A Data Augmentation Pipeline: Leveraging Controllable Diffusion Models and Automotive Simulation Software

J. van Leuven[1], J. Kober[2], S. Tong[2,3]

[1]Siemens Digital Industries Software, Interleuvenlaan 68, 3001 Leuven, Belgium
[2]Technical University of Delft, Department of Cognitive Robotics, Mekelweg 5, 2628 CD Delft, The Netherlands

Abstract

Training models for autonomous vehicles (AVs) necessitates substantial volumes of high-quality data due to the strong correlation between dataset size and model performance. However, acquiring such datasets is labor-intensive and expensive, requiring significant resources for collection and labeling. To optimize the utility of available data, augmenting the dataset or generating synthetic data presents a cost-effective and efficient solution. Traditional methods that operate within the RGB domain frequently overlook crucial information, such as object frequency, scene composition, and agent trajectories. To address these limitations, a pipeline employing controllable diffusion models and vehicle simulation software is proposed. This approach involves loading collected data into a physics-based simulator, which allows for augmentation beyond the pixel space into the structural space. The augmented simulated data is subsequently transformed back into the photorealistic domain using generative artificial intelligence. This process generates high-fidelity synthetic data, thereby enabling models to train effectively on an expanded and varied dataset, enhancing robustness through the increased variation. The proposed method is evaluated in both image and video domains to assess its effectiveness.

Master of Science at Delft University of Technology
To be defended publicly on 26/07/2024
Faculty: 3mE
Department: Cognitive Robotics
Programme: Robotics

Mentors / Supervisors:
Dr. J. (Jens) Kober, Dr S. (Son) Tong

Graduation Committee:
Dr. J. (Jens) Kober, Dr S. (Son) Tong, Dr. H. (Holger) Caesar

July 2024

A Data Augmentation Pipeline: Leveraging Controllable Diffusion Models and Automotive Simulation Software

J. van Leuven,[1]⋆ J. Kober,[2] S. Tong[2,3]
[1]*Siemens Digital Industries Software, Interleuvenlaan 68, 3001 Leuven, Belgium*
[2]*Technical University of Delft, Department of Cognitive Robotics, Mekelweg 5, 2628 CD Delft, The Netherlands*

1 August 2024

ABSTRACT

Training models for autonomous vehicles (AVs) necessitates substantial volumes of high-quality data due to the strong correlation between dataset size and model performance. However, acquiring such datasets is labor-intensive and expensive, requiring significant resources for collection and labeling. To optimize the utility of available data, augmenting the dataset or generating synthetic data presents a cost-effective and efficient solution. Traditional methods that operate within the RGB domain frequently overlook crucial information, such as object frequency, scene composition, and agent trajectories. To address these limitations, a pipeline employing controllable diffusion models and vehicle simulation software is proposed. This approach involves loading collected data into a physics-based simulator, which allows for augmentation beyond the pixel space into the structural space. The augmented simulated data is subsequently transformed back into the photorealistic domain using generative artificial intelligence. This process generates high-fidelity synthetic data, thereby enabling models to train effectively on an expanded and varied dataset, enhancing robustness through the increased variation. The proposed method is evaluated in both image and video domains to assess its effectiveness.

Keywords: Diffusion models – Domain gap – Simulation in the loop – Autonomous driving – End-to-end – Domain randomization – Domain augmentation – Synthetic data

1 INTRODUCTION

Achieving full self-driving capabilities in autonomous vehicles (AVs) presents significant societal benefits. AVs are anticipated to reduce traffic accidents, thereby potentially saving millions of lives and decreasing injury-related costs [1, 2]. Furthermore, AVs can enhance economic productivity by allowing passengers to engage in other activities during travel time [3] and mitigate congestion costs. For instance, in the Netherlands, congestion costs were estimated between EUR 3.3 and 4.3 billion in 2018, representing approximately 0.5% of the Dutch GDP, primarily due to delays in the transport of goods [4]. Training AI models for AVs necessitates extensive datasets, with a well-established correlation between dataset size and model performance [5]. However, the collection and annotation of these datasets are labor-intensive and costly. To address these challenges and reduce expenses, researchers employ data augmentation techniques or generate synthetic data to maximize the utility of the collected data. Traditional data augmentation methods leverage geometric and photometric techniques such as rotation, scaling, flipping, and adding noise. These methods help the model learn invariant features and enhance its robustness during inference. More advanced augmentation methods employ techniques like image erasing [6], masking parts of the image [7, 8], cutting parts of the image [9], mixing multiple images together [10, 11, 12], and copy-paste strategies that replicate image samples [13]. Recent methods [14, 15, 16, 17, 18, 19] utilize generative models, including diffusion models [20, 21], to produce more diverse augmented training images tailored for various downstream tasks such as image classification [22, 19, 23, 24, 16, 25], object detection [26, 27, 28, 17], and semantic segmentation [29, 30, 31, 28, 32, 33]. Despite their applicability, these methodologies exhibit significant limitations. The synthetic data generated is predominantly derived from latent spaces and directed by prompts, which provide minimal structural guidance. This often results in synthetic data that is irrelevant in real-world contexts. To address these constraints, structural guidance is typically integrated directly from auto-labeling simulators, which confine the model to produce valid samples. However, the scenarios within these simulators do not accurately emulate real-world agent behaviors, as they are inherently simulated. Certain strategies attempt to resolve this issue by either utilizing pre-existing structural information within the dataset or by extracting it directly from RGB data through algorithmic techniques, thereby generating valid and realistic data. Nonetheless, this data remains structurally tethered to the original dataset. The objective is to further enhance synthetic generation techniques to increase the utility of real driving datasets. Our proposed pipeline aims to address these deficiencies by leveraging real driving data to obtain realistic agent behaviors. This pipeline incorporates structural guidance from the simulator to ensure consistent sample generation by the diffusion model and perturbs the dataset within the simulator to expand the dataset across the structural dimension, as illustrated in Figure 3.

1.1 Contributions

The primary contribution of this paper is the introduction of a novel data augmentation technique aimed at increasing the utility of real driving datasets, such as nuScenes [34], nuPlan [35], and

⋆ E-mail: —-@gmail.com

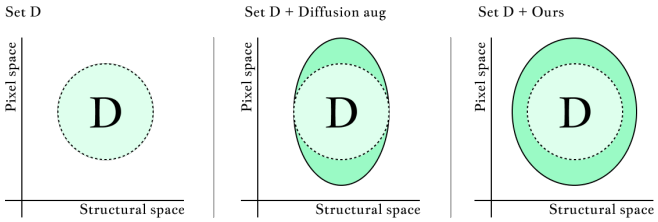


Figure 1. Juxtaposition of diffusion based augmentation methods and ours.

the Waymo Motion Dataset [36], by increasing the variance in the structural space, as depicted in Figure 1. The pipeline is applicable to both image and video domain and operates without the need for large GPU clusters or extensive labelled datasets. Dissecting this algorithm, the primary contributions can be summarized into the following key points:

- A data loading interface and data augmentation algorithm that can load multiple datasets into simulation software (Prescan [37]) and augment them. These datasets include nuScenes [34], nuPlan [35], and the Waymo Motion Dataset [36].
- A diffusion architecture capable of generating photorealistic samples in the image and video domains. This architecture can adhere to segmentation information without requiring a labeled dataset or extensive computational resources.
- Experiments demonstrate that training end-to-end autonomous driving models with the augmented dataset, incorporating both simulation and diffusion models, results in improved average performance compared to using only real data.

This paper continues with Section 2, which elaborates on the necessary background knowledge. Following this, each quadrant of Figure 3 is discussed in Section 3, explaining the rationale behind the proposed method. Experiments are conducted, of which the implementation details and results elaborated on in Section 5, Followed by a discussion in Section 6. Future works are proposed in Section 7, where after there will be concluded in Section 8

2 BACKGROUND

This section delves into the landscape of generative AI, with a particular focus on diffusion models. Unlike standard discriminative or autoregressive models, diffusion models employ a unique learning mechanism, which will be briefly explained. Furthermore, this section covers the prominent datasets and metrics commonly used in the automotive domain to evaluate performance.

2.1 Generative AI

Generative AI models are designed to model the entire data distribution, enabling the generation of new data samples based on seen data. Architectures include auto regressive models, latent variable models, flow-based models, and energy-based models [38].

This research focuses on prescribed latent variable models, which define a probability distribution over data and latent variables. Variational Autoencoders (VAEs) [39] and diffusion models [21, 20] are key examples. The combination of these two models, utilizing the VAE as a compressor and decompressor and the diffusion architecture as the generative power, has resulted in remarkable success in recent years, revolutionizing the field of generative AI. Diffusion

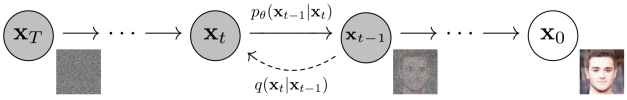


Figure 2. Markov chain of the forward and reverse diffusion process [21].

based models such as DALL-E [40, 41], Imagen [42], Midjourney [43], Stable Diffusion [44], and Sora [45] have demonstrated their robustness and versatility, establishing diffusion models as leading models in the generative AI landscape. Their ability to generate high-quality, diverse images from textual descriptions has garnered significant attention. These models excel in generating images that are not only realistic but also adhere closely to given prompts, making them highly useful for applications in art, design, and content creation. However, their applicability to engineering practices is limited due to their stochastic nature, which can result in inconsistency.

2.2 Diffusion mechanism

Diffusion models are composed of two processes: a forward process and a backward process, as depicted in Figure 2. The forward process follows a Markov chain ranging from 0 to T, where Gaussian noise is added to a sample $\mathbf{x}_0$ in a structured manner through a sampler defined by $q(\mathbf{x}_t \mid \mathbf{x}_{t-1})$. After T steps, this transforms the sample into pure noise. The reversal of the forward process can be learned by a model defined as $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)$, parameterized by θ. Predicting $\mathbf{x}_{t-1} \mid \mathbf{x}_t$ is computationally expensive. Therefore, as proposed by [21], the noise ϵ_t is predicted instead, with a model parameterized by θ, $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$, accepting the time t and sample x_t using the loss function in Equation 1.

$$L = \mathbb{E}_{t\sim[1,T],\mathbf{x}_0,\boldsymbol{\epsilon}_t}\left[\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|^2\right] \tag{1}$$

After being trained on extensive image datasets, diffusion models learn the reverse process for an entire distribution of images. During inference, these models can generate novel images from Gaussian noise. Furthermore, by utilizing cross-attention mechanisms as demonstrated in [46], the generation process can be guided through various modalities.

2.3 Simulators

Simulators generate data by iteratively allowing an agent to take actions and then dynamically updating the environment based on these actions. This feedback loop ensures that the agent's decisions continuously influence the state of the environment, hypothetically allowing for the possibility to generate infinite amounts of data. Various benchmarks are used to assess performance within these simulators. For instance, the CARLA simulator [47] offers several benchmarks, including, noCrash [48], Town05 [49], LAV [50], Roach [51], and Longest6 [52], each focusing on different aspects of autonomous driving such as generalization, safety, and adaptability to diverse environments. The nuPlan benchmark [35] introduces a large-scale driving dataset and a closed-loop simulator, specifically designed for evaluating long-term planning through motion-planning metrics. This thesis utilizes the simulator provided by Siemens, named Simcenter Prescan [37]. The Prescan [37] simulation software enables automatic massive scenario generation with physics-based sensors models and the extraction of various types of sensory data, including images from a monocular camera and segmentation maps.

Domain gap: While these simulators can generate vast datasets, the

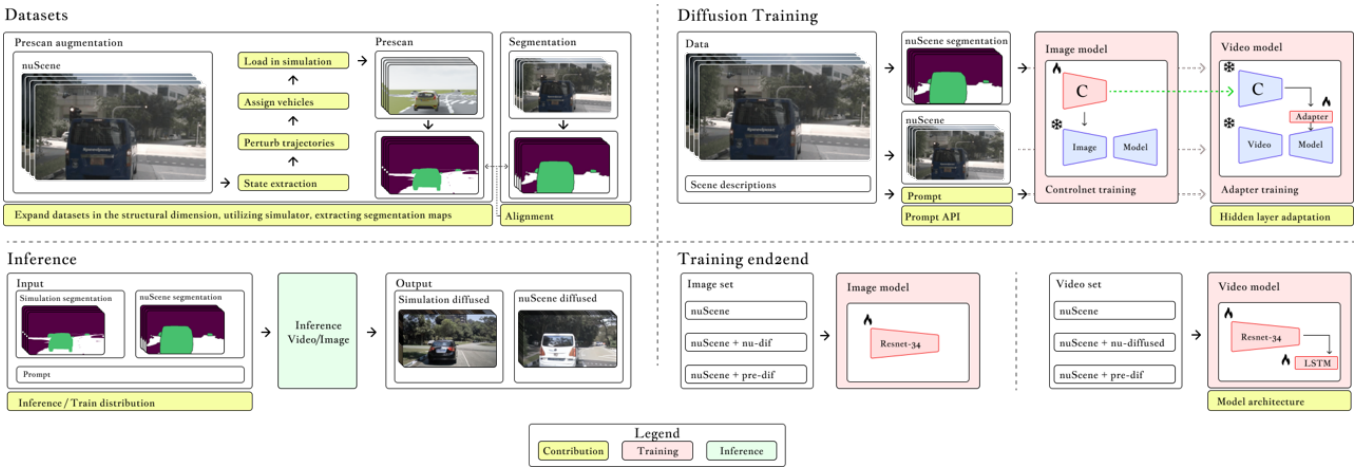


Figure 3. Complete outline of the proposed architecture.

primary challenges of utilizing this data as a surrogate for real data, lies in the discrepancy between simulated agent behavior and real-world conditions, as well as the photo realistic differences between the simulated environment and reality.

2.4 Datasets For Autonmous Driving

Datasets typically encompass sensor readings, destination points, object labels, bounding boxes, and trajectory information. They are utilized to train autonomous vehicle agents by comparing the system's predicted values, derived from the dataset's sensor inputs and objectives, to the actual ground truth labels. The accuracy of these predictions relative to the ground truth serves as a crucial evaluation metric. This research will employ the nuScenes [34] dataset to validate the data augmentation pipeline, owing to its comprehensive documentation of labels and agent trajectories, along with the inclusion of front-view RGB images.

Data distribution: Despite offering realistic driving scenarios, these datasets do not cover every possible situation. This limitation can create gaps in data coverage, excluding scenarios that are possible but not captured in the dataset. Training on such a confined set can lead to unexpected behavior when the model encounters data during inference which lies outside the training distribution.

3 METHOD

The proposed architecture, illustrated in Figure 3, will be explained and justified. The primary objective is to maximize the utility of a real driving dataset. Consequently, the generated data from the pipeline will be assessed both quantitatively and qualitatively. This assessment will utilize the Fréchet Inception Distance (FID) [53] and Dreamsim [54] metrics, and the results will be visualized using t-SNE. Additionally, its efficacy will be tested using an end-to-end image and video model. These models require an image or video as input and output a steering angle, with the video model additionally outputting a longitudinal acceleration action. These results are then compared to a diffusion-only augmentation technique dataset. The following section shall elaborate on the first quadrant, datasets.

3.1 Datasets

To enhance the utility of the augmented dataset, the expansion in the structural space must add novel information; therefore, the scene composition must be altered in a meaningful way. The structural perturbations should be minor enough to ensure that the original ground truth labels remain accurate, and the new dataset must remain valid from a physics-based perspective. Therefore, a physics-based simulator is utilized to ensure that the new scene compositions, although altered, remain physically plausible. Both the image and video-based downstream models are designed to predict the steering angle. By optimizing for this control action, meaningful properties of the scene can be identified. When perturbed, these properties expand the dataset, enhancing the downstream model's robustness to variations. The hypothesis is that the steering angle of a car, excluding cut-in or collision events, remains consistent regardless of the types of vehicles surrounding the ego vehicle. Similarly, it is hypothesized that the steering angle remains consistent irrespective of the longitudinal displacement of other agents with respect to the ego vehicle, again excluding cut-in or collision events. Therefore, these two properties are selected to be perturbed. To achieve this, the pose and dimension information of static obstacles such as traffic cones, road blockages, speed bumps, lanes, and crossroads, including are extracted. Subsequently, the trajectories and classifications of all dynamic entities within the dataset are extracted and subjected to minor perturbations. The resultant modified dataset is then loaded into the simulation environment. The specific implementation details of the simulator are provided in Section 3.1. After loading the data, the rendered images must be transformed from simulation images to the photorealistic domain using diffusion models. These models require caption and segmentation information. Segmentation maps can be directly extracted from the simulator, while captions are obtained using the scene descriptions provided in the datasets. The motivation for using segmentation information and the structure of the prompts will be elaborated on in Subsection 3.2.

To test the efficacy of the structural modifications, a benchmark will be used, inspired by diffusion-based augmentation algorithms [14]. Since the nuScenes [34] dataset does not inherently include segmentation maps, a high-performing segmentation algorithm is employed [55], to aquire the maps. The same diffusion process will be applied, expanding the dataset in the pixel space, but without any structural modifications. This approach allows for assessing whether the struc-

tural modifications provide any benefits. This data acquisition process is visually depicted in Figure 3 (the top left corner).

3.2 Diffusion Architecture

To leverage diffusion models for data generation, the type of guidance needs to be selected for the sample generation, which can include textual information, depth data, pose information, segmentation maps, HD maps, bounding boxes, or edge information. Many types of structural guidance can be extracted from the physics-based simulator Prescan [37]. When choosing the appropriate structural guidance technique, two factors must be considered: the maximum encoded information per pixel and the discrepancy of the structural guidance data between the training and inference sets.

To choose the appropriate guidance technique, the perspective must first be specified. The downstream models will operate on monocular front-view camera images, so the guidance techniques must be relevant from this view. From this perspective, depth and segmentation images contain the most and similar amounts of information per pixel, whether it be depth per pixel or label per pixel.

To select the appropriate guidance method between the two options, the second criterion is employed, which necessitates understanding the gap between the training data and the inference data. The diffusion models must utilize structural guidance to generate photorealistic images, ensuring consistent and plausible samples. The training data for the diffusion models will consist of the original RGB data from the nuScenes [34] dataset, as this is the data that needs to be modeled. The structural guidance for the RGB images must closely align with the RGB samples to avoid misalignments that could lead to poor generative capabilities. The structural guidance provided by the simulator does not align closely enough with the RGB data for a successful training process. Therefore, the guidance must be directly extracted from the nuScenes [34] RGB images for training. Since the diffusion model is trained on structural guidance from the RGB images, the guidance passed during inference from the simulator must be similar. If this is not the case, the model will not understand the data. Hence, a structural guidance technique must be chosen where the training data and inference data do not significantly differ.

The discrepancy for depth data in real and simulated scenarios is considerable because static obstacles, such as buildings, are not imported into the simulator. This results in the diffusion models having access to more information during training than during inference, leading to poor results. This issue persists even when using segmentation images; however, in the pre-processing of the segmentation data for training, labels that are known not to be in the simulation dataset, such as buildings, can be dropped and mapped to the background. This is not trivial in the case of depth data because removing points from static buildings without known labels is challenging. Therefore, segmentation maps are chosen to facilitate structural guidance.

Although more structural conditions, such as bounding boxes, could be added to include more structural data, segmentation maps will suffice for testing the pipeline. In addition to structural guidance, other diffusion augmentation techniques [14, 15] utilize prompts as additional guidance. Although not foundational, the pipeline includes prompts to showcase the versatility of diffusion models in altering the scene with different parameters, such as the geographical location, building types, daytime, and weather conditions. The method of injecting these conditionals differs between image and video models, and thus these models will be discussed separately, starting with the image-based diffusion model.

3.2.1 Diffusion Model: Image Generation

Training diffusion models is computationally intensive; for instance, latent diffusion models like those used in Stable Diffusion [44] comprise approximately 400 million parameters [56]. Training such a model on ImageNet, a dataset with 14 million hand-annotated images [57], using a V100 GPU, takes approximately 271 days [56]. Some pre-trained model's parameters are publicly available, providing access to trained models which have internalized complex relationships between images and captions, allowing for high-fidelty generation of samples though prompt. However, as detailed in Section 3.2, it is also necessary to incorporate structural control into the model to ensure generated samples are consistent with the calculated structure.

Training the entire model from scratch to incorporate additional conditions auxiliary to the prompt would disregard the existing pre-trained weights. Fine-tuning the model using the pre-trained weights for new conditions can lead to catastrophic forgetting, where the model overwrites valuable previously learned information. To mitigate this, the ControlNet architecture [58] is utilized. This architecture is designed to allow the model to adapt to new conditions while preserving the knowledge it has already acquired, thus preventing the loss of valuable information from the pre-trained weights.

This architecture duplicates the encoder portion of the diffusion model and provides this duplicate with the same input as the base model alongside the additional conditioning. The feature maps that are generated during the forward pass are combined with the original feature maps using zero-convolutional layers [58]. During training, the original model weights remain frozen, and only the copied weights are trainable. This approach preserves the original knowledge while allowing the model to learn new conditioning.

At the onset of training, the duplicate of the encoder is initialized with the weights of an open-source pre-trained ControlNet [59] conditioned on segmentation maps to speed up the training process. For fine-tuning this model specifically for driving images, 700 scenes approximately 168000 images from the nuScenes dataset [34] are utilized, focusing exclusively on the front camera images. This method ensures that the model retains its original capabilities while adapting to new, specific tasks relevant to autonomous driving.

3.2.2 Diffusion Model: Video Generation

Extending diffusion models to generate videos requires the model to produce frames that are temporally consistent with one another. Generating multiple frames simultaneously is computationally expensive, and training these models demands more resources compared to training image-based models. Methods proposed in [60, 61, 62, 63, 46, 64, 65] have demonstrated the ability to generate video. However, none of these methods accept segmentation maps as conditional inputs, or have publicly available training scripts, and those that do require extensive data to train effectively.

To address these challenges, the already trained image ControlNet[58] is extended the to the video domain, depicted in Figure 4. It must be noted that the image ControlNet lacks a temporal dimension and to mitigate this, an approach inspired by CTRL-Adapter [66] was used. During training, similar to the image ControlNet [58], the base model's weights are frozen, and the ControlNet's [58] feature maps are added to the feature maps of the new video base model. However, due to the additional temporal dimension in the new base model architecture, direct zero-convolution yields poor results as the feature maps of the image ControlNet [58] are not temporally aligned. Therefore, an additional adapter is introduced between the

(a) Original [34] (b) Simulator

Figure 4. Loading real driving data into the Prescan [37] simulator

feature maps. This adapter consists of the following layers: spatial convolution, temporal convolution, spatial attention, and temporal attention [66]. These layers are trained during the process and align the image ControlNet feature maps over the temporal dimension, enabling the usage of an image ControlNet on a video base model without requiring extensive datasets.

In addition to accepting a prompt and a condition, the initial starting frame of the video is provided to the model, allowing the model to understand which color palette to use. Combined with the segmentation images from the simulator and the initial reference frame, the model can create videos of 16 frames. This method ensures the temporal consistency of generated frames while incorporating structural control based on the provided segmentation maps, as depicted in the diffusion training section of Figure 3.

4 DOWNSTREAM MODELS

To validate the performance of the proposed architecture and assess the quality of the generated samples, the downstream models are trained on three datasets, as depicted in the training end-to-end section of Figure 3.

- **Dataset 1: nuScenes** This experiment trains the model only on the 700 training scenes of nuScenes [34] approximately, 168000 images.
- **Dataset 2: nuScenes + nu-dif** This experiment trains on the 700 nuScenes [34] scenes plus 120 generated scenes originating from the segmentation maps obtained directly from the RGB data using the ODISE segmentation algorithm [55], totalling 196800 images.
- **Dataset 3: nuScenes + pre-dif** This experiment trains on the 700 nuScenes [34] scenes plus 120 generated scenes originating from the Prescan [37] augmented dataset and segmentation maps obtained through the simulator, totalling 196800 images.

This setup allows for a comprehensive evaluation of the model's ability to generate high-quality samples and its impact on the performance of downstream tasks in both image and video contexts. In both cases, 17.1% of the original dataset was augmented, adding 120 scenes. The augmented scenes were randomly selected from the nuScenes [34] training dataset.

The image end-to-end model processes an image through a ResNet-34 network, which outputs a steering angle mapped between [0,1], similar to the controller described in [67]. The video model employs a similar approach, where six frames are processed through a ResNet-34 network that functions as a feature extractor. The extracted features are then passed to a Long Short-Term Memory (LSTM) network, which interprets the relationships between the features of the frames. This LSTM network outputs a steering angle and because it has access to multiple time frames a longitudinal acceleration as well, mapped between [0,1]. By comparing the performance of the models trained on different datasets, insights can be gained into whether the augmentation of the dataset in both pixel and structural space leads to a significant improvement in the performance of the downstream models.

5 EXPERIMENTS & RESULTS

Similar to Section 3, each quadrant depicted in Figure 3 will be explained independently, highlighting the implementation details and discussing both quantitative and qualitative results. The section will commence with an examination of the datasets used in the experiments.

5.1 Processing Datasets

The data loading API consists of two main parts: initially preprocessing the data, which involves extracting the desired information from the real driving datasets so that it can be loaded into the simulator, and the augmentation process. These two steps are detailed in the following sections.

5.1.1 Data Loading

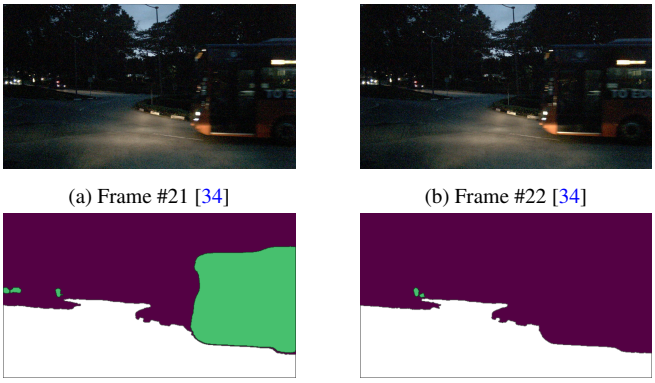
To maximize the utility of the data loading API for Prescan [37], it is crucial to ensure that the API is not tailored to a single dataset. This can be achieved by employing a generic datatype capable of loading multiple datasets and building the API based on this more versatile data structure. Researchers from [68] have developed such a generic datatype, which allows for the loading of various datasets, including Waymo [36], Woven [69], nuPlan [35], and nuScenes [34]. This generic datatype is leveraged for the API, ensuring broad compatibility and flexibility.

Prescan [37] requires data to be parsed in a specific format. The developed API extracts the necessary objects, such as actors, road surfaces, lanes, crossroads, speed bumps, and their dimensions, positions, and velocity vectors. This data is then formatted to be compatible with the Prescan [37] simulator. In Figure 4, it can be observed that the API successfully loads all the dynamic actors and static obstacles, such as the road, into the simulator.

Diffusion models often have difficulty interpreting numerical data when injected via a prompt. Therefore, the prompt focuses solely on categorical aspects such as weather type, time of day, and geographical location. To extract the caption that will accompany the samples in the generation process, a separate Python module has been developed to parse the description of the nuScenes [34] data, forming sentences such as:

(a) Normal scene (b) Augmented scene

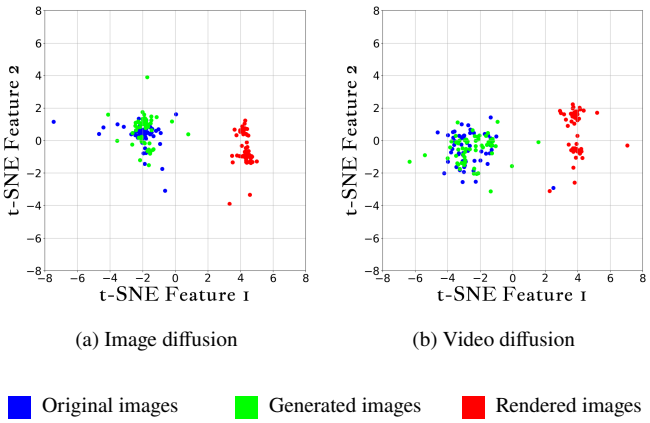
Figure 5. Comparison of a scene and an augmented scene, illustrating variations in car types and slight forward displacement of agents at the same timestamp.



(a) Frame #21 [34] (b) Frame #22 [34]

(c) Segmentation Figure 6a (d) Segmentation Figure 6b

Figure 6. The detection failure of the ODISE [55] segmentation algorithm highlighted in consecutive frames.



(a) Image diffusion (b) Video diffusion

Figure 7. T-sne projection of 100 images, showing the gap between simulated data and real data, and how diffusion models are able to map simulation information to the real domain.

*A **clear** driving scene, in **Boston**, during the **day**.*

5.1.2 Augmentation

Once the data is loaded into Prescan [37], as depicted in Figure 4, various parameters such as agent trajectories and car types become accessible. The types of cars are randomly assigned, and in addition to varying the car types, the trajectories are perturbed. To avoid diverging significantly from the original dataset, the velocities of all actors are increased by a random percentage between 0 and 40%. These increased velocities are uniformly applied to all actors for 1.6 seconds. After this period, the actors are teleported back to their original positions, and a new random velocity is applied for the next 1.6 seconds. Although this increase seems high, the simulator's rendering methods incorporate a smoothing function over the positional trajectories. As a result, the differences in position are not pronounced, typically varying by less than a few meters upon inspection. This method ensures that the ground truth control labels remain valid while expanding the dataset. This randomization process is depicted in Figure 5, where it can be observed that the agents have different car types and at the same timestamp, the cars in the augmented scene are slightly ahead.

5.1.3 Automatic Semantic Segmentation

To segment the frames of the original nuScenes [34] scenarios, it is first necessary to align the field of view (FOV) of the Prescan [37] images with that of the nuScenes [34] images. This alignment ensures that the discrepancy between the conditions during training and inference time is minimized. This alignment is achieved through a pre-processing method that adjusts the FOV of the nuScenes [34] images to match the camera angle of the Prescan [37] camera. Once this pre-processing is complete, the samples are fed into the segmentation algorithm. The algorithm is slightly modified to perform semantic segmentation instead of panoptic segmentation, ensuring better alignment between training and inference conditions.

The model demonstrates a high degree of robustness; however, errors occur during inference, as shown in Figure 6. One notable issue is the failure to detect the same vehicle in consecutive frames due to poor lighting conditions. In contrast, the segmentation maps generated by the simulator are not affected by these errors because the simulator can automatically label the scene. Nonetheless, as discussed in Section 3.2, the segmentation maps originating from the Prescan [37] simulator do not perfectly align with RGB data from nuScenes [34], leading to worse conditioning alignment than the detection failures observed with ODISE [55]. Therefore, the segmentation maps produced by ODISE, combined with the RGB scenes from which they are derived, are used for training the diffusion model.

Similarity Metric	Simulated	Generated image	Generated video
FID [53]	316.02	124.38	**105.66**
Dreamsim [54]	20.57	14.83	**10.29**

Table 1. A quantitative assessment of the generated images by the video and image model, comparing them with rendered images using FID [53] and the DreamSim [54] metrics.

5.2 Diffusion Training & Inference

An image and a video diffusion model are trained, sharing the same ControlNet. To obtain the weights for the ControlNet [58], the image model is initially trained with the segmentation images from ODISE [55] alongside the RGB images from nuScenes [34] and a textual prompt provided through the prompt generator. During this training phase, the weights of the base model [44] are kept frozen. The training parameters include a learning rate of 0.00001, an input size of 640x360 pixels, no weight decay, and a batch size of 64. The 8-bit Adam optimizer is employed for 10,000 steps on a single NVIDIA A6000 GPU with 48 GB VRAM.

For the video model, during training, the image ControlNet [58] and the base model [70] are frozen, and only the weights of the adapter are trained. The learning rate remains at 0.00001, but the input size is adjusted to 256x160 pixels, and the batch size is reduced to 16. The 8-bit Adam optimizer is used for 70,000 steps, also on a single NVIDIA A6000 GPU with 48 GB VRAM. During inference, the segmentation maps of 120 scenes from ODISE [55] are processed through the models, producing the set **nu-diff**. The Prescan [37] segmentation maps are collected at a rate of 20 Hz. Due to the 12 Hz collection rate of the nuScenes [34] dataset, these segmentations of the same 120 scenes are first sampled before being passed to the image and video models for inference, creating the set **pre-diff**.

The efficacy of the generated samples will be tested through the performance of the downstream models. In addition to this, a qualitative

and a quantitative assessment will be performed. A popular metric to assess the photorealism of a generated sample with respect to a real sample is the Fréchet Inception Distance (FID) score [53].

The FID score measures how similar the generated images are to real images by comparing the statistical properties of features extracted from both sets of images using a pre-trained neural network. A lower FID score indicates higher image quality and greater similarity to real images. FID scores are calculated for 400 Prescan images, 400 generated diffusion images, and 400 generated video diffusion images, yielding the scores shown in Table 1. The simulated images obtained directly from the simulator are the least similar to the real images. The image diffusion models can map the segmentation maps of the simulator to images that are closer to the real images than the simulated images, indicating that the generated images add valuable information, making the samples look more realistic. The video models generate samples closest to the original dataset, which can be attributed to the fact that the video model accepts an initial reference frame, thereby knowing beforehand which color palette to use, creating images more closely resembling the original image. These numbers have no physical meaning, therefore an addition metric is used to validate the results.

Dreamsim [54] is a similar state-of-the-art metric utilizing a pre-trained model to assess the quality of the samples. However, it does not merely measure the photorealism of the generated images but also the structural similarities. It can be seen in Table 1 that the simulated images have the lowest structural similarity, which can be attributed to two reasons. The first cause is the photorealism gap between the simulated images and the real images. The second reason is that although the dynamic actors are imported, the static obstacles are missing, creating a structural gap. The images produced by the image and video diffusion models exhibit a closer resemblance to the original images. Notably, the video models outperform the image models. This superior performance can be explained by the same reasoning applied to the FID [53] score, where the temporal consistency and continuity in video data provide a more coherent and realistic representation compared to individual images.

To visually analyze the gap between the simulated images and the generated images, the samples are projected to a low-dimensional space using PCA, after which the features are mapped to a 2D plane using t-SNE, visualized in Figure 7. Again, a similar trend can be observed where the generated samples are closer to the real images.

To showcase the versatility and high performance of the diffusion models, results of the image diffusion model are depicted in Figure 8. It can be seen that the gap between the training segmentation maps and the inference segmentation maps has been proficiently minimized, allowing the inference segmentation maps to generate high-quality samples. Figure 8a is generated using a segmentation map from ODISE and Figure 8b from augmented maps from Prescan [37]. Visually the sample generated from the simulated segmentation map does not show any discrepancy with the guidance it was given, indicating the gap between the training and inference data was sufficiently small. Utilizing prompt guidance, it is observed that the diffusion model can change weather conditions, demonstrating the potential to generate vast amounts of structurally correct data that can enhance the robustness of downstream models. Results of the video diffusion model are depicted in Figures 1 2,3.

(a) Real image [34]

(b) Generated from ODISE [55]

(c) Generated from Prescan [37]

(d) "A rainy driving scene"

Figure 8. Figure 8b illustrates a generated sample using the segmentation map from Figure 8a with the default prompt "A driving scene." Figure 8c presents a generated sample utilizing the segmentation map from a perturbed Prescan [37] scenario, again with the default prompt. Lastly, Figure 8d depicts a sample generated with the same segmentation map as in Figure 8c, but with the prompt "a rainy driving scene."

5.3 Validation Using Downstream Models

To validate the efficacy of the additional augmented data, the image and video models are trained on the three datasets specified in Section 4. The augmented data for the image end-to-end model is generated using the image diffusion model, whereas the augmented data for the video end-to-end model is produced by the video diffusion model. For the image model, inspired by the methodology presented in [67], a ResNet-34 model is trained to accept an image as input and output a single steering angle, normalized between 0 and 1. Training is performed with a learning rate of 0.001, without weight decay, and a batch size of 512. The Adam optimizer is used for 50 epochs on a single NVIDIA A6000 GPU with 48 GB VRAM, utilizing L1 loss. Extending this architecture to the video domain, a different ResNet-34 batch processes 6 frames and extracts 1024 features for each image. These are then fed into an LSTM network, which can extract the underlying relationships between the features, outputting a steering angle and a longitudinal acceleration. Training is performed with a learning rate of 0.001, with weight decay after 30 steps with step size 1, and a batch size of 64. The Adam optimizer is used for 50 epochs on a single NVIDIA A6000 GPU with 48 GB VRAM. The performance of the models on the nuScenes [34] test set are depicted in Table 2 and visualized in Figure 9

Analyzing the image model first, it can be seen in Table 2 that augmenting the dataset with data that has been perturbed solely in the pixel space, namely the **nuScenes + nu-diff** dataset, increases the performance of the model. The addition of the augmented diffused data, which inherently has some noise, made the model more robust in scenarios where the RGB images are not clear. Training the model on the **nuScenes+pre-diff** set, where the dataset also has perturbations along the structural dimension, further improves performance, showcasing that adding augmentation over the structural dimension indeed improves robustness.

The video models outperform their image counterparts. This is due to the fact that passing multiple frames to the models makes them inherently more robust. The trend of performance increase in the augmented dataset follows that of the image models for the steering angle, although it is less pronounced. This is because the generated samples are constrained over the temporal dimension, resulting in less pixel-wise randomization and, therefore, less overall randomization. The improvement in the acceleration angle is observed but is not significant. This can be attributed to the methodology of per-

Model architecture	Dataset	Steering error	Acceleration error
ResNet-34	nu	0.07248	N/A
ResNet-34	nu + nu-dif	0.03884	N/A
ResNet-34	**nu + pre-dif**	**0.02171**	**N/A**
ResNet-34 + LSTM	nu	0.02794	0.04813
ResNet-34 + LSTM	nu + nu-dif	0.02485	0.04765
ResNet-34 + LSTM	**nu + pre-dif**	**0.02069**	**0.04691**

Table 2. Performance of downstream models trained on the different datasets

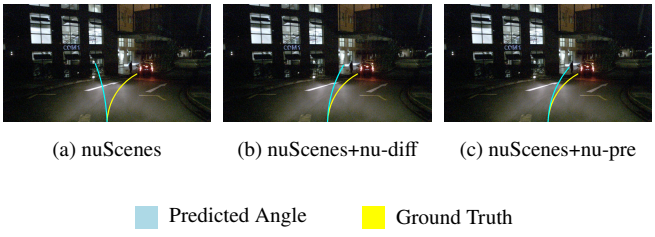
turbing the data in the structural dimension not being optimized for acceleration purposes.

These results showcase that adding structural perturbations using a physics based simulator in the loop indeed enhance the dataset more than merely using diffusion based augmentation techniques.

6 DISCUSSION

Data acquisition is inherently expensive, making the augmentation of datasets a critical tool for enhancing the utility of collected data. Recent advancements in diffusion models offer a promising avenue for dataset augmentation. However, the generative nature of diffusion models often leads to the production of artifacts. To mitigate this issue, the incorporation of structural control within diffusion models has been shown to be effective in generating more realistic data. By leveraging the complementary strengths of physics-based simulators and diffusion models, the structural integrity tasks can be managed by the simulators, while diffusion models enhance photorealism. This synergistic approach results in a robust data augmentation technique that has been demonstrated to outperform methods relying solely on diffusion-based augmentation. However ensuring the validity of ground truth labels after dataset perturbation requires more sophisticated algorithms, beyond simple perturbations such as altering velocity or changing car types. Simple perturbations can generate scenes that misalign with the original ground truth labels, leading to inconsistencies and inaccuracies in the training data. Perturbations along the structural dimension have not significantly improved the performance of predicted acceleration, indicating the need for more refined augmentation techniques for this aspect.

Currently, augmented scenes are sampled at random. While this approach is somewhat effective, it can lead to inefficient use of computational resources when generated samples do not contribute to enhancing the model's robustness. A more efficient strategy would involve analyzing the samples that cause the greatest errors and then using the pipeline to generate new scenarios specifically targeting these problematic samples.



(a) nuScenes (b) nuScenes+nu-diff (c) nuScenes+nu-pre

Figure 9. Predicted steering angle of image models trained on the different datasets visualized [34].

Moreover, to facilitate video generation, an attempt was made to create a video ControlNet architecture. This involved extending video-based models, such as those described in [46, 70], with an auxiliary video ControlNet that incorporates a temporal dimension to accept temporally aligned conditions. However, this approach proved infeasible due to the extensive training data required. After more than 771 computation hours on a single V100, the model learned to disregard the conditions entirely.

The CTRL-adapter architecture has functioned as an effective surrogate for video generation methods. However, given the potential of video generation algorithms and the advantages of utilizing more parameters, it is worth considering the implementation of not merely an image ControlNet but also a video ControlNet. The incorporation of additional parameters could generate even higher fidelity samples.

Additionally, as simulators continue to evolve towards more realistic renderings, ignoring the RGB data generated from the simulator may result in the loss of valuable information that could benefit the diffusion models. Integrating this RGB data could enhance the realism and utility of the augmented datasets.

7 FUTURE WORK

Currently, the segmentation maps provide the structural guidance, but sometimes the model has trouble recognizing if the car is pointing towards the camera or facing away. To alleviate this problem, additional conditions could be added to the diffusion models, such as bounding box control. By incorporating bounding box information, the model can better understand the orientation and position of objects within the scene, thus improving the accuracy of the generated samples.

Moreover, the Prescan simulator should enhance its realism by importing static backgrounds. This would help to bridge the gap between simulated and real data even further, making the augmented datasets more valuable. By incorporating realistic backgrounds, the simulation environment will better mimic real-world conditions, thereby improving the overall quality and applicability of the generated samples.

Although the data augmentation method significantly improves performance, it is not without limitations. Despite generating high-fidelity synthetic data, the inherent differences between synthetic and real-world data and data coverage can still affect model performance. This highlights the need for continuous refinement and validation.

8 CONCLUSION

This paper introduces a novel data augmentation pipeline that utilizes controllable diffusion models and autonomous simulation software to enhance the utility of real driving datasets. By incorporating structural guidance and simulation-based perturbations, the proposed method generates data that is both realistic and structurally valid.

The integration of diffusion models with the Prescan [37] simulator enables augmentation in both pixel and structural spaces. The efficacy of the augmented data is evidenced by the improved performance of downstream models in both image and video domains. Quantitative assessments, as demonstrated by metrics such as the Fréchet Inception Distance (FID) [53] and Dreamsim [54] scores, show similar improvements. The generated samples closely resemble real images, effectively bridging the simulation-to-reality gap.

The results indicate that adding 17.1% of augmented data, amounting to 120 scenes, led to a significant increase in performance for both

image and video end-to-end models. This underscores the robustness and versatility of the proposed pipeline, demonstrating its capability to produce high-quality, photorealistic samples that effectively enhance the training data for autonomous driving models.

ACKNOWLEDGEMENTS

I would like to express my gratitude to Dr. Jens Kober for his supervision throughout this thesis. Additionally, I extend my thanks to Dr. Son Tong for his invaluable guidance and for providing this wonderful learning opportunity. I also appreciate the assistance of Samuele Ruffino and Joost Zaalberg, whose support was crucial in addressing implementation issues. Furthermore, I would like to thank Willem Momma and Paul IJzermans for their help in improving the readability of this paper.

References

[1] World Health Organization. *Global Status Report on Road Safety 2018*. Accessed: 2024-07-08. 2018. URL: https://apps.who.int/iris/bitstream/handle/10665/277370/WHO-NMH-NVI-18.20-eng.pdf?ua=1.

[2] National Highway Traffic Safety Administration. *The economic and societal impact of motor vehicle crashes, 2010 (Revised)*. Accessed: 2024-07-08. 2015. URL: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812013.

[3] National Highway Traffic Safety Administration. *Automated vehicles for safety*. Accessed: 2024-07-08. 2020. URL: https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety.

[4] Knowledge Institute for Mobility Policy. *Mobiliteitsbeeld 2019*. Accessed: 2024-07-08. 2019. URL: https://www.kimnet.nl/mobiliteitsbeeld/publicaties/rapporten/2019/11/12/mobiliteitsbeeld-2019-vooral-het-gebruik-van-de-trein-neemt-toe.

[5] John Houston et al. "One thousand and one hours: Self-driving motion prediction dataset". In: *Conference on Robot Learning*. PMLR. 2021, pp. 409–418.

[6] Zhun Zhong et al. "Random erasing data augmentation". In: *Proceedings of the AAAI conference on artificial intelligence* 34 (2020), pp. 13001–13008.

[7] Pengguang Chen et al. "Gridmask data augmentation". In: *arXiv preprint arXiv:2001.04086* (2020).

[8] Pu Li, Xiangyang Li, and Xiang Long. "Fencemask: a data augmentation approach for pre-extracted image features". In: *arXiv preprint arXiv:2006.07877* (2020).

[9] Terrance DeVries and Graham W Taylor. "Improved regularization of convolutional neural networks with cutout". In: *arXiv preprint arXiv:1708.04552* (2017).

[10] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection". In: *arXiv preprint arXiv:2004.10934*. 2020.

[11] Hongyi Zhang et al. "mixup: Beyond empirical risk minimization". In: *arXiv preprint arXiv:1710.09412* (2017).

[12] Sangdoo Yun et al. "Cutmix: Regularization strategy to train strong classifiers with localizable features". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 6023–6032.

[13] Golnaz Ghiasi et al. "Simple copy-paste is a strong data augmentation method for instance segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 2918–2928.

[14] Yuru Jia et al. "DGInStyle: Domain-Generalizable Semantic Segmentation with Image Diffusion Models and Stylized Semantic Control". In: *arXiv preprint arXiv:2312.03048* (2023).

[15] Harsh Goel and Sai Shankar Narasimhan. "Improving End-To-End Autonomous Driving with Synthetic Data from Latent Diffusion Models". In: *First Vision and Language for Autonomous Driving and Robotics Workshop*.

[16] Mert Bulent Sariyildiz et al. *Fake it till you make it: Learning transferable representations from synthetic ImageNet clones*. 2023. arXiv: 2212.08420 [cs.CV].

[17] Andrew Farley, Mohsen Zand, and Michael Greenspan. *Diffusion Dataset Generation: Towards Closing the Sim2Real Gap for Pedestrian Detection*. 2023. arXiv: 2305.09401 [cs.CV].

[18] Haoyang Fang et al. "Data augmentation for object detection via controllable diffusion models". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024, pp. 1257–1266.

[19] Lisa Dunlap et al. "Diversify your vision datasets with automatic diffusion-based augmentation". In: *Advances in Neural Information Processing Systems* 36 (2024).

[20] Jiaming Song, Chenlin Meng, and Stefano Ermon. "Denoising Diffusion Implicit Models". In: (2022). arXiv: 2010.02502 [cs.LG].

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models". In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[22] Shekoofeh Azizi et al. "Synthetic data from diffusion models improves ImageNet classification". In: *arXiv preprint arXiv:2304.08466* (2023).

[23] Ruifei He et al. "Is synthetic data from generative models ready for image recognition?" In: *arXiv preprint arXiv:2210.07574* (2023).

[24] Zheng Li et al. "Is synthetic data from diffusion models ready for knowledge distillation?" In: *arXiv preprint arXiv:2305.12954* (2023).

[25] Brandon Trabucco et al. "Effective data augmentation with diffusion models". In: *arXiv preprint arXiv:2302.07944* (2023).

[26] Kai Chen et al. "GeoDiffusion: Text-prompted geometric control for object detection data generation". In: *arXiv preprint arXiv:2306.04607* (2023).

[27] Zhenyu Wu et al. "Synthetic data supervised salient object detection". In: *In ACM International Conference on Multimedia* (2022), pp. 5557–5565.

[28] Weijia Wu et al. "Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models". In: *arXiv preprint arXiv:2303.11681* (2023).

[29] Rui Gong et al. "Prompting diffusion representations for cross-domain semantic segmentation". In: *arXiv preprint arXiv:2307.02138* (2023).

[30] Neehar Kondapaneni et al. "Text-image alignment for diffusion-based perception". In: *arXiv preprint arXiv:2310.00031* (2023).

[31] Duo Peng et al. "Diffusion-based image translation with label guidance for domain adaptive semantic segmentation". In: *arXiv preprint arXiv:2303.11681* (2023).

[32] Lihe Yang et al. "Freemask: Synthetic images with dense annotations make stronger segmentation models". In: *arXiv preprint arXiv:2310.15160* (2023).

[33] Jiwen Yu et al. "FreeDoM: Training-free energy-guided conditional diffusion model". In: *arXiv preprint arXiv:2303.09833* (2023).

[34] Holger Caesar et al. "nuScenes: A Multimodal Dataset for Autonomous Driving". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[35] *nuPlan Benchmark*. https://www.nuscenes.org/nuplan. Accessed: 2023-12-05.

[36] Google. *waymo wesite*. Accessed: 2023-10-15. URL: https://waymo.com/waymo-driver/.

[37] Siemens. *Simcenter Prescan Software simulation platform*. https://plm.sw.siemens.com/en-US/simcenter/autonomous-vehicle-solutions/prescan/. Accessed: 2023-11-29. 2023.

[38] J.M. Tomczak. *Deep Generative Modeling*. Springer International Publishing, 2022. ISBN: 9783030931575. URL: https://books.google.nl/books?id=GxO9zgEACAAJ.

[39] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[40] OpenAI. *DALL·E-2*. 2023. URL: https://openai.com/dall-e-2.

[41] OpenAI. *DALL·E-3*. 2023. URL: https://openai.com/dall-e-3.

[42] Google. *Imagen*. 2023. URL: https://imagen.research.google/.

[43] Midjourney. *Midjourney*. 2023. URL: https://www.midjourney.com/home/?callbackUrl=%2Fapp%2F.

[44] *Stable Diffusion v1.5*. https://huggingface.co/runwayml/stable-diffusion-v1-5. Accessed: 2023-12-05.

[45] OpenAI. *Video Generation Models as World Simulators*. Accessed: 2024-05-14. 2024. URL: https://openai.com/index/video-generation-models-as-world-simulators/.

[46] Andreas Blattmann et al. *Stable Video Diff'usion: Scaling Latent Video Diffusion Models to Large Datasets*. 2023. arXiv: 2311.15127 [cs.CV]. URL: https://arxiv.org/abs/2311.15127.

[47] CARLA Simulator. *CARLA: An Open Urban Driving Simulator*. https://carla.org/. Accessed: 2023-11-08. n.d.

[48] Felipe Codevilla et al. *On Offline Evaluation of Vision-based Driving Models*. 2018. arXiv: 1809.04843 [cs.CV].

[49] Aditya Prakash, Kashyap Chitta, and Andreas Geiger. *Multi-Modal Fusion Transformer for End-to-End Autonomous Driving*. 2021. arXiv: 2104.09224 [cs.CV].

[50] Dian Chen and Philipp Krähenbühl. *Learning from All Vehicles*. 2022. arXiv: 2203.11934 [cs.RO].

[51] Zhejun Zhang et al. *End-to-End Urban Driving by Imitating a Reinforcement Learning Coach*. 2021. arXiv: 2108.08265 [cs.CV].

[52] Kashyap Chitta et al. *TransFuser: Imitation with Transformer-Based Sensor Fusion for Autonomous Driving*. 2022. arXiv: 2205.15997 [cs.CV].

[53] Martin Heusel et al. *GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium*. 2018. arXiv: 1706.08500 [cs.LG].

[54] Stephanie Fu* et al. "DreamSim: Learning New Dimensions of Human Visual Similarity using Synthetic Data". In: *arXiv:2306.09344* (2023).

[55] Jiarui Xu et al. *Open-Vocabulary Panoptic Segmentation with Text-to-Image Diffusion Models*. 2023. arXiv: 2303.04803 [cs.CV]. URL: https://arxiv.org/abs/2303.04803.

[56] Robin Rombach et al. "High-Resolution Image Synthesis with Latent Diffusion Models". In: (2022). arXiv: 2112.10752 [cs.CV].

[57] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[58] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. *Adding Conditional Control to Text-to-Image Diffusion Models*. 2023. arXiv: 2302.05543 [cs.CV].

[59] lllyasviel. *ControlNet*. https://huggingface.co/lllyasviel/ControlNet. Accessed: 2024-06-15. 2024.

[60] Andreas Blattmann et al. "Align your Latents: High-Resolution Video Synthesis with Latent Diffusion Models". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023.

[61] Jay Zhangjie Wu et al. "Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 7623–7633.

[62] Xiang* Wang et al. "VideoComposer: Compositional Video Synthesis with Motion Controllability". In: (2023).

[63] Liang Lin et al. "Control-A-Video: Controllable Text-to-Video Generation with Diffusion Models". In: *arXiv preprint arXiv:2305.13840* (2023). URL: https://arxiv.org/abs/2305.13840v1.

[64] Xiaofeng Wang et al. "Drivedreamer: Towards real-world-driven world models for autonomous driving". In: *arXiv preprint arXiv:2309.09777* (2023).

[65] Ruiyuan Gao et al. "Magicdrive: Street view generation with diverse 3d geometry control". In: *arXiv preprint arXiv:2310.02601* (2023).

[66] Han Lin et al. "Ctrl-Adapter: An Efficient and Versatile Framework for Adapting Diverse Controls to Any Diffusion Model". In: *arXiv preprint arXiv:2404.09967* (2024).

[67] Qihang Zhang, Zhenghao Peng, and Bolei Zhou. *Learning to Drive by Watching YouTube Videos: Action-Conditioned Contrastive Policy Pretraining*. 2022. arXiv: 2204.02393 [cs.CV].

[68] Quanyi Li et al. *ScenarioNet: Open-Source Platform for Large-Scale Traffic Scenario Simulation and Modeling*. 2023. arXiv: 2306.12241 [cs.RO].

[69] J. Houston et al. *One Thousand and One Hours: Self-driving Motion Prediction Dataset*. https://woven.toyota/en/prediction-dataset. 2020.

[70] Shiwei Zhang et al. "I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models". In: *arXiv preprint arXiv:2311.04145* (2023).

(a) Original frame (Source:[34]) (b) Generated frame 1 (c) Generated frame 6 (d) Generated frame 12

Figure 1. Augmenting scene in Prescan [37], the new scene is mapped to the photorealstic domain using the video diffusion model. Car in front is idle while the car in the back of the scene is driving away.

(a) Original frame (Source:[34]) (b) Generated frame 1 (c) Generated frame 6 (d) Generated frame 12

Figure 2. Newly rendered video frames based on the original frame, showcasing the model abilites to render scenes in the night.

(a) Original frame (Source:[34]) (b) Generated frame 1 (c) Generated frame 6 (d) Generated frame 12

Figure 3. Augmenting scene in Prescan [51], the new scene is mapped to the photorealstic domain using the video diffusion model now during the day.

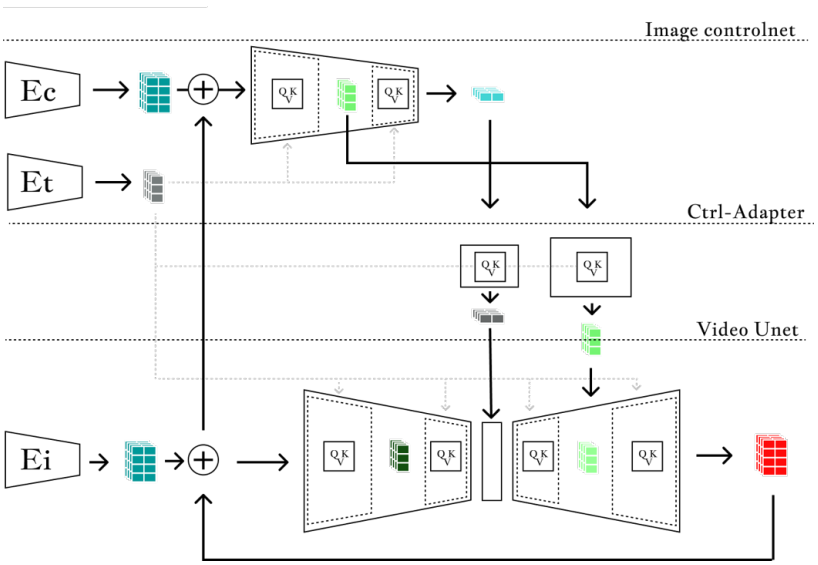


Figure 4. Enlarged view of the video diffusion model where E_c is the encoder for the conditioning, E_t the text encoder, and E_i the reference frame encoder. This is a simplified representation; not all layers are shown, and the dimensions are adjusted for visualization purposes.