南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

CrashSimGen: Generating Dangerous Scenarios with Diffusion Models for self-Driving

CS329 Machine Learning (H)

Project of Scenario Generation via Diffusion Model

12211612 Haibin Lai, 12211717 Wenkun Shen

2024 Dec 8th

1 Abstract

CrashSimGen is a project that generates critical road scenarios using diffusion models for autonomous driving risk assessment. The project begins by encoding a large set of driving scenes as image data, and training a dangerous scenario recognizer that automatically identifies and labels potentially hazardous situations. These recognized dangerous scene thumbnails are then fed into a diffusion model to generate new dangerous scene thumbnails.

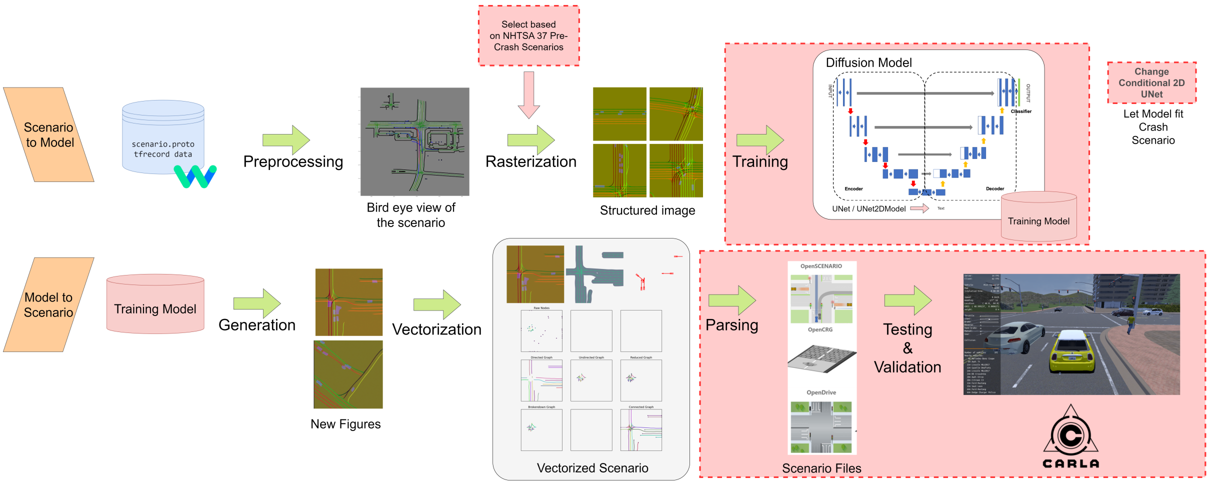

Figure 1: Project Workflow and Main Contribution

To transform the generated thumbnails into a format usable by autonomous driving systems, we try to develop a parser that utilizes OpenCV image processing tools to expand the scene thumbnails into OpenDrive formatted driving scenes. These scenes are then imported into the autonomous driving simulation software Carla, where they are tested to assess and improve the vehicle's response to dangerous scenarios in a simulated environment.

2 Introduction

In this part, we will show the **Background and Significance** in section 2.1 of data-driven approaches in autonomous driving research, and give a brief **analysis of Current Research Status** in section 2.2.

2.1 Data-driven Critical Driving Scenario Generation

In recent years, data-driven methods, particularly deep learning models, have become increasingly popular in the autonomous driving research community. As figure 2 displays, these models rely heavily on vast amounts of data collected from real-world driving scenarios to improve their performance[13]. The key advantage of such approaches lies in their ability to generalize across diverse situations, making them highly effective for tasks such as perception, decision-making, and control in autonomous systems.

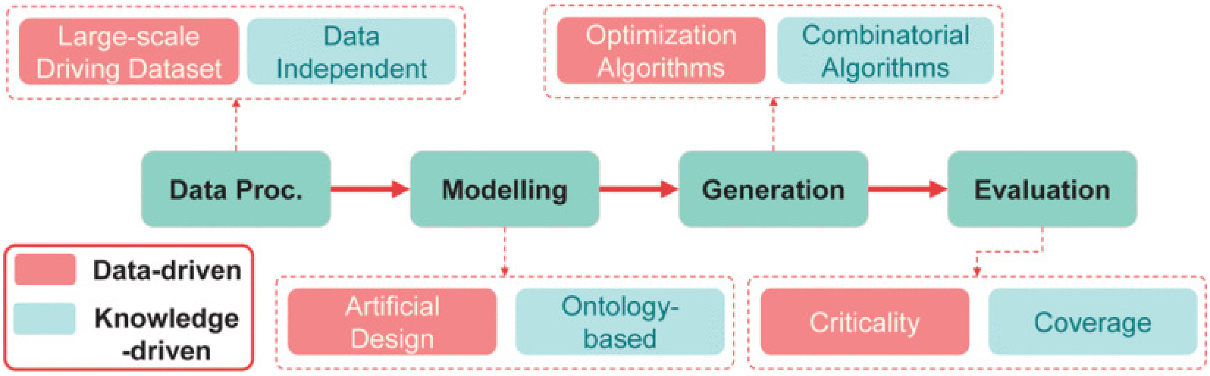

Figure 2: Comparison between data-driven and knowledge-driven scenario generations.[8]

Nowadays as the development of AI and self-driving models continues to grow, the significance of safety-critical data-driven techniques becomes ever more apparent , as they enable the development of systems that are robust to real-world complexity and variability [20].

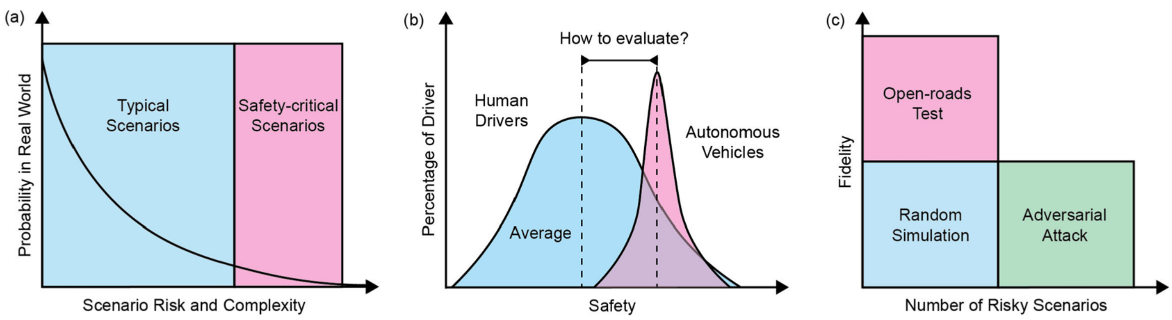

Figure 3: The importance and high value of critical scenario [5]

The significance of studying critical dangerous self-driving scenarios is multifaceted. Firstly, the safety and reliability of autonomous vehicles (AVs) hinge on their ability to perceive, understand, and react appropriately to a wide array of traffic situations.[2]Real-world data, while valuable, is often limited in its diversity and quantity, especially when it comes to rare as figure 4 shows, but critical scenarios that are pivotal for testing the limits of AV safety. This is where the generation of diverse and realistic driving scenarios becomes crucial, as it allows for the simulation of these rare events in a controlled environment, thereby enhancing the robustness of AV systems [3].

Secondly, the generation of critical self-driving scenarios facilitates the development and validation of AV technologies in a cost-effective and time-efficient manner. Physical testing of every possible scenario is impractical due to the sheer scale and variability of traffic conditions.[22] Data-driven approaches on various scenarios, on the other hand, enable the creation of synthetic scenarios that can mimic real-world complexities, offering a scalable solution for AV testing and evaluation [7].

Furthermore, the study of critical self-driving scenarios is essential for advancing the field of artificial intelligence and machine learning. It provides a platform for testing and refining algorithms under conditions that are representative of the real world, thus pushing the boundaries of what is achievable with current technology[9]. The ability to generate diverse and realistic scenarios is not just about creating more data; it's about creating the right kind of data—data that can effectively train AV systems to handle uncertainty and make reliable decisions in unpredictable environments [6].

In summary, the study of self-driving scenarios is worth pursuing due to its direct impact on the safety, reliability, and advancement of autonomous driving technology.[15] It addresses the critical need for diverse and extensive data in AV development, offering a solution to the limitations of real-world data collection and providing a robust framework for testing AV systems under a multitude of conditions. The research is not only significant for the automotive industry but also for the broader field of AI, as it contributes to the development of more intelligent and adaptive systems capable of navigating the complexities of the real world [16].

2.2 Current Research Status

There have been several ways on data-driven critical scenarios generation with the rapid development of Generative Model as figure. These methods are mainly divided into two parts.

$$\begin{aligned}\mathcal{L} &= \mathbf{E}_{q,t}\left[D_{KL}\left(q(\mathbf{F}_{t-1} \mid \mathbf{F}_t, \mathbf{F}_0) \parallel p_\theta(\mathbf{F}_{t-1} \mid \mathbf{F}_t)\right)\right] \\ &= \mathbf{E}_{q,t}\left[D_{KL}\left(\mathcal{N}(\mathbf{F}_{t-1}; \mu_q, \Sigma_q(t)) \parallel \mathcal{N}(\mathbf{F}_{t-1}; \mu_\theta, \Sigma_t)\right)\right] \\ &= \mathbf{E}_{q,t}\left[|||\mu_\theta - \mu_q||_2^2\right]\end{aligned}$$

First, it samples scenarios from dataset, then it will use an estimation model $p_\theta(x)$ to learn the distribution of scenarios, usually measured by minimizing KL-Divergence or maximizing the log-likelihood [5].

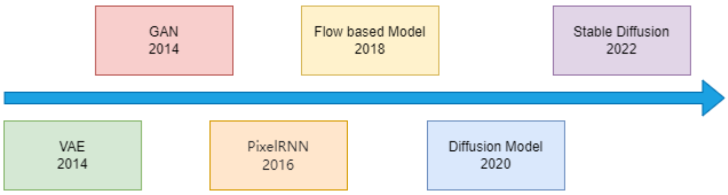

Figure 4: The Development of Generative Model

With the power of VAE, [4] learns a latent space of encounter trajectories and generates unseen scenarios by sampling from the latent space. However, with less understanding of the latent code, the generation is not controllable. As for the usage of GAN, [1] introduces recurrent models to generate realistic scenarios of highway lane changes. They use real-world data in the discriminator to help the improvement of the generator. SurfelGAN is proposed in [18] to directly generate point cloud data to represent scenarios from the view of the AV. However, GAN model needs more fine-tuning and maintains a bad KL-Divergence with the real critical data.

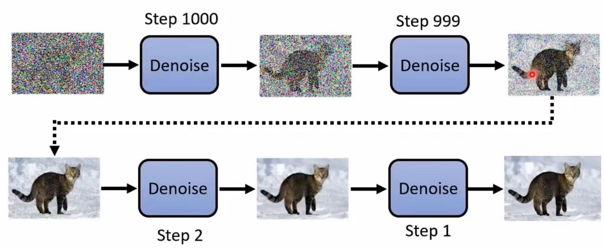

Figure 5: Diffusion Model's denoising example

In recent years more models like Diffusion Model [10] comes to the tip of the Generative Models, enumerating more strong and powerful models in critical scenario generation. [14] introduces DriveSceneGen, it uses a classical diffusion model to generate novel driving scenarios that align with real-world data distributions with diversity. However, most of the generative scenario haven't been denoised enough and most scenario are too common or trival rather than critical. [17] use Denoising Diffusion Probabilistic Models (DDPM) enhanced with Diffusion with Transformer (DiT) blocks generating both realistic and diverse trajectories. However their works mainly focus on trajectory generation rather than more generalize critical scenarios as NHTSA 37 defined. [21] provides both controllability and realism traffic model using diffusion model, but it focus on safety scenarios rather than dangerous scenarios. [19] leverages the capabilities of LLMs to generate safety-critical scenarios for autonomous vehicles. But LLMs needs instructions and can only generate scenarios that close to the given limit tokens, while the real world scenario may be more various. [11] proposed VBD that utilizes diffusion generative models to predict scene-consistent and controllable multi-agent interactions in closed-loop settings. [12] combine latent diffusion, object detection and trajectory regression to generate distributions of synthetic agent poses, orientations and trajectories simultaneously.

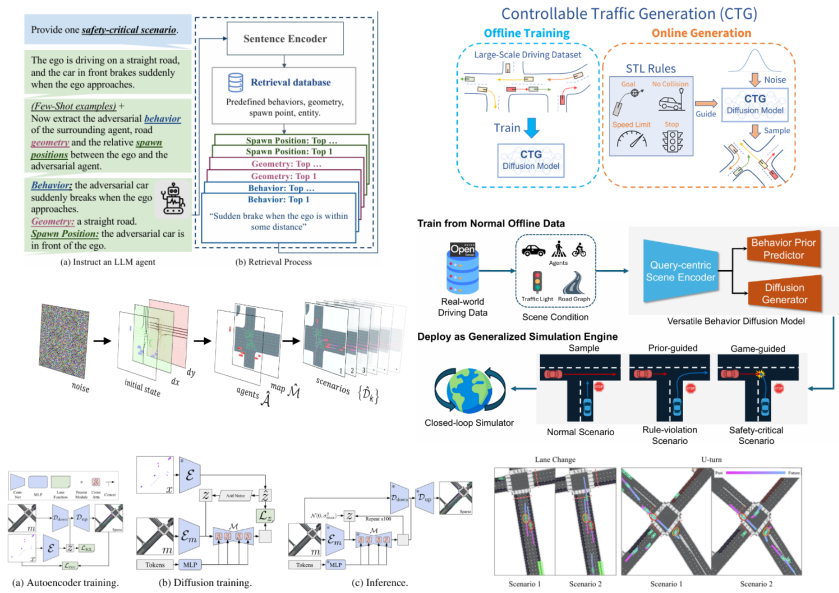

Figure 6: Different projects works on scenario generation using Diffusion Model

3 Related Work

Critical Scenario recognition. Critical Scenario Recognition in autonomous driving faces several challenges. First, accurately identifying and classifying complex road situations, such as sudden pedestrian crossings or emergency braking, requires advanced perception technologies. Second, handling the vast amount of sensor data from cameras, LiDAR, and radar and effectively fusing it to understand the environment is computationally demanding. Third, real-time processing is crucial to make immediate decisions and avoid accidents in high-risk situations. Fourth, predicting the behavior of dynamic objects, like other vehicles and pedestrians, is a difficult task due to their unpredictable nature. Finally, ensuring the system's reliability across a wide range of driving conditions, including rare or edge cases, remains a significant hurdle

Generative Artificial Intelligence. Generative Artificial Intelligence (Generative AI) refers to a subset of AI techniques that focus on creating new content, such as text, images, and scenarios, based on patterns learned from existing data. This is achieved through advanced models like Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Transformer-based models.

Diffusion Model. Diffusion models are built on probabilistic principles and Markov Chain, and there are a few key formulas that describe their core components: the forward diffusion process, the reverse process, and how they are trained.

We will show the mathmatical process of Diffusion Model in section 5.1 .

4 CrashSimGen Workflow

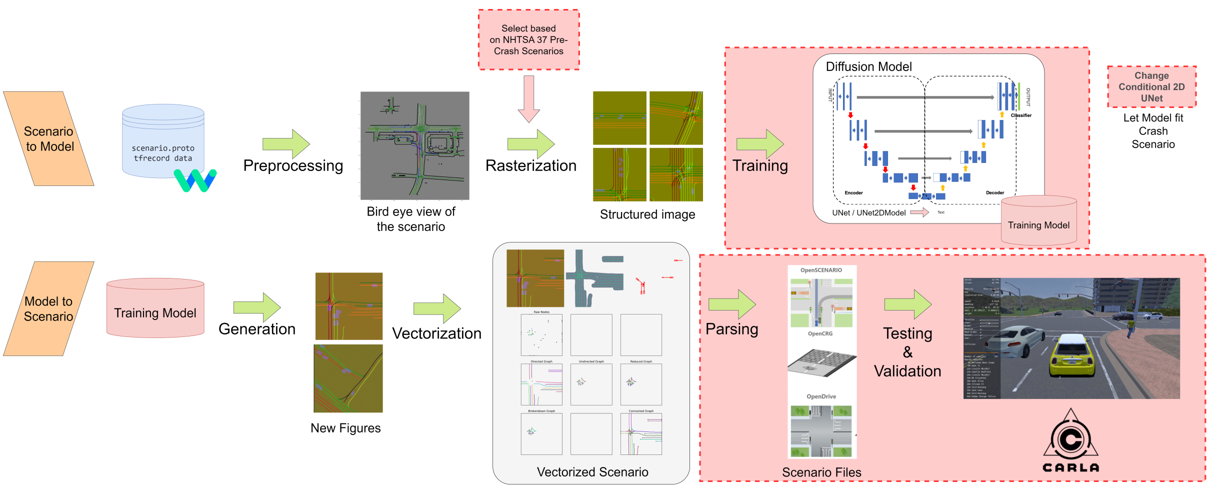

Figure 7: Project Workflow and Main Contribution

The CrashSimGen project focuses on generating critical and potentially hazardous road scenarios to assess the safety and performance of autonomous driving systems. The key objective is to enhance the ability of self-driving vehicles to respond to dangerous situations by simulating real-world risks in a controlled environment.

1. Dangerous Scenario Recognition on dataset

By encoding a large set of real-world driving scenes into image data, we plan to train or build a dangerous scenario recognizer. It can automatically identify and label scenes that are potentially hazardous for autonomous driving based on rules like NHTSA 37, or having many intersections with cars etc. These images represent various road scenarios, and the recognition system flags those that could pose a risk to vehicle safety.

2. Critial Scenario Diffusion Model Training

Once the dangerous scene thumbnails are identified, they are fed into a diffusion model. The model generates new thumbnails of dangerous road scenarios by utilizing its noise generation capability. This process helps create novel potential hazards that might not have been captured in the original dataset, providing additional scenarios for testing and evaluation. And We may change the model architecture, like adding Latent Diffusion Model or changing the architecture from current UNet2DModel to new Models.

3. Image Parsing and OpenDrive & OpenScenario Generation

To make the generated thumbnails usable for autonomous driving systems, we will develop a parser. This parser uses OpenCV, a popular image processing tool, to expand the scene thumbnails into OpenDrive-formatted driving scenarios. OpenDrive is a standardized format widely used in the autonomous driving industry to represent road networks and traffic environments. And OpenScenario can tell CARLA to imitate the scene based on the OpenDrive maps. They all can be generate from python package with information grabbed by OpenCV.

4. Testing and Evaluation in Simulation

The final step involves importing the generated and formatted dangerous driving scenes into Carla. In this simulated environment, we will test and evaluate current AI on vehicle's response to these hazardous scenarios. The simulation helps assess how well the autonomous driving system can handle and mitigate potential risks in real-world driving conditions.

5 Methodology and Experiment

In this part, we show our fundamental knowledge acquisition, including learning about models and its internal principles.

Then, we show the Deployment results of models.

5.1 Mathematical Principle of Diffusion Model

Diffusion models are built on probabilistic principles and Markov Chain . Inside it are three processes : the forward diffusion process, the reverse process, and training process.

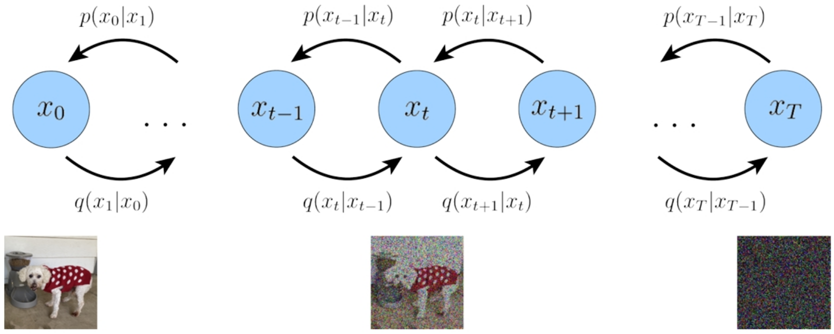

Figure 8: Diffusion Model's forward and reverse process

Forward Diffusion Process The forward diffusion process gradually adds noise to an image over a series of time steps. This process is modeled as a Markov chain that gradually corrupts a clean image into pure noise.

Let $\mathbf{x}_0$ be the original data sample (e.g., an image), and $\mathbf{x}_T$ be the noisy image at time step T (where T is the total number of steps). The forward process can be defined as:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$$

Here: β_t is a noise schedule (a small positive value). $\mathbf{x}_t$ is the data sample at time t, which becomes increasingly noisy as t increases. $\mathcal{N}(\mu, \sigma^2)$ denotes a Gaussian distribution with mean μ and variance σ^2.

The above equation models the addition of Gaussian noise at each time step, with the standard deviation controlled by β_t.

Total Forward Process (Corrupting the Data Over Time) Forward diffusion process Describes a single step in the process of adding noise, and Total Forward Process describes the entire sequence of transitions. The forward process is iterated over multiple time steps, resulting in the corrupted sample $\mathbf{x}_T$ after T steps. The final noisy sample is given by:

$$q(\mathbf{x}_T|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$

This equation expresses the total probability of transitioning from $\mathbf{x}_0$ to $\mathbf{x}_T$ via a series of noisy steps.

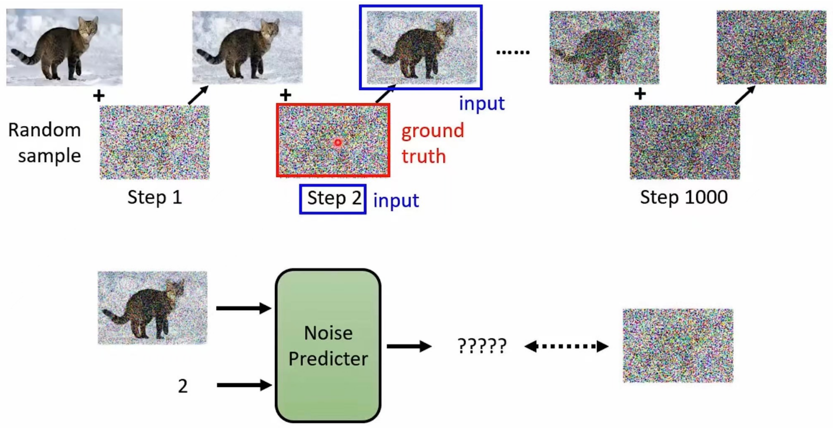

Figure 9: Diffusion Model's forward diffusion process

Reverse Process (Denoising the Sample) The reverse process is where the generative model learns to reverse the forward process, starting with random noise and gradually denoising it to recover the original data sample. The reverse process is modeled as a Markov chain:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sigma_\theta^2(t)\mathbf{I})$$

Where: $\mu_\theta(\mathbf{x}_t, t)$ is the mean of the predicted denoised image at time t, and it is learned by the model. $\sigma_\theta^2(t)$ is a learned variance at each step.

The reverse process is learned by the model and parameterized by a neural network (often a U-Net architecture).

Training Objective (Learning the Reverse Process) The goal of training the model is to learn the parameters θ of the reverse process such that it can accurately predict the denoised image at each step. To achieve this, we optimize a loss function that

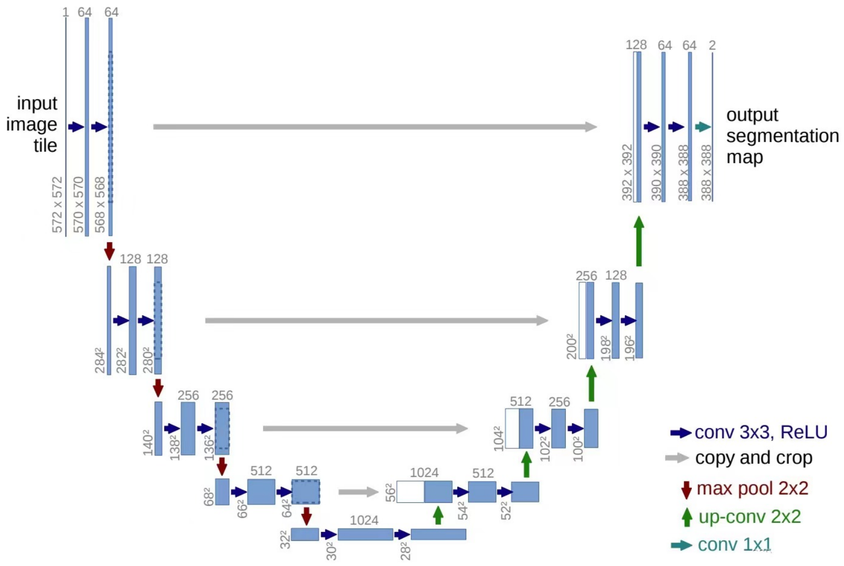

Figure 10: Diffusion Model's UNet architecture

minimizes the difference between the predicted denoised image and the true data sample.

The standard objective is based on variational inference and the evidence lower bound (ELBO). The loss function is typically the **denoising score matching** objective, which can be written as:

$$L(\theta) = E_{q(\mathbf{x}_0,\mathbf{x}_1,\ldots,\mathbf{x}_T)}\left[\sum_{t=1}^{T} \|\mathbf{x}_t - \hat{\mathbf{x}}_t\|^2\right]$$

Where: $\hat{\mathbf{x}}_t$ is the predicted denoised version of $\mathbf{x}_t$ at time step t. The expectation is taken over the distribution of all noisy data sequences $q(\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_T)$.

This loss function encourages the model to accurately recover the clean image from noisy samples at each time step.

Inference (Sampling from the Model) Once the model is trained, generating new data samples involves sampling from the reverse process, starting with pure noise $\mathbf{x}_T$ and iteratively applying the learned denoising steps:

$$\mathbf{x}_{T-1} \sim p_\theta(\mathbf{x}_{T-1}|\mathbf{x}_T)$$

This process is repeated from $t = T$ down to $t = 1$, generating a new sample $\mathbf{x}_0$ after T steps.

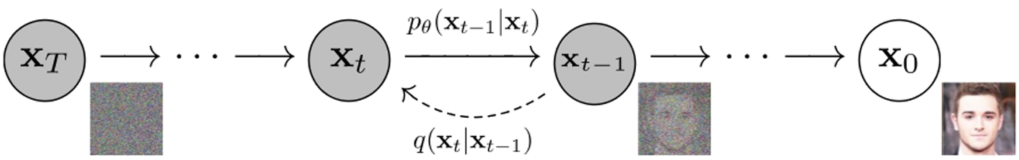

Figure 11: Diffusion Model's inference workflow

5.2 Completed Works

So far, we have done these works.

1. Dataset Acquisition

The Waymo Open Dataset is one of the biggest dataset in self-Drving. It is composed of two datasets - the Perception dataset with high resolution sensor data and labels for 2,030 scenes, and the Motion dataset with object trajectories and corresponding 3D maps for 103,354 scenes. In this project, we use Motion dataset to generate scene.

However, the data contains 4.5 TB, and the computer Network can't afford such large data and can't download it in a few weeks. To handle that, we build a **high performance network cluster** with the help of SUSTech Center of Computation for Science and Engineering (CCSE) as figure 12 shows.

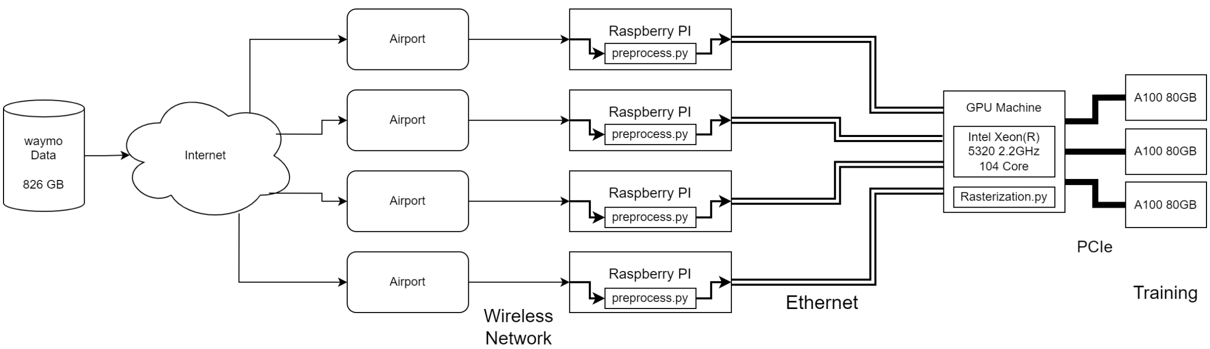

Figure 12: Our High Performance Cluster

By using the cluster, we reach a download rate of nearly 10 MB/s with good QoS and consistency. And we successfully download all the data into our Compute cluster.

2. Model Training

The author of [14] used 4 NVIDIA V100 16GB to train a specific Diffusion Model on 60000 images. To train our model with big memory requirement, we train our model on two cluster. The first cluster contains 1 NVIDIA A100 80GB GPU with PCIe 4.0 connected with 128GB CPU Memory. The second cluster contains 3 NVIDIA A100 80GB

GPU with PCIe 4.0 connected with 328 GB CPU Memory, with a NCCL communication library and Infiniband connection.

We conduct our experiment first with 60K images on 1 GPU. With model parameters of 56574595, batch size 128, 50 epoch, 1000 denoising iter per epoch. we train 70 hours for the entire training. If we plan to train on bigger critical data, we must move the process into the bigger cluster.

Figure 13: A100 GPU (Our Memory is 80GB)

3. Model Inference

So far, we have generate some of the good scenarios from our model. However, these scenarios comes out nearly every 100 generations. That means the mean generations quality is really low.

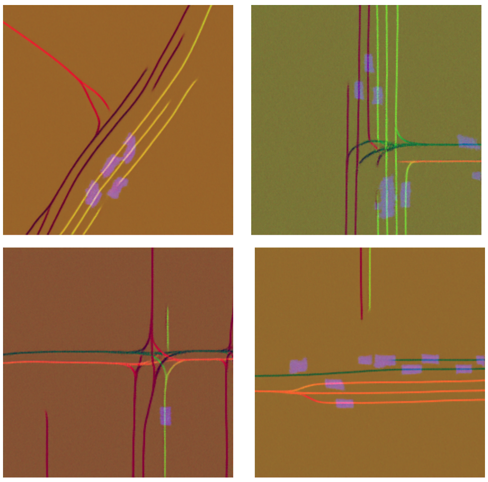

Figure 14: Some of our good generations of traffic scenario

4. Project building

We have built our repo based on current research. And we implement a simple Diffusion Model based on [14] for our project. We also finish data preprocessing and vectorization process. Our project can be seen at: https://github.com/HaibinLai/CrashSimGen.git

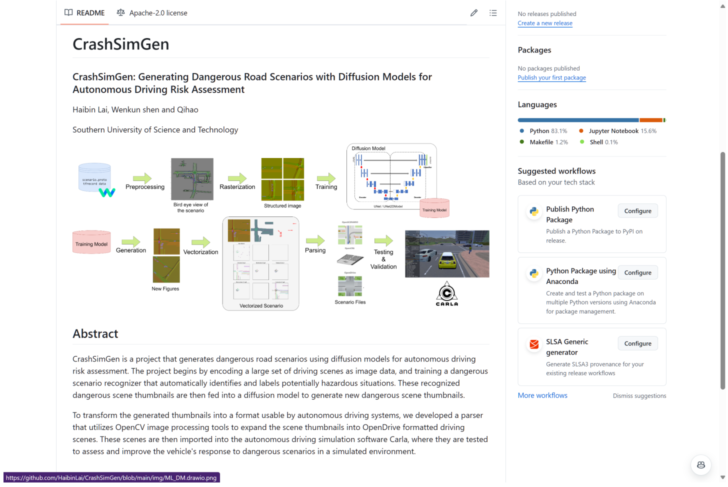

Figure 15: Our Github project

In summary, our finished and unfinished work are listed below:

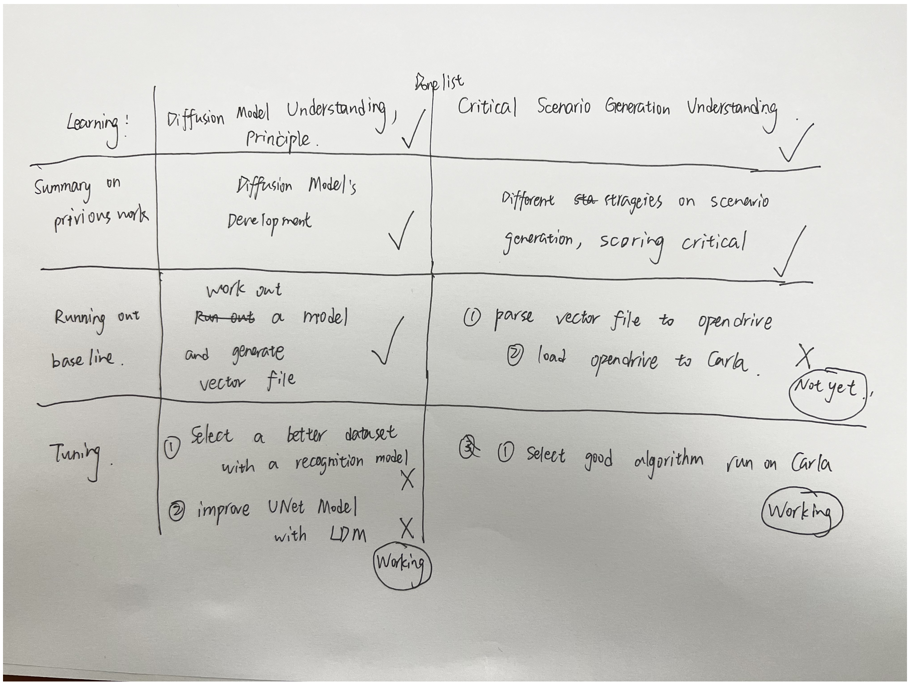

Figure 16: finished and unfinished work

6 Research Plan and Expected Results

Primary Step

In this step, We expected to transform the driving figures into driving scenarios. We will test the scenarios in the Carla simulator platform and we expect these scenarios generated by our model can successfully pass the test.

Level Up

In this step, We will select the dangerous scenarios from the dataset and use these data to train the LDM(Latent Diffusion Model).

7 Potential Challenges and Solutions

Security assessment

The challenge is that we don't how to ensure that the defined scenarios can fully cover all situations that may pose a threat to the auto-drive system.

Solution: Conduct extensive research on accidents and hazardous situations, including historical data, expert opinions, and simulation testing.

Limitations of computing resources and Storage cost

The dataset size is 4TB and it requires a large amount of computing resources, which may limit the scalability and training time of the model.

Solution: Use multi A100 GPU to do Parallelism. And we use NCCL comunication library with UCX_TLX Communication framework to acceralting our inference.

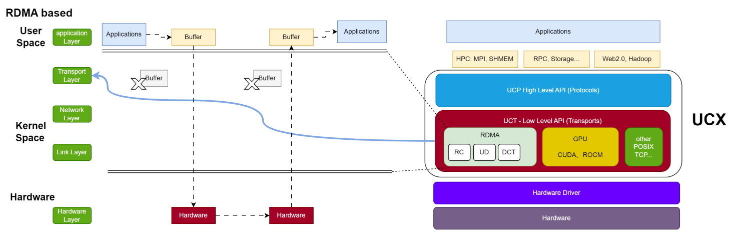

Figure 17: We try to use UCX$_TLStoboostthecommunicationworkonmodelinference$

Image recognition accuracy

We may not have completed denoising or have a low level of completion when generating images

Solution: Optimize the denoising process of the generated model, use more advanced image processing techniques, or use higher quality datasets during model training. Design or train an image recognition model that can evaluate the denoising level and overall quality of an image

8 Project Scheduling

Week14(12.9 $\sim$ 12.15)

We have already generated the driving figures, so the next step of our work is to transform the figures into scenarios. Haibin Lai will finish this job.

Week15(12.16 $\sim$ 12.22)

After the scenarios have been generated Wenkun Shen will test them in the Carla simulation platform

Week16(12.23 $\sim$ 12.29)

After we test the scenarios, we will then select the dangerous driving scenarios from the NHTSA 37 dataset and retrain the Latent Diffusion Model(also called "critical attention awareness Diffusion model") and the same as the above division of labor.

References

[1] Andreas Demetriou, Henrik Allsvåg, Sadegh Rahrovani, and Morteza Haghir Chehreghani. Generation of driving scenario trajectories with generative adversarial networks. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, 2020.

[2] W. Ding, B. Chen, B. Li, K. J. Eun, and D. Zhao. Multimodal safety-critical scenarios generation for decision-making algorithms evaluation. *IEEE Robotics and Automation Letters*, 6(2):1551–1558, 2021.

[3] W. Ding, B. Chen, M. Xu, and D. Zhao. Learning to collide: An adaptive safety-critical scenarios generating method. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2243–2250. IEEE, 2020.

[4] Wenhao Ding, Wenshuo Wang, and Ding Zhao. A multi-vehicle trajectories generator to simulate vehicle-to-vehicle encountering scenarios. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4255–4261, 2019.

[5] Wenhao Ding, Chejian Xu, Mansur Arief, Haohong Lin, Bo Li, and Ding Zhao. A survey on safety-critical driving scenario generation—a methodological perspective. *IEEE Transactions on Intelligent Transportation Systems*, 24(7):6971–6988, 2023.

[6] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.

[7] Y. Guo, V. V. Kalidindi, M. Arief, W. Wang, J. Zhu, H. Peng, and D. Zhao. Modeling multi-vehicle interaction scenarios using gaussian random field. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3974–3980. IEEE, 2019.

[8] Kunkun Hao, Wen Cui, Lu Liu, Yuxi Pan, and Zijiang Yang. Integrating data-driven and knowledge-driven methodologies for safety-critical scenario generation in autonomous vehicle validation. In *2024 IEEE 24th International Conference on Software Quality, Reliability, and Security Companion (QRS-C)*, pages 970–981, 2024.

[9] G. Harshvardhan, M. K. Gourisaria, M. Pandey, and S. S. Rautary. A comprehensive survey and analysis of generative models in machine learning. *Computer Science Review*, 38:100285, 2020.

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

[11] Zhiyu Huang, Zixu Zhang, Ameya Vaidya, Yuxiao Chen, Chen Lv, and Jaime Fernández Fisac. Versatile behavior diffusion for generalized traffic agent simulation, 2024.

[12] Ethan Pronovost, Meghana Reddy Ganesina, Noureldin Hendy, Zeyu Wang, Andres Morales, Kai Wang, and Nicholas Roy. Scenario diffusion: Controllable driving scenario generation with diffusion, 2023.

[13] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer. Survey on scenario-based safety assessment of automated vehicles. *IEEE Access*, 8:87456–87477, 2020.

[14] Shuo Sun, Zekai Gu, Tianchen Sun, Jiawei Sun, Chengran Yuan, Yuhang Han, Dongen Li, and Marcelo H. Ang. Drivescenegen: Generating diverse and realistic driving scenarios from scratch. *IEEE Robotics and Automation Letters*, 9(8):7007–7014, 2024.

[15] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT Press, 2018.

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

[17] Chen Yang, Yangfan He, Aaron Xuxiang Tian, Dong Chen, Jianhui Wang, Tianyu Shi, and Arsalan Heydarian. Wcdt: World-centric diffusion transformer for traffic scene generation, 2024.

[18] Zhenpei Yang, Yuning Chai, Dragomir Anguelov, Yin Zhou, Pei Sun, Dumitru Erhan, Sean Rafferty, and Henrik Kretzschmar. Surfelgan: Synthesizing realistic sensor data for autonomous driving, 2020.

[19] Jiawei Zhang, Chejian Xu, and Bo Li. Chatscene: Knowledge-enabled safety-critical scenario generation for autonomous vehicles, 2024.

[20] Z. Zhong, Y. Tang, Y. Zhou, V. d. O. Neves, Y. Liu, and B. Ray. A survey on scenario-based testing for automated driving systems in high-fidelity simulation. *arXiv preprint arXiv:2112.00964*, 2021.

[21] Ziyuan Zhong, Davis Rempe, Danfei Xu, Yuxiao Chen, Sushant Veer, Tong Che, Baishakhi Ray, and Marco Pavone. Guided conditional diffusion for controllable traffic simulation, 2022.

[22] M. Zhou, J. Luo, J. Villella, Y. Yang, D. Rusu, J. Miao, W. Zhang, M. Alban, I. Fadakar, and Z. Chen. Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving, 2020.