 그림 3의 과정으로 획득한 latent space에 노이즈 스케줄에 따라 스텝마다 노이즈를 추가하는 forward diffusion( $q$ ) 과정을 거친다.  $\bar{\alpha}_t$ 는 (1)과 같이 구성되어  $t$  스텝을 거쳤을 때, 원본 데이터의 보존 정도를 나타내고, 이때 latent space( $x_t$ )는 (2)의 가우시안 분포로 나타난다. 마지막 스텝을 거친 latent space는 그림 5 우측과 같이 가우시안 노이즈 형태를 띄게 된다.

$$\alpha_t = 1 - \beta_t, \quad \bar{\alpha}_t = \prod_{s=1}^t (1 - \beta_s) \quad (1)$$

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)\mathbf{I}) \quad (2)$$

오염된 데이터를 기존의 latent space와 같은 형태로 되돌리기 위해 reverse diffusion( $p$ ) 과정을 거친다.  $t$  스텝을 거친 노이즈는 (3)의 가우시안 분포를 갖게 된다. (4)를 통해 분포의 평균( $\mu_\theta$ )을 예측할 수 있고  $\sigma_t$ 는 노이즈 스케줄에 의해 결정된다.

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2\mathbf{I}) \quad (3)$$

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) \quad (4)$$

이때,  $\epsilon_\theta$ 는 노이즈를 학습하는 함수로, Resnet [25]과 self-attention [26]을 이용한 U-Net [27] 구조로 구성하였다. 함수의 파라미터( $\theta$ )를 학습시키기 위한 손실함수는 아래와 같다.

$$\mathcal{L}_{ldm} = \mathbb{E}_{\epsilon \sim \mathcal{N}(0, \mathbf{I})} \left( \left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t) \right\|^2 \right) \quad (5)$$

#### 2. Inference

학습한 LDM을 이용하여 데이터 셋에 표현되어 있지 않은 새로운 시나리오를 생성한다. 그림 5의 우측과 같이 가우시안 노이즈에 reverse diffusion 적용하여 새로운 latent space를 생성

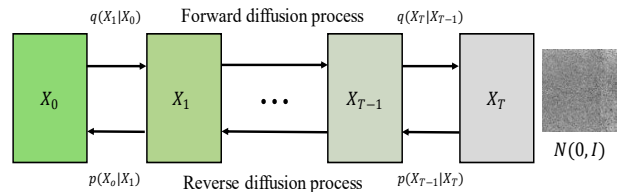


그림 5. 디퓨전 모델.

Fig. 5. Diffusion model.

한다. 생성된 latent space는 그림 4와 같이 VAE decoder의 입력이 되어 기존의 경로에서 벗어난 새로운 시나리오를 생성한다.

#### IV. 학습 결과 및 분석

##### 1. 학습 환경 및 결과

본 논문에서는 python 기반 pytorch 1.11 [28]을 이용하여 LDM을 구현하였다. 학습에 NVIDIA GeForce RTX 3060 GPU와 Intel i7-10700F@2.9GHz CPU를 이용하였다. 300 에폭동안 OneCycleLR scheduler (최대 학습 계수=1e-03)의 학습 계수를 가지는 AdamW optimizer [29]로 최적화를 진행하였다. 이때 배치 사이즈는 128로 구성하였다. nuScenes dataset을 6:2:2의 비율로 나누어 각각 학습/검증/시험 데이터 셋으로 분할하였다. 학습결과는 그림 6과 같다. 학습 손실(loss)은 약 0.028로 수렴하였고, 약 4시간이 소요되었다. 검증 손실이 가장 낮게 나타난 290번째 에폭의 모델을 사용하여 시험 데이터 셋에서 latent space를 2000개 생성하였다. 약 100분동안 생성된 latent space와 decoder를 이용하여 총 3619개의 도로 상황에 대해 시나리오를 10개씩 생성하였고 VAE 기반으로 생성된 시나리오와의 비교를 통해 결과를 분석하였다.

##### 2. 생성 시나리오

각 시험 데이터 셋에 존재하는 차량 한 대당 6초간의 경로로 10개씩 생성하였다. 표 2는 VAE와 LDM으로 생성된 시나리오들의 성능을 비교하여 표로 나타낸 것이다. 표 3에 각 인자들의 의미를 나타내었다. 이때, ADE (Average Displacement Error)는 기존 경로와 차이의 평균을, FDE (Final Displacement Error)는 기존 경로의 최종 값과의 차이를 의미한다.

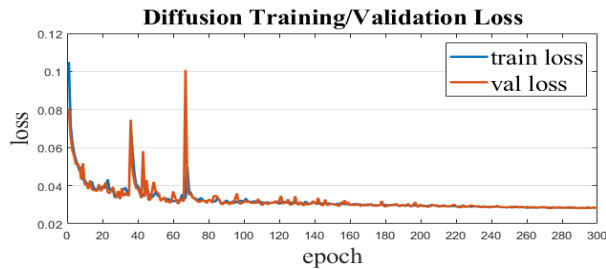


그림 6. LDM의 학습 및 검증 손실.  
Fig. 6. Training and validation loss of LDM.

표 3. 성능비교에 사용된 인자들.

Table 3. Parameters used for comparison.

Parameter	Description
Mean pairwise distance	생성된 경로 샘플 간 거리의 평균
Mean Env. collision	환경과 충돌한 경로 비율의 평균
Mean vehicle collision	차량 간의 충돌 경로 비율의 평균
Mean position min ADE	10개의 경로 샘플 중 좌표의 ADE가 가장 작은 것들의 평균
Mean position min FDE	10개의 경로 샘플 중 좌표의 FDE가 가장 작은 것들의 평균
Mean angle min ADE	10개의 경로 샘플 중 headings의 ADE가 가장 작은 것들의 평균
Mean angle min FDE	10개의 경로 샘플 중 headings의 FDE가 가장 작은 것들의 평균

표 2에서 mean pairwise distance는 LDM 기반의 시나리오들이 더 큰 값을 가진 것으로 나타났으며, 시나리오 간의 다양성을 확보한 것이라 할 수 있다. 또한, mean Env. collision을 확인해보았을 때 LDM으로 생성한 시나리오들이 더 작은 값을 갖는 것으로 보인다. 이는 환경과의 충돌로 사용하지 못하는 시나리오의 비율이 낮아져 시나리오 생성의 효율이 향상된 것이라 할 수 있다. Mean vehicle collision의 경우, LDM에서 그 비율이 오히려 높지만, 차량 간 충돌 시나리오는 자율주행의 충돌 회피 능력 검증에 이용할 수 있어 검증용 시나리오로 가치가 있다. 마지막으로 ADE와 FDE에서는 VAE 기반의 시나리오들이 약간의 우세한 성능을 보인다. 기존의 경로와 연관이 있는 지표의 특성상, 다양성 확보를 위해 성능이 약화된 것으로 판단된다.

표 2에서 mean pairwise distance는 LDM 기반의 시나리오들이 더 큰 값을 가진 것으로 나타났으며, 시나리오 간의 다양성을 확보한 것이라 할 수 있다. 또한, mean Env. collision을 확인해보았을 때 LDM으로 생성한 시나리오들이 더 작은 값을 갖는 것으로 보인다. 이는 환경과의 충돌로 사용하지 못하는 시나리오의 비율이 낮아져 시나리오 생성의 효율이 향상된 것이라 할 수 있다. Mean vehicle collision의 경우, LDM에서 그 비율이 오히려 높지만, 차량 간 충돌 시나리오는 자율주행의 충돌 회피 능력 검증에 이용할 수 있어 검증용 시나리오로 가치가 있다. 마지막으로 ADE와 FDE에서는 VAE 기반의 시나리오들이 약간의 우세한 성능을 보인다. 기존의 경로와 연관이 있는 지표의 특성상, 다양성 확보를 위해 성능이 약화된 것으로 판단된다.

그림 7과 그림 8에 생성된 시나리오의 예시를 나타냈다. 각 그림에서 nuScenes dataset의 도로 레이아웃 drivable area, road divider, lane divider와 생성된 시나리오의 궤적 그리고 각 차량의 기존 경로 표현하였다. 도로의 레이아웃은 각 회색, 주황색, 노란색으로 나타내었다. 각 차량의 생성된 시나리오는 bounding box의 색상과 동일한 색상의 선으로, 기존 경로는 흰색의 선으로 표시하였다. 또, 매 스텝의 계산 위치마다 빨간색~보라색의 점으로 시간의 경과를 표현했다.

그림 7의 경우, 3지 교차로에서 그림 위쪽 차량의 좌회전 신호 상황이다. 또한, 상업(공사)차량이 도로의 진행 방향과

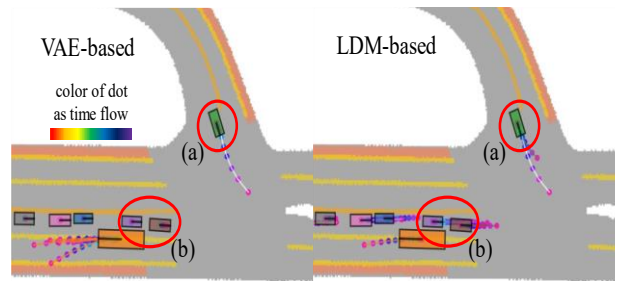


그림 7. 시나리오 예시 1.  
Fig. 7 Example of scenario 1.

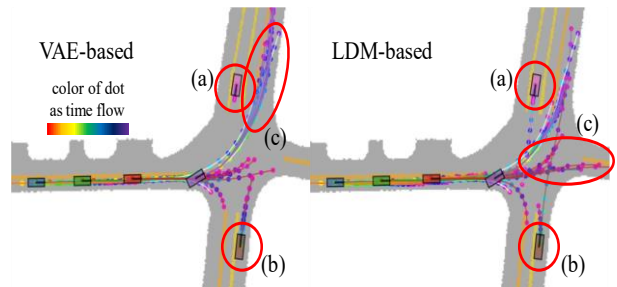


그림 8. 시나리오 예시 2.  
Fig. 8 Example of scenario 2.

반대로 정차 중에 있다. VAE 기반의 시나리오와 LDM 기반의 시나리오의 mean pairwise distance는 각각 0.04m, 0.74m로 나타났다. 그림 7의 (a)에서 LDM 기반 시나리오는 기존 경로에서 벗어나 더 많은 시나리오를 생성하고 있는 것을 확인할 수 있고, (b)에서는 LDM기반의 시나리오에서 더 많은 대수의 차량들에 대해 정지해 있는 기존의 경로에서 벗어나 동작을 하고 있는 시나리오를 생성한 것을 확인할 수 있다.

그림 8의 경우, 4지 교차로에서 그림 좌측의 차량들의 좌회전 및 직진 신호 상황이다. VAE 기반의 시나리오와 LDM 기반 시나리오의 mean pairwise distance는 각각 2.18m, 4.95m로 나타났다. 그림 8의 (a)와 (b)에서 LDM이 더 다양한 시나리오를 생성한 것을 확인할 수 있다. 특히, (c)에서, VAE 기반의 시나리오는 기존의 경로에서 벗어나지 못하고 흰색의 기존 경로 주변에 시나리오가 주로 생성된 반면, LDM 기반의 시나리오는 기존 경로에서 벗어난 시나리오가 생성된 것을 확인할 수 있다.

두 예시에서 모두 LDM 기반의 생성 방식이 기존 경로에서 벗어난 새로운 시나리오를 생성함을 확인할 수 있다. 신호 정보가 포함되지 않아 신호 상황과는 일치하지 않지만 도로 상황에 맞는 자연스러운 교통 흐름 역시 확인할 수 있다. 이로 미루어 보아, ADE와 FDE의 감소의 영향은 크지 않은 것으로 확인된다. 결론적으로 LDM 기반의 시나리오 생성 방식이 VAE 기반의 시나리오 생성 방식에 비해 현실성과 자연스러움에서 뒤쳐지지 않으면서도, 다양한 시나리오를 생성함을 확인하였다.

## V. 결론

본 논문에서는 LDM 기반의 주행 시나리오 생성모델을 제시하였다. 기존의 방식인 VAE 기반의 시나리오 생성 방식의 latent space를 학습함으로써 LDM 기반의 시나리오들을 생성하였다. 기존의 방식과 대비해 ADE, FDE 지표에서는 성능이 저하되었지만, mean Env. collision, pairwise distance 지표에서는 기존의 방식보다 우수한 성능을 보였다. 지표들의 분석에서 LDM 기반의 시나리오 생성 방식이 다양성을 확보한 것을 확인할 수 있었다. 또한, 두 방식의 실제 예시들을 통해 성능이 저하된 지표들의 영향과 기존의 경로에서 벗어난 시나리오들의 생성을 확인하였다. 결과적으로 LDM 기반의 방식이 VAE 기반의 방식에 비해 다양한 시나리오를 생성함으로써 ADS 검증 시나리오 설계에 있어 더 풍부한 주행상황을 반영할 수 있음을 확인하였다.

다만, 본 연구는 임의로 주행 시나리오들을 생성하는 과정만을 담고 있다. ADS에서 문제가 되는 edge-case나 특정 상황을 생성하기 위해서는 도로나 과거 경로에 변화를 주는 수밖에 없다. 특정 상황에 대한 확실한 검증을 위해 조건에 따라 시나리오들을 생성하는 방식에 대해 추가 연구할 예정이다.

## REFERENCES

[1] G. H. Jo and T. H. Park, "Deep reinforcement learning-based decision making of lane change considering rear vehicle deceleration," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 28, no. 6, pp. 602-607, Jun. 2022.

[2] V. Mahajan, C. Katrakazas, and C. Antoniou "Prediction of lane-

changing maneuvers with automatic labeling and deep learning," *Transportation Research Record*, vol. 2674, no. 7, pp. 336-347, Jul. 2020.

[3] Waymo, Google Self-Driving Car Project. <https://waymo.com/>, 2024.

[4] I. Mun, K. Cho, and H. Lee, "Development of a car-following model based on reference governor for autonomous driving system verification scenario," *Transaction of the Korean Society of Automotive Engineers (in Korean)*, vol. 31, no. 2, pp. 107-115, Feb. 2023.

[5] G. W. Kim and J. Y. Kang, "3D space object and road detection for autonomous vehicles using monocular camera images and deep learning algorithms," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 30, no. 7, pp. 677-684, Jul. 2024.

[6] S. Kuutti, R. Bowden, Y. Jin, P. Barber and S. Fallah, "A survey of deep learning applications to autonomous vehicle control," *IEEE Transactions on Intelligent Transportation Systems* vol. 22, no. 2, pp. 712-733, 2021.

[7] H. Alghodhaifi and S. Lakshmanan, "Autonomous vehicle evaluation: A comprehensive survey on modeling and simulation approaches," *IEEE Access*, vol. 9, pp. 151531-151566, 2021.

[8] D. Zhao, H. Lam, H. Peng, S. Bao, D. J. LeBlanc, K. Nobukawa, and C. S. Pan, "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques," *IEEE Transactions on Intelligent Transportation Systems.*, vol. 18, no. 3, pp. 595-607, Mar. 2017.

[9] A. M. Nascimento, L. F. Vismari, C. B. S. T. Molina, P. S. Cugnasca, J. B. Camargo, J. R. de Almedia, R. Inam, E. Fersman, M. V. Marquezini, and A. Y. Hata, "A systematic literature review about the impact of artificial intelligence on autonomous vehicle safety," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 12, pp. 4928-4946, Dec. 2019.

[10] Y. Abeysirigoonawardena, F. Shkurti, and G. Dudek, "Generating adversarial driving scenarios in high-fidelity simulators," *2019 IEEE International Conference on Robotics and Automation*, pp. 8271-8277, 2019.

[11] L. Bergamini, Y. Ye, O. Scheel, L. Chen, C. Hu, L. D. Pero, B. Osinski, H. Grimmet, and P. Ondruska, "Simnet: Learning reactive self-driving simulations from real-world observations," *2021 IEEE International Conference on Robotics and Automation*, pp. 5119-5125, 2021.

[12] S. Suo, S. Regalado, S. Casas, and R. Urtasun, "TrafficSim: Learning to simulate realistic multiagent behaviors," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10400-10409, 2021.

[13] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y. -P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wiessner, "Microscopic Traffic Simulation using SUMO," *21st International Conference on Intelligent Transportation Systems*, HI, USA, pp. 2575-2582, Nov. 2018.

[14] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," *Conference on Robot Learning*, California, USA, pp. 1-16, Nov. 2017.

[15] D. Remppe, J. Phillion, L. J. Guibas, S. Fidler, and O. Litany, "Generating useful accident-prone driving scenarios via a learned traffic prior," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17305-17315, Jun. 2022.

- [16] Z. Zhong, D. Rempe, D. Xu, Y. Chen, S. Veer, T. Che, B. Ray, and M. Pavone, "Guided conditional diffusion for controllable traffic simulation," *IEEE International Conference on Robotics and Automation*, pp. 3560-3566, 2023.
- [17] E. Pronovost, M. R. Ganesina, N. Hendy, Z. Wang, A. Morales, K. Wang, and N. Roy, "Scenario diffusion: Controllable driving scenario generation with diffusion," *Advances in Neural Information Processing Systems*, vol. 36, pp. 68873-68894, 2023.
- [18] Y. Yuan, X. Weng, Y. Ou, and K. M. Kitani, "Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting," *IEEE/CVF International Conference on Computer Vision*, pp. 9813-9823, Mar. 2021.
- [19] D. P. Kingma and M. Welling, "Auto-encoding variational bays," *International Conference on Learning Representations*, 2014.
- [20] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684-10695, 2022.
- [21] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11621-11631, 2020.
- [22] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61-80, Jan. 2009.
- [23] F. A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10850-10869, Sep. 2023.
- [24] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, W. Zhang, B. Cui, and M. H. Yang, "Diffusion models: A comprehensive survey of methods and applications," *ACM Computing Surveys*, vol. 56, no. 4, pp. 1-39, Apr. 2024.
- [25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.
- [26] D. Bahdanau, "Neural machine translation by jointly learning to align and translate," *International Conference on Learning Representations*, 2015.
- [27] O. Ronneberger, P. Fischer and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Medical Image Computing and Computer-assisted Intervention*, vol. 9351, pp. 234-241, 2015.
- [28] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala "PyTorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, pp. 8024-8035, 2019.
- [29] I. Loshchilov, "Decoupled weight decay regularization," *International Conference on Learning Representations*, 2019.



### 전 명 환

2023년 한양대학교 전기생체공학부 전기공학과(공학사). 2023년~현재 한양대학교 대학원 전기공학과 석사과정 재학 중. 관심분야는 자율주행, 인공지능.



### 조 건 희

2017년 한국기술교육대학교 기계공학과(공학사). 2019년 한국기술교육대학교 기계공학과(공학석사). 2019년~현재 한양대학교 대학원 전기공학과 박사과정 재학 중. 관심분야는 자율주행, 최적제어, 차량 동역학 및 제어.



### 이 형 철

1988년 서울대학교 기계공학과(공학사). 1990년 서울대학교 대학원 기계공학과(공학석사). 1997년 University of California, Berkeley, Mechanical Engineering(공학박사). 2004~현재 한양대학교 서울캠퍼스 전기생체공학부 교수. 관심 분야는 적응 제어 및 비선형 제어, 임베디드 시스템, 차량 제어 응용 및 차량 동역학 등.