DrivingDiffusion: Layout-Guided Multi-View Driving Scenarios Video Generation with Latent Diffusion Model

Xiaofan Li[1], Yifu Zhang[1], and Xiaoqing Ye*[1]

Baidu Inc.

Abstract. With the surge in autonomous driving technologies, the reliance on comprehensive and high-definition bird's-eye-view (BEV) representations has become paramount. This burgeoning need underscores the demand for extensive multi-view video datasets, meticulously annotated to facilitate advanced research and development. Nonetheless, the acquisition of such datasets is impeded by prohibitive costs associated with data collection and annotation. There are two challenges when synthesizing multi-view videos given a 3D layout: Generating multi-view videos involves handling both view and temporal dimensions. How to generate videos while ensuring cross-view consistency and cross-frame consistency? 2) How to ensure the precision of layout control and the quality of the generated instances? Addressing this critical bottleneck, we introduce a novel spatial-temporal consistent diffusion framework, DrivingDiffusion, engineered to synthesize realistic multi-view videos governed by 3D spatial layouts. DrivingDiffusion adeptly navigates the dual challenges of maintaining cross-view and cross-frame coherence, along with meeting the exacting standards of layout fidelity and visual quality. The framework operates through a tripartite methodology: initiating with the generation of multi-view single-frame images, followed by the synthesis of single-view videos across multiple cameras, and culminating with a post-processing phase. We corroborate the efficacy of DrivingDiffusion through rigorous quantitative and qualitative evaluations, demonstrating its potential to significantly enhance autonomous driving tasks without incurring additional costs. `https://drivingdiffusion.github.io`

Keywords: Multi-View Video Generation · Autonomous Driving

1 Introduction

Autonomous driving has drawn extensive attention both from industry and academia for decades. The well-known bird's-eye-view (BEV) is a natural and straightforward candidate view to serve as a unified representation. BEV perception takes multi-view images as input and learns a unified BEV representation for

* Corresponding author.

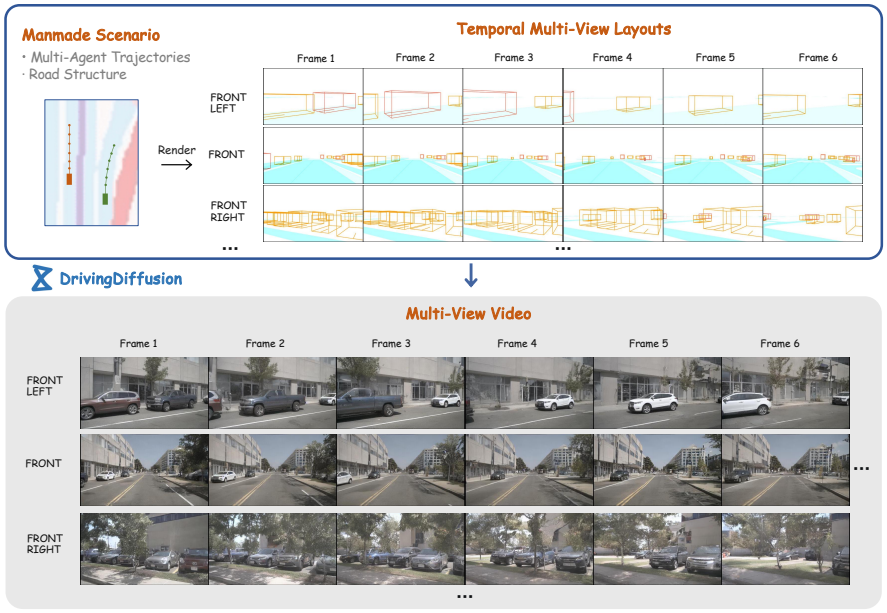


Fig. 1: Examples of the generated multi-view video frames from a man-made 3D layout (including the 3D bounding boxes of obstacles and the road structure) by DrivingDiffusion. The 3D layout is projected to the six camera views using camera parameters. We present the generated results of 3/6 camera views and 6 consecutive frames.

downstream tasks such as 3D object detection [16,18,40], segmentation [4,11,24], tracking [12,45,46] or trajectory prediction [6,13,17]. High-quality and large-scale multi-view video data is critical for learning a powerful BEV representation for multiple perception tasks. However, high-quality data is often difficult to obtain due to the unaffordable annotation costs. As a result, there is a large demand for generating multi-view temporal-consistent video data that conforms to the real distribution and guarantees high consistency with 3D annotations and images.

Previous or simultaneous studies in this domain have explored layout-to-image translation [2, 47], generation of single-view driving videos [1], and the synthesis of images for multi-view driving scenarios as in BEVGen [32], along with concurrent works like BEVControl [41] and MAGICDRIVE [5], the single-view video generation of DriveDreamer [36]. Our DrivingDiffusion addresses the novel challenge of generating multi-view video data within intricate urban environments, surpassing previous methods in complexity (multi-view vs. single-view, multi-frame vs. single-frame, complex urban scenarios vs. simplistic road scenarios), controllability (with 3D layout and high-quality generated instances), and potential applications. The synthesis process is governed by a 3D layout surrounding the ego-vehicle, as illustrated in Figure 1, where our methodology transforms 3D layouts into sequences of multi-view frames. Notably, we pioneer

in multi-view video synthesis, markedly enhancing the quality of the generated content, which bears considerable significance. For instance, this can act as a novel data augmentation avenue for BEV detection/tracking models, facilitating the training of both single-frame and sequential models. Manipulating the 3D layout allows us to amass a substantial volume of multi-view videos, invaluable for model training. Another innovative application is the video editing of specific objects to emulate uncommon scenarios, such as collisions, thereby augmenting simulator capabilities to bolster driving system safety(See Visualizations in Appendix).

First, to ensure a cohesive and realistic output, it is crucial to maintain both cross-view and cross-frame consistency in the generated multi-view images and video. Cross-view consistency means that multi-view images taken by different cameras in the same frame should reflect the same physical location accurately. Simultaneously, cross-frame (or temporal) consistency ensures that the video frames are temporally aligned to produce a visually coherent sequence. Second, the quality of the generated instances (*i.e.* vehicles and pedestrians) within the video is significant, as these instances are the direct subjects of perception tasks and their realism directly impacts the effectiveness of the simulation.

We propose DrivingDiffusion, a multi-stage scheme for generating multi-view videos in complex urban scenes using an artificial 3D layout and a diffusion model. DrivingDiffusion includes multi-view single-frame generation, a shared single-view temporal model, and specialized post-processing for enhanced frame consistency and video extension. Key components include a cross-view attention module for image consistency, geometric constraints for consistency loss, local prompts for improved instance quality, and a Key-Frame controller for temporal consistency. Optical flow priors are embedded for each camera to avoid separate model training, and a "fine-tune" multi-view model enhances frame consistency post-generation. The scheme extends videos by inserting new keyframes through sliding window processing.

In summary, our key contributions are as follows:

(1) We address the novel challenge of generating multi-view, long-duration videos from 3D layouts in complex urban settings. To achieve this, we propose a three-stage generative framework, DrivingDiffusion, specifically designed to ensure the effective generation of multi-view videos for driving scenarios.

(2) We introduce a consistency module and local prompt to enhance the cross-view and cross-frame consistency and the quality of instances in the video.

(3) Our DrivingDiffusion achieve state-of-the-art in the tasks of multi-view image generation, single-view video generation, and multi-view long video generation on nuScenes dataset.

2 Related Work

2.1 Text-to-Image Generation

The text-to-image (T2I) generation task aims at generating realistic images based on text inputs. Early works solve the problem by turning it into a sequence-

to-sequence problem. For example, DALL-E [27] translates text tokens to discrete image embeddings obtained by VQ-VAE [35]. The following works improve the quality of the generated images by more advanced architectures such as image tokenizers [43], encoder-decoder architectures [42] or hierarchical transformers [3]. Recently, Denoising Diffusion Probabilistic Models (DDPM) [8] begin to conquer the T2I task. The quality of the generated image is further increased by utilizing the excellent ability to generate realistic images of diffusion models [20, 29] or improving the text-image alignments [26] using powerful text encoders [25].

2.2 Text-to-Video Generation

The text-to-video (T2V) generation task generates video from the text descriptions Compared with the T2I task, the text-to-video (T2V) generation task is more challenging due to the temporally consistent characteristics involved. Some methods extend T2I methods to perform T2V by changing image tokens to video tokens [37] or equipping the temporal attention modules to T2I models [10]. The following work [38] proposes an auto-regressive framework to solve both T2I and T2V tasks. More recently, Video Diffusion Models [9] utilizes a space-time factorized U-Net with joint image and video data training. Make-A-Video [30] directly translates the tremendous recent progress in T2I to T2V by introducing an effective spatial-temporal module on top of T2I models. Tune-A-Video [39] presents an efficient mechanism for time interaction. Similarly, we also adopt pre-trained T2I models to generate multi-view videos.

2.3 Layout-to-Image generation

This task can be seen as the reverse process of object detection or segmentation, which generates images with the input of bounding boxes or segmentation maps. Previous works utilize GANs [7, 15, 22, 33] or diffusion models [2, 47] to generate synthetic images based on 2D layout. These methods encode the layout as a condition image that is downsampled and upsampled jointly with the data. Recently, BEVGen [32] generates multi-view urban scene images from BEV layout based on VQ-VAE [35]. Compared with BEVGen, we tackle a more complex task that generates temporally consistent multi-view videos. We also adopt diffusion models [9] to generate more realistic videos.

3 Method

In this section, we provide an overview of DrivingDiffusion in Section 3.1. We then detail the pipeline, encompassing the cross-view model, the temporal model, and post-processing, along with the training and inference strategies, which are discussed in Sections 3.2 and 3.3. We also present our methods for enhancing cross-view and cross-frame consistency in the video as well as our local prompt design for improving instance generation quality in Section 3.4 and Section 3.5, respectively. An overview of our approach is shown in Figure 2.

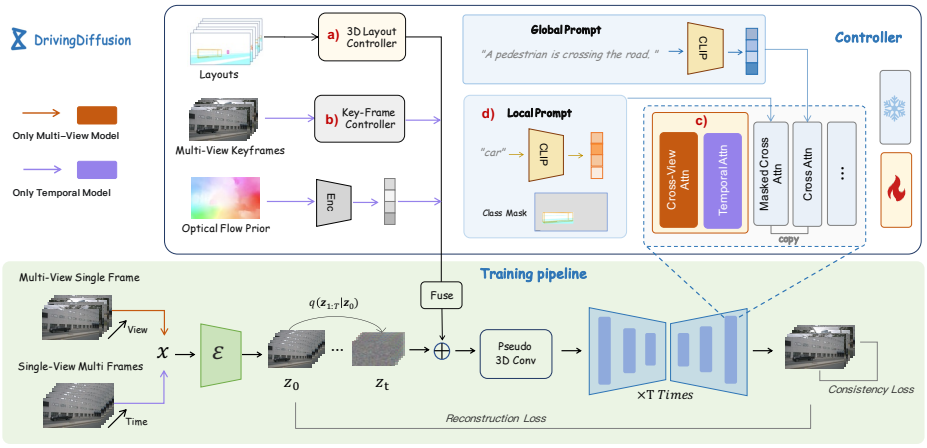


Fig. 2: Diagram of the multi-view model and temporal model in DrivingDiffusion. For training, we separately train the multi-view model and the temporal model. These two models share similar structures, with the exception of the orange and purple components. During the inference stage, the two models are concatenated in a cascaded manner. First, the multi-view model generates the initial multi-view frame of the video. This frame is then set as the keyframe for the temporal model. Lastly, the temporal model generates video frames for each view, forming the multi-view video.

3.1 Overview

base structrue. The popular denoising diffusion models used in image synthesis typically employ a U-Net architecture with spatial downsampling and upsampling, composed of 2D convolutional residual layers and attention blocks, including self-attention and cross-attention mechanisms. For video generation, we extend the pre-trained diffusion model to accommodate temporal and multi-view dimensions using extended $1 \times 3 \times 3$ convolution kernels like VDM [9], and integrate time self-attention layers within transformer blocks. Our model, DrivingDiffusion, incorporates modules for 3D layout, keyframe control, cross-view/frame consistency, and local prompting to enhance instance generation quality, demonstrated to align well with 3D layouts and match real-world data quality.

3D Layout Controller. In order to ensure an accurate mapping between the target instance position in the physical space around the ego-car and the generated image, we project the road structure and each instance onto the 3D layout image using the parameters of the camera. In our approach, we encode essential contextual details—road structure (R), target category (C), and instance ID (I)—into RGB values for each pixel $P_{i,j}$ using an encoding function E. This encoding, $E(P_{i,j}) = \text{concat}(E(R_{i,j}), E(C_{i,j}), E(I_{i,j}))$, ensures that the same instance maintains consistent layout colors across different frames, enhancing temporal coherence. These RGB values serve as inputs for the neural network, enriching the image synthesis process with pixel-level contextual infor-

mation and improving the semantic consistency of generated images. Next, the aforementioned input is fed into a ResNet-like 3D layout controller to encode the U-Net model at different resolutions (*i.e.* 64 × 64, 32 × 32, 16 × 16, 8 × 8) corresponding to different levels of features. We inject this additional control information at different resolutions into each layer of the U-Net model through residual connections, as is shown in a) Figure 2. The advantage of this type of control mechanism is that it is able to influence the diffusion model from various receptive fields without introducing too many parameters (i.e., a trainable copy of ControlNet [44]).

3.2 Multi-View Model and Temporal Model

For the multi-view single-frame model, we employed the 3D layout of all views as input, along with textual descriptions of the scene, to generate highly aligned, multi-view images. At this stage, we only utilized one 3D layout controller.

As for the temporal single-perspective model, we input the 3D layout of all video frames from a single perspective, the first frame image of the video sequence, and the optical flow prior corresponding to the camera. The output is a temporally consistent video from a single viewpoint. Specifically, the added dimension in our model changes from multiple views to multiple frames. Additionally, we include a Key-Frame controller that utilizes the first frame of the multi-view video sequence as a control condition for generating consistent videos. To avoid the need to train a separate video generation model for each camera, statistical optical flow trend at each camera view, with a fixed value as a prior and encoded as a token, is input into the model in the same way as time embedding.

3.3 Post-Processing

In our temporal model, we take the multi-view images at the initial moment as the keyframe. This is because, in a short period of time, we assume that most of the information needed for subsequent frames of each view can be found in the multi-view images at the initial moment. However, when the time interval is too long, there is still a probability that multiple views of the subsequent frames of the video may be inconsistent.

To handle long sequence videos, we train a restoration model, denoted as $R(\cdot)$, subsequent to the cross-view $C(\cdot)$ and temporal $T(\cdot)$ models to enhance the consistency of the following frames. Specifically, the restoration model $R(\cdot)$ integrates the existing structure and parameters of the cross-view model $C(\cdot)$, augmented with an additional controller $K(\cdot)$ for processing frames generated by the temporal model $T(\cdot)$. During training, the restoration model $R(\cdot)$ leverages a substantial dataset of temporal subsequent frames $\{F_t\}_{t=1}^{N}$ generated by $T(\cdot)$ from the original multi-view images $\{V_t\}_{t=1}^{M}$ in the dataset as inputs, and aligns these with the corresponding original images $\{O_t\}_{t=1}^{M}$ in the dataset as ground truth values. The training objective is defined as minimizing the loss function $\mathcal{L}(R(V_t), O_t)$, where $V_t = T(F_{t-1})$ and O_t are the ground truth frames. During

inference, $R(\cdot)$ takes the multi-view images $\{V_t'\}_{t=1}^{M'}$ of subsequent frames of the generated video as input and outputs the restored images $\{R(V_t')\}_{t=1}^{M'}$. Given that this phase is dedicated to refining the generation quality, it is considered an engineering effort and part of the post-processing for long video generation tasks, aiming to bridge the gap between the synthesized and real-world visual fidelity.

Finally, we take a certain moment of subsequent multi-view images as new keyframes and conduct temporal inference again. The video length is extended with the idea of sliding window.

Multi-Stage Pipeline for Inference. We introduce a multi-stage inference strategy to generate multi-view long videos, as shown in Figure 3: 1) We first adopt the multi-view model to generate the first frame panoramic image of the video sequence. 2) Then we use the generated image from each perspective as input for the temporal model, allowing for parallel sequence generation for each corresponding viewpoint. 3) For subsequent frames, we employ the restoration model for parallel inference as well. 4) Extend the video after identifying new keyframes, just like the sliding window algorithm does. Finally we obtain the entire synthetic multi-view video.

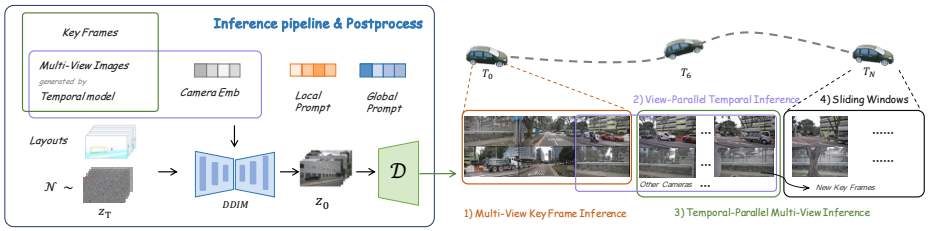


Fig. 3: Diagram of long video generation pipeline. The upper part represents the control inputs under different conditions, while the lower part demonstrates the steps of long video generation.

3.4 Cross-View and Cross-Frame Consistency

As described in section 3.1, the input of 3D layout provides the model with the location, category, and instance information of each target, to some extent, aiding in enhancing local multi-view and temporal consistency. However, this alone is insufficient to ensure global consistency, particularly in static scenes. To address this, we propose an efficient attention mechanism that emphasizes cross-view and cross-frame interactions and introduces geometric constraints to supervise generated results.

The right part of Figure 4 shows the range in which the Consistency Module is in effect: (1) adjacent views in multi-view, (2) the current frame and the first/previous frames in multi-frame.

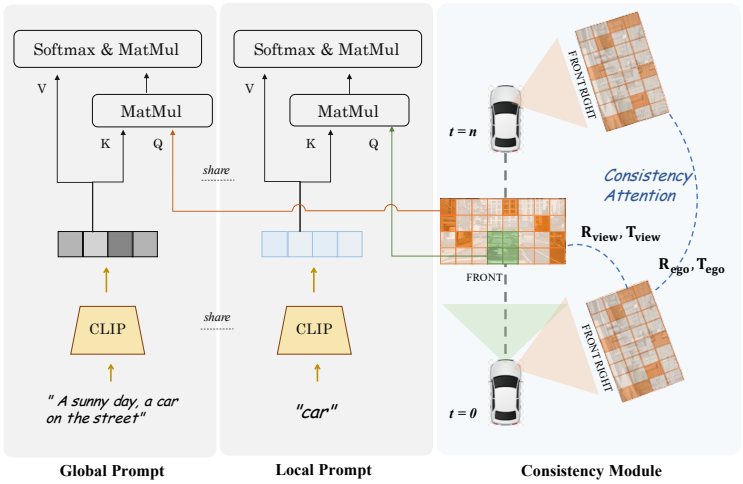


Fig. 4: A schematic of the Consistency module(the right part) and local prompt(the left part) methods.

Consistency Attention. For the cross-view model, we posit that each perspective should primarily focus on its adjacent left and right perspectives. The latent representation of a view and its left and right neighbors in the multi-view model v are denoted as z_v^i, z_v^{i-1}, and z_v^{i+1}.

We attach a consistency attention layer to each attention block to model the new dimensions. For the multi-view model, in order to enhance the cross-view consistency, cross-view attention was calculated using the latent expression of the i *th* view z_v^i and the sum of the latent expression of the two adjacent views z_v^{i-1} and z_v^{i+1}. Our cross-view consistency attention is formulated as:

$$\text{Attention}_{cv}(Q_v, K_v, V_v) = \sigma\left(\frac{\left(W^{Q_v} z_v^i\right)\left(W^{K_v}\left[z_v^{i-1}, z_v^{i+1}\right]\right)^T}{\sqrt{d}}\right) \cdot \left(W^{V_v}\left[z_v^{i-1}, z_v^{i+1}\right]\right) \tag{1}$$

where $[,]$ denotes concatenation operation. W^{Q_v}, W^{K_v}, and W^{V_v} are learnable matrices in the cross-view model that project the inputs to query, key and value, respectively, and d is the output dimension of key and query features. σ denotes the softmax operation.

For the temporal model, the jth frame in the temporal model t such that $j > 1$ is denoted as z_t^j. Cross-frame attention was calculated using the sum of the latent representation of the jth frame and the sum of the first frame with the *(j-1)th* frame. Similarly, we implement cross-frame consistency attention as:

$$\text{Attention}_{cf}(Q_t, K_t, V_t) = \sigma\left(\frac{\left(W^{Q_t} z_t^j\right)\left(W^{K_t}\left[z_t^1, z_t^{j-1}\right]\right)^T}{\sqrt{d}}\right) \cdot \left(W^{V_t}\left[z_t^1, z_t^{j-1}\right]\right) \tag{2}$$

where W^{Q_t}, W^{K_t}, and W^{V_t} are learnable matrices in the temporal model, and σ denotes the softmax operation.



Fig. 5: Street view generation results of the multi-view model. We adopt the 3D layout to generate the images of 6 surrounding cameras in 4 different scenarios. Results with and without text control are displayed separately.

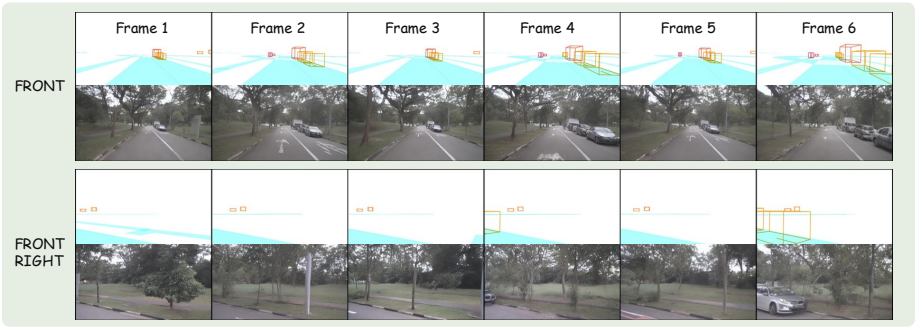


Fig. 6: Street view generation results of the temporal model. We show the visualization results of 6 consecutive frames of 2 adjacent views.

Consistency Loss. We leverage a pre-trained network $\mathcal{F}$ (details in Appendix) to supervise the pose consistency in our generated outputs, addressing both cross-view and temporal dimensions. The network predicts relative poses $(\hat{R}, \hat{t}) = \mathcal{F}(\text{Input}_1, \text{Input}_2)$, where Input_1 and Input_2 are sequential inputs from adjacent cameras or consecutive frames. These predictions are then compared to the ground truth poses (R, t) to compute the loss, ensuring alignment in both spatial (cross-view) and temporal (sequential frames) contexts.

Since camera parameters and ego-pose are known during scene construction, they are utilized for inter-frame and temporal pose supervision, respectively. It's important to note that when predicting cross-view poses, we mask out regions that cannot overlap between two frames to enhance the accuracy of our pose estimation.

Method	Multi-View	Multi-Frame	FID↓	FVD↓
BEVGen [14]	✓		25.54	-
BEVControl [41]	✓		24.85	-
MAGICDRIVE [5]	✓		16.20	-
DriveDreamer [36]		✓	52.6	452
DrivingDiffusion	✓		15.89	-
DrivingDiffusion		✓	15.85	335
DrivingDiffusion	✓	✓	**15.83**	**346**

Table 1: Comparison with other methods on nuScenes validation. The FID indicator and FVD indicator feed back the image and video quality, respectively.

3.5 Local Prompt

To enhance the generation quality of instances, we propose the local prompt module. Reference can be made to part (d) of Figure 2. First, we pre-store the category k and the corresponding text T_k for the category name, and encode the category embedding $F_k = \phi(T_k)$, where ϕ represents the CLIP [25] encoder. For each category k, we use the smallest surrounding rectangular area of the projected 3D layouts as the mask M_k. Cross attention is then computed using the latent representation z and category text encoding E_k in the same manner as the global prompt, with M_k serving as the attention mask. The left part of Figure 4 illustrates the cooperation between global prompt and local prompts.

We did not employ fixed digit encoding during training, as we hypothesized that the concepts of local and global prompts were well-aligned, with the only difference being their respective scopes. Therefore, our local prompt replicates the same structure and parameters as the global prompt, enabling it to fully leverage the parameters of the pre-trained diffusion model, which already encompasses an understanding of our target class.

In summary, our approach of aligning the local and global prompts not only streamlines the training process but also capitalizes on the existing strengths of the pre-trained diffusion model, ensuring that the generated instances are of high quality and closely aligned with the intended target class.

4 Experiment

In this section, we will evaluate DrivingDiffusion based on several metrics. Firstly, we will provide a detailed description of the experimental setup, dataset, and evaluation criteria. Secondly, we will analyze our method qualitatively and compare it quantitatively with other methods. Lastly, we will conduct an ablative analysis, which will be thoroughly discussed and analyzed.

4.1 Dataset

We present the results of generated images from all six camera views trained on the nuScenes dataset. We compared the image quality of our method with other

Method	Road mIoU↑	Vehicle mIoU↑	Drivable mIoU↑	Object NDS↑
Baseline	71.3	36.0	81.7	41.2
BEVGen	50.2 (-21.1%)	5.9 (-39.1%)	-	-
Sparse BEVGen	50.9 (-20.4%)	6.7 (-29.4%)	-	-
BEVControl	60.8(-10.5%)	26.8(-9.2%)	-	-
MAGICDRIVE	61.5(-9.8%)	27.0(-9.0%)		30.6(-10.6%)
DrivingDiffusion	**63.2**(-8.1%)	**31.6**(-4.4%)	**67.8** (-13.9%)	**33.1** (-8.1%)

Table 2: The detection and segmentation results on the nuScenes validation set and the pseudo-validation set generated from the corresponding layout. In green is the drop compared with the standard nuScenes validation data. Road mIoU and Vehicle mIoU are obtained by the CVT [48] segmentation model while Drivable mIoU and Vehicle NDS are obtained by the BEVFusion [19] multi-task model.

multi-view single-frame approaches, and the video quality with other single-view video methods. To evaluate our work, we utilized metrics such as FID and FVD, along with quantitative demonstrations highlighting the proposed method's advantages for autonomous driving. Furthermore, we qualitatively showcased our work through visualization. In multi-view visualizations, we horizontally flipped the back left and right cameras to emphasize our model's consistency across images. Consequently, the side, front, and back cameras align at their outer edges in all figures.

4.2 Experiment Details

We implement our approach based on the official code base for stable diffusion [28] and the publicly available 1.4 billion parameter T2I model. We freeze the image autoencoder and encode each view or video frame individually as a potential representation. Our training regimen for the cross-view model on nuScenes consisted of 50,000 steps, for the temporal model 40,000 steps, and for the fine-tuning model 10,000 steps. We selected six groups of six cameras each with a resolution of 512×512 for training. The training process was executed on eight NVIDIA Tesla A100 GPUs with 40G of memory, and both the cross-view model and cross-frame model were completed within 36 hours. For inference, we utilized a DDIM sampler [31] to generate a multi-view video guided by 3D layouts in the experiment.

4.3 Qualitative Results

We demonstrate the quality and the consistency of single-frame multi-view and single-view video generation in Figure 5 and Figure 6, respectively. We can see that our model obtains both cross-view and cross-frame consistency. The generated instances are accurately controlled by the input 3D layout. When the 3D layout of the vehicle category is very close to the main vehicle, the layout tends to be projected into the images of two perspectives, and even some images are out of the scope. In this case, our model can still generate cross-view

consistent adjacent vehicles. For the quality of instance generation, whether it is the non-rigid pedestrian class, highly occluded/truncated vehicles, or the cone bucket class with less sample size in the dataset, our model can give high-quality instance generation results and maintain self-consistency with the surrounding environment.

4.4 Quantitative Results

To assess the performance of our models, we employ the frame-wise Fréchet Inception Distance (FID) [23] to evaluate the quality of generated images. For video quality evaluation, we utilize the Fréchet Video Distance (FVD) [34]. Unless otherwise noted, all metrics are calculated on the nuScenes validation set.

Due to the lack of related work, we have sorted out the recent or contemporary works and compared the quality of image/video generation under various tasks. For instance, in the domain of autonomous driving scenes, we have analyzed image generation tasks as presented in papers such as BEVGen [32], BEVControl [41], and MAGICDRIVE [5]. In the context of video generation, we have similarly assessed works like DriveDreamer [36].

As presented in Table 1, we made every effort to use the most closely related metrics and settings for comparison. Despite generating both multi-view and multi-frame outputs without performing frame interpolation or super-resolution, our method achieved FID and FVD scores of 15.83 and 346, respectively. These results demonstrate considerable advantages over other approaches. The last three rows in Table 1 show that the indexes of the complete DrivingDiffusion and multi-view/temporal model are similar, proving our method's stability.

Although methods such as FID are commonly used to measure the quality of image synthesis, they do not fully capture the design objectives of our task, nor do they reflect the quality of synthesis across different semantic categories. We take advantage of the official model provided by CVT [48] and BEVFusion [19] as evaluators. We adopted the same set of generated images conditioned on a ground-truth 3D layout as the nuScenes validation set, applied CVT and BevFusion to each set of generated images, and then compared the predicted layout to the ground-truth BEV layout. We report average intersection crossing (mIoU) scores for drivable areas and NDS for all the object classes in Table 2.

We focus on the drop compared to the standard nuScenes validation data. The green numbers in Table 2 represent this drop. Observing the quality of road generation, it is apparent that DrivingDiffusion significantly outperforms other approaches, with a mere -8.1% difference in metrics compared to the baseline. Likewise, in the quality of instance generation, DrivingDiffusion substantially exceeds other methods, with a -4.4% difference from the baseline. These notable enhancements demonstrate the effectiveness of the method we propose.

Effectiveness of Synthesized Data Augmentation Table 3 presents the performance improvement achieved in the BEV detection task through synthesized data-augmentation. In the original training data, there was an issue with long-tail distribution, specifically with small targets, close-range vehicles, and vehicle

Setting	Extra-Data Amount	Object NDS↑	mAOE↓
(a)	0	0.412	0.5613
(b)	2,000	0.418	0.5295
(c)	6,000	0.434	0.5130

Table 3: The impact of data augmentation using synthetic data on detection performance. We use the BEVFusion visual model as the baseline.

orientation angles. We focused on generating additional data for these categories with limited samples to address this. In Setting (b), we only added 2,000 frames of synthetic data, focusing on improving the distribution of obstacle orientation angles. Comparing Setting (a) and (b), and using BEVFusion as the baseline, although NDS only showed a slight improvement, mAOE significantly decreased from 0.5613 to 0.5295. Comparing Setting (a) and (c), by augmenting the data with 6,000 frames of synthetic data, which is more focused on rare scenarios, we observed notable enhancements on the nuScenes validation set. The NDS results increased from 0.412 to 0.434, while the mAOE decreased from 0.5613 to 0.5130. This proves that the data augmentation of synthetic data brings significant benefits to perceptual tasks.

Setting	Extra-Data Amount	AMOTA↑	AMOTP↓	RECALL↑	MOTA↑
(a)	0	0.479	1.227	59.0%	0.435
(b)	2,000	0.485	1.220	60.1%	0.444
(c)	6,000	0.522	1.115	64.6%	0.465

Table 4: The impact of data augmentation using synthetic data on tracking performance. We use the PF-Track visual model as the baseline.

Table 4 presents the performance improvement achieved in the BEV tracking task through synthesized data-augmentation. We also synthesized 2,000 and 6,000 sets of composite data to augment the training data, whose layouts were obtained by applying controlled perturbations to selected original tracking datasets. In our experiments, we utilized PF-Track [21], an open-source tracking tool that has been previously validated on the nuScenes dataset, serving as our baseline algorithm. The metrics considered are AMOTA, AMOTP, RECALL, and MOTA. Comparing settings (a), (b), and (c) reveals that synthetic data significantly enhances tracking tasks. After supplementing with 2,000 and 6,000 sets of data, AMOTA increased from 0.479 to 0.485 and 0.522, respectively.

Ablation Study of Consistency Module and Local Prompt. To verify the validity of our design choices, an ablation study was performed on the key features of the model, including the consistency module and the local prompt. To verify the necessity of the above designs, we conducted ablation experiments on the consistency module and the local prompt. We evaluated both the global quality

by FID and FVD under different settings and the instance quality by Object NDS. The results are shown in Table 5.

Setting	Consistency Attention	Consistency Loss	Local Prompt	FID↓	FVD↓	Object NDS↑
(a)				17.30	420	28.1
(b)	✓	✓		16.67	365	28.9
(c)			✓	17.62	409	32.6
(d)	✓		✓	16.02	359	32.9
(e)		✓	✓	17.00	385	32.8
(f)	✓	✓	✓	**15.83**	**346**	**33.1**

Table 5: Ablation of the consistency module and the local prompt. The Consistency module includes Consistency attention and Consistency loss.

Comparing settings (a) and (b), FID and FVD decreased by 0.63 and 55, signaling that the consistency module significantly enhances image/video quality, particularly impacting FVD due to its influence on video temporal coherence. When comparing Settings (a) and (c), the local prompt notably boosted Object NDS from 28.1 to 32.6, indicating effective instance-level generation guidance. Lastly, Setting (f) with both the consistency module and local prompt yielded the best results: FID, FVD, and NDS were 15.83, 346, and 33.1, respectively.

We also assessed the individual contributions of Consistency Attention and Consistency Loss. When Consistency Attention was removed, comparing Settings (e) and (f), there was a significant decline in the performance across all metrics, with FID increasing to 17.00, FVD to 385, and a slight decrease in Object NDS to 32.8. This underscores the vital role of Consistency Attention in enhancing the model's fidelity and consistency. In contrast, comparing Settings (d) and (f), the removal of Consistency Loss had a less pronounced, yet noticeable impact on performance. FID increased to 16.02, FVD to 359, and Object NDS slightly decreased to 32.9. These results emphasize that while both components positively contribute to the model, Consistency Attention plays a more critical role in optimizing the quality of the generated instances.

5 Conclusion

Our work is the first to realize multi-view video generation of autonomous driving scenes and implement precise layout control, which is of great significance for autonomous driving data generation and simulation. Our paradigm can be easily extended to the depth map, point cloud, occupancy, rendered assets, etc. for control of video generation. Furthermore, our approach can be trained using autoregressive techniques on a substantial number of videos(preferably those obtained through data mining). This not only allows us to get a world model that can predict future videos but also makes it an excellent pre-trained model for subsequent downstream tasks. This may be due to the model's enhanced understanding of driving scenarios by predicting future events.

References

1. Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. arXiv preprint arXiv:2304.08818 (2023)
2. Cheng, J., Liang, X., Shi, X., He, T., Xiao, T., Li, M.: Layoutdiffuse: Adapting foundational diffusion models for layout-to-image generation. arXiv preprint arXiv:2302.08908 (2023)
3. Ding, M., Zheng, W., Hong, W., Tang, J.: Cogview2: Faster and better text-to-image generation via hierarchical transformers. arXiv preprint arXiv:2204.14217 (2022)
4. Fang, S., Wang, Z., Zhong, Y., Ge, J., Chen, S., Wang, Y.: Tbp-former: Learning temporal bird's-eye-view pyramid for joint perception and prediction in vision-centric autonomous driving. arXiv preprint arXiv:2303.09998 (2023)
5. Gao, R., Chen, K., Xie, E., Hong, L., Li, Z., Yeung, D.Y., Xu, Q.: Magicdrive: Street view generation with diverse 3d geometry control (2023)
6. Gu, J., Hu, C., Zhang, T., Chen, X., Wang, Y., Wang, Y., Zhao, H.: Vip3d: End-to-end visual trajectory prediction via 3d agent queries. arXiv preprint arXiv:2208.01582 (2022)
7. He, S., Liao, W., Yang, M.Y., Yang, Y., Song, Y.Z., Rosenhahn, B., Xiang, T.: Context-aware layout to image generation with enhanced object appearance. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 15049–15058 (2021)
8. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems **33**, 6840–6851 (2020)
9. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. arXiv:2204.03458 (2022)
10. Hong, W., Ding, M., Zheng, W., Liu, X., Tang, J.: Cogvideo: Large-scale pretraining for text-to-video generation via transformers. arXiv preprint arXiv:2205.15868 (2022)
11. Hu, A., Murez, Z., Mohan, N., Dudas, S., Hawke, J., Badrinarayanan, V., Cipolla, R., Kendall, A.: Fiery: future instance prediction in bird's-eye view from surround monocular cameras. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15273–15282 (2021)
12. Hu, H.N., Yang, Y.H., Fischer, T., Darrell, T., Yu, F., Sun, M.: Monocular quasi-dense 3d object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(2), 1992–2008 (2022)
13. Jiang, B., Chen, S., Wang, X., Liao, B., Cheng, T., Chen, J., Zhou, H., Zhang, Q., Liu, W., Huang, C.: Perceive, interact, predict: Learning dynamic and static clues for end-to-end motion prediction. arXiv preprint arXiv:2212.02181 (2022)
14. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: Pointpillars: Fast encoders for object detection from point clouds. In: CVPR. pp. 12697–12705 (2019)
15. Li, Z., Wu, J., Koh, I., Tang, Y., Sun, L.: Image synthesis from layout with locality-aware mask adaption. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13819–13828 (2021)
16. Li, Z., Wang, W., Li, H., Xie, E., Sima, C., Lu, T., Yu, Q., Dai, J.: Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. arXiv preprint arXiv:2203.17270 (2022)

17. Liang, M., Yang, B., Zeng, W., Chen, Y., Hu, R., Casas, S., Urtasun, R.: Pnpnet: End-to-end perception and prediction with tracking in the loop. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11553–11562 (2020)
18. Liu, Y., Wang, T., Zhang, X., Sun, J.: Petr: Position embedding transformation for multi-view 3d object detection. arXiv preprint arXiv:2203.05625 (2022)
19. Liu, Z., Tang, H., Amini, A., Yang, X., Mao, H., Rus, D., Han, S.: Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. arXiv preprint arXiv:2205.13542 (2022)
20. Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., Chen, M.: Glide: Towards photorealistic image generation and editing with text-guided diffusion models. arXiv preprint arXiv:2112.10741 (2021)
21. Pang, Z., Li, J., Tokmakov, P., Chen, D., Zagoruyko, S., Wang, Y.X.: Standing between past and future: Spatio-temporal modeling for multi-camera 3d multi-object tracking (2023)
22. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2337–2346 (2019)
23. Parmar, G., Zhang, R., Zhu, J.Y.: On aliased resizing and surprising subtleties in gan evaluation. In: CVPR (2022)
24. Philion, J., Fidler, S.: Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In: ECCV (2020)
25. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
26. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 (2022)
27. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: International Conference on Machine Learning. pp. 8821–8831. PMLR (2021)
28. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR. pp. 10684–10695 (2022)
29. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S.K.S., Ayan, B.K., Mahdavi, S.S., Lopes, R.G., et al.: Photorealistic text-to-image diffusion models with deep language understanding. arXiv preprint arXiv:2205.11487 (2022)
30. Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al.: Make-a-video: Text-to-video generation without text-video data. arXiv preprint arXiv:2209.14792 (2022)
31. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)
32. Swerdlow, A., Xu, R., Zhou, B.: Street-view image generation from a bird's-eye view layout. arXiv preprint arXiv:2301.04634 (2023)
33. Sylvain, T., Zhang, P., Bengio, Y., Hjelm, R.D., Sharma, S.: Object-centric image generation from layouts. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 2647–2655 (2021)
34. Unterthiner, T., Van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., Gelly, S.: Towards accurate generative models of video: A new metric & challenges. arXiv preprint arXiv:1812.01717 (2018)

35. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. Advances in neural information processing systems **30** (2017)
36. Wang, X., Zhu, Z., Huang, G., Chen, X., Lu, J.: Drivedreamer: Towards real-world-driven world models for autonomous driving (2023)
37. Wu, C., Huang, L., Zhang, Q., Li, B., Ji, L., Yang, F., Sapiro, G., Duan, N.: Godiva: Generating open-domain videos from natural descriptions. arXiv preprint arXiv:2104.14806 (2021)
38. Wu, C., Liang, J., Ji, L., Yang, F., Fang, Y., Jiang, D., Duan, N.: Nüwa: Visual synthesis pre-training for neural visual world creation. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVI. pp. 720–736. Springer (2022)
39. Wu, J.Z., Ge, Y., Wang, X., Lei, W., Gu, Y., Hsu, W., Shan, Y., Qie, X., Shou, M.Z.: Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. arXiv preprint arXiv:2212.11565 (2022)
40. Xiong, K., Gong, S., Ye, X., Tan, X., Wan, J., Ding, E., Wang, J., Bai, X.: Cape: Camera view position embedding for multi-view 3d object detection. arXiv preprint arXiv:2303.10209 (2023)
41. Yang, K., Ma, E., Peng, J., Guo, Q., Lin, D., Yu, K.: Bevcontrol: Accurately controlling street-view elements with multi-perspective consistency via bev sketch layout (2023)
42. Yu, J., Li, X., Koh, J.Y., Zhang, H., Pang, R., Qin, J., Ku, A., Xu, Y., Baldridge, J., Wu, Y.: Vector-quantized image modeling with improved vqgan. arXiv preprint arXiv:2110.04627 (2021)
43. Yu, J., Xu, Y., Koh, J.Y., Luong, T., Baid, G., Wang, Z., Vasudevan, V., Ku, A., Yang, Y., Ayan, B.K., et al.: Scaling autoregressive models for content-rich text-to-image generation. arXiv preprint arXiv:2206.10789 (2022)
44. Zhang, L., Agrawala, M.: Adding conditional control to text-to-image diffusion models. arXiv preprint arXiv:2302.05543 (2023)
45. Zhang, T., Chen, X., Wang, Y., Wang, Y., Zhao, H.: Mutr3d: A multi-camera tracking framework via 3d-to-2d queries. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4537–4546 (2022)
46. Zhang, Y., Wang, X., Ye, X., Zhang, W., Lu, J., Tan, X., Ding, E., Sun, P., Wang, J.: Bytetrackv2: 2d and 3d multi-object tracking by associating every detection box. arXiv preprint arXiv:2303.15334 (2023)
47. Zheng, G., Zhou, X., Li, X., Qi, Z., Shan, Y., Li, X.: Layoutdiffusion: Controllable diffusion model for layout-to-image generation. arXiv preprint arXiv:2303.17189 (2023)
48. Zhou, B., Krähenbühl, P.: Cross-view transformers for real-time map-view semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13760–13769 (2022)