

A. More Implementation Details

In this section, we provide additional implementation specifics for the core components of DriveLaW, including the video generation backbone (DriveLaW-Video), the trajectory planning module (DriveLaW-Act), and the motion-conditioned prompting mechanism.

A.1. DriveLaW-Video: Video Generation Backbone

DriveLaW-Video adopts a diffusion-based architecture optimized for high-compression spatiotemporal encoding and efficient chained generation with the downstream planner. This design balances computational efficiency and generation quality, enabling long-horizon driving scenario synthesis under practical hardware constraints.

Spatiotemporal VAE and Compression Optimization.

The Video-VAE serves as the core spatiotemporal compression module, applying a $32 \times 32 \times 8$ spatial-temporal downsampling with 128 output channels. This configuration achieves a total compression ratio of 1:192 (pixels-to-tokens ratio of 1:8192), approximately twice the compression rate of common text-to-video pipelines. To enable such aggressive compression without compromising generation fidelity, we introduce the following architectural and training modifications:

- **Causal 3D Encoder:** Ensures each temporal step depends only on current and past frames, preserving the autoregressive consistency critical for driving prediction tasks.
- **Hybrid Decoding Strategy:** Instead of completing all denoising steps in the latent space, the final rectified-flow step (t_1) is executed by the VAE decoder directly in the pixel space. This design recovers high-frequency details (e.g., road texture, reflections, traffic signs) without requiring a separate super-resolution stage.
- **Reconstruction GAN:** The discriminator receives paired real-reconstructed samples and focuses on fine-grained detail differences. This improves training stability and perceptual quality under high compression.
- **Multi-layer Noise Injection:** Introduces per-channel learned stochasticity in the decoder, enhancing the diversity of synthesized textures.
- **Uniform Log-variance Across Channels:** Ensures balanced KL regularization and avoids underutilized latent dimensions, improving the efficiency of the latent space.
- **Video-DWT Loss:** Complements MSE and perceptual losses by explicitly penalizing high-frequency errors across eight 3D wavelet sub-bands, strengthening the preservation of structural details.

Video Transformer Backbone. The diffusion backbone adopts a 3D Transformer architecture adapted from PixArt-

α , with 28 self-cross attention blocks, a hidden size of 2048, a feed-forward expansion factor of $\times 4$, and RMSNorm normalization in place of LayerNorm for better stability.

To maintain spatial-temporal consistency across varying resolutions and durations, we employ normalized fractional Rotary Positional Embeddings (RoPE) computed with exponential frequency spacing. Unlike patchifier-based designs (e.g., $2 \times 2 \times 1$ patch size), tokens are serialized directly from the VAE latents at a $1 \times 1 \times 1$ granularity, eliminating redundant patchification operations and preserving geometric consistency.

A.2. DriveLaW-Act: Trajectory Planning Module

DriveLaW-Act is implemented as a lightweight diffusion planner (133M parameters) that is tightly integrated with the DriveLaW-Video backbone. Its key design details are as follows:

Input Conditioning. The planner is directly conditioned on cached Video-DiT latents from the first denoising step. These latents encode rich scene information, including current geometry and agent dynamics, and serve as keys in the planner’s cross-attention mechanism, paired with the trajectory noise input. Additionally, the planner receives structured context embeddings, including: ego-vehicle kinematics and navigation commands.

Training and Inference. The planner is trained with a flow-matching objective to generate smooth, physically consistent trajectories. It predicts continuous (x, y, θ) positions at a sampling rate of 2 Hz over a 4 s planning horizon. During inference, the planner operates purely in the latent space without requiring video decoding, significantly reducing computational overhead. Notably, gradient isolation between the video generation and planning modules is preserved during training, ensuring stable optimization of each component.

A.3. Motion-Conditioned Prompting Mechanism

To align video generation with realistic driving dynamics, we design a structured motion-conditioned prompting mechanism that unifies dynamic ego-state information and static scene context into interpretable text guidance for the Video-DiT.

Prompt Construction Logic. Ego-state numerical variables (speed, steering angle, displacement) are first discretized into semantic bins (e.g., "low speed", "steady motion", "turning left/right"). These semantic labels are integrated into a fixed prompt template, which also includes technical numerical grounding to ensure precise control. The template is defined as follows:

A high-quality, photorealistic dashboard camera view of autonomous driving. Based on the past T_h seconds, predict and generate the next T_p seconds of realistic driving continuation, moving at [speed bin] with [motion descriptor], smoothly continue for the next T_p seconds. Maintain temporal consistency, stable camera perspective, natural motion flow without jitter or artifacts, clear details, and realistic physics. [Technical: forward Δx m, lateral Δy m, yaw $\Delta \theta^\circ$, speed v m/s]

The text prompt is encoded by the frozen T5-XXL encoder, and the resulting embeddings are injected via cross-attention into all layers of the Video-DiT. This allows semantic motion cues to modulate the generation process, ensuring alignment between the synthesized video and the ego-vehicle’s dynamic constraints.

B. Additional Experimental Results

B.1. Qualitative analysis of latent representations.

To demonstrate that VGM (Video Generation Model) latent features can serve as more efficient and informative conditions for action learning, we conduct a systematic analysis. As shown in Fig. 4, we visualize and compare three types of latent representations. We apply PCA (Principal Component Analysis) [1] to project each representation to 3 principal components mapped to RGB channels, all upsampled to 1280×704 (Note that for BEV features, limited by the single-view visual input, we extract intermediate backbone features before the BEV query transformation to ensure fair comparison). The visualization clearly shows that BEV and VLM features are diffuse, unstable, and exhibit irregular focus patterns. In contrast, VGM features are sharper, less noisy, and demonstrate superior semantic coherence with strong spatial structure awareness, even under challenging driving conditions. This suggests that VGM features provide a more suitable representation for action learning in autonomous driving.

B.2. Quantitative Evaluation of Video Generation

Tab. 8 presents extensive quantitative evaluation of video generation quality across multiple datasets and horizon lengths.

Table 8. **Quantitative evaluation of video generation.** We report FID and FVD on NuScenes and OpenDV, and FVD at varying horizon lengths on NuPlan.

Methods	NuScenes		OpenDV		NuPlan			
	FID↓	FVD↓	FID↓	FVD↓	FVD ₂₄ ↓	FVD ₄₀ ↓	FVD ₈₀ ↓	FVD ₁₀₀ ↓
Epona	7.5	82.8	6.9	80.7	61.3	74.9	239.6	277.3
DriveLaW	4.6	81.3	4.6	72.9	55.6	71.2	230.2	296.1

Cross-Dataset Generalization. In Tab. 8, we evaluate zero-shot generalization on the OpenDV dataset following GEM (CVPR 2025). DriveLaW-Video outperforms Epona on OpenDV, indicating robust generalization beyond the training domains of NuScenes/NuPlan.

Long-Horizon Generation. Tab. 8 reports FVD at varying prediction horizons (24, 40, 80, and 100 frames) on NuPlan. DriveLaW consistently outperforms Epona up to 80 frames and Epona shows better performance at 100 frames. Moreover, Epona is substantially slower (Tab. 10), and this gap increases with horizon length. Considering both quality and efficiency, DriveLaW provides a more practical trade-off for realistic driving horizons.

B.3. Inference Speed Analysis

To evaluate the efficiency of our video generation stage, we compare the per-frame speed of DriveLaW with the unified world-model baseline Epona under identical experimental settings: single NVIDIA 4090 GPU, 30 DiT sampling steps, and matching resolutions as listed in Tab. 9.

Table 9. Video generation speed per frame on a single NVIDIA 4090 GPU with 30 DiT sampling steps.

Method	Resolution	Params	Times
Epona	1024 × 512	~ 1.9B	0.88 s
	768 × 512		0.12 s
DriveLaW (Ours)	1024 × 512	~ 2.0B	0.18 s
	1280 × 704		0.39 s

As shown in Tab. 9, DriveLaW achieves substantially faster generation at lower resolutions. For 768 × 512, DriveLaW requires only 0.12 s per frame, while at 1024 × 512 the speed remains modest at 0.18 s, despite the model size being slightly larger than Epona’s. At the highest resolution (1280 × 704), DriveLaW achieves 0.39 s per frame, which is more than twice as fast as Epona’s result 0.88 s, even though our output resolution is significantly higher.

These results indicate that the proposed architectural optimizations, which include a higher compression ratio and hybrid decoding, preserve runtime efficiency across resolutions. This allows DriveLaW to deliver competitive generation speed while maintaining high video fidelity.

B.4. Runtime Performance on H20 GPU

For completeness, we also report inference speed on an NVIDIA H20 GPU in Table 10.

B.5. Ablation on Noise Reinjection Usage

We conduct an ablation study to evaluate the effect of enabling or disabling the proposed noise reinjection mechanism on video generation quality. The experiments are

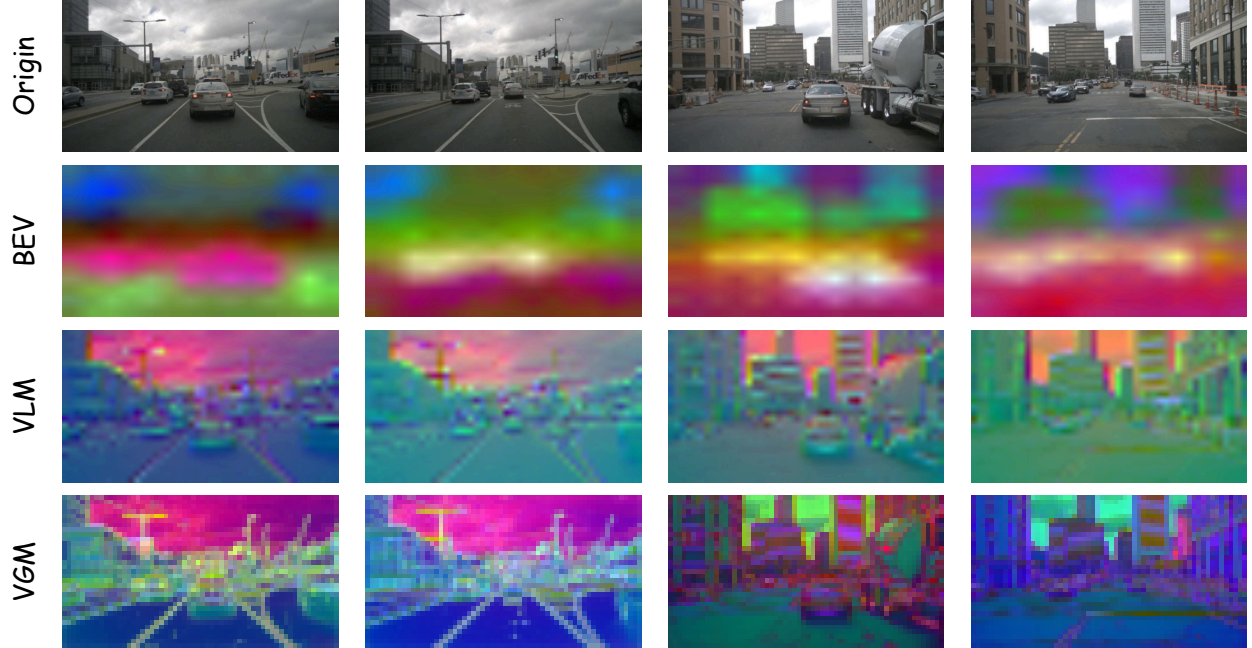


Figure 4. **Qualitative analysis of latent representations.** We visualize the quality of latent representations from three different feature sources: perspective-view features extracted from BEVFormer [45]’s ResNet-101 backbone, VLM features from the pretrained Qwen2.5-VL model in ReCogDrive [44], and VGM (Video Generation Model) features from our DriveLaW-Video. To enable visual comparison, we apply PCA to reduce each representation to its top 3 principal components and map them to RGB channels. From top to bottom, each row displays: (1) the original input frame, (2) BEV features, (3) VLM features, and (4) VGM features, all upsampled to 1280×704 for visualization. While the BEV and VLM features appear diffuse, unstable, and exhibit irregular focus shifts, our VGM features are notably sharper, contain significantly less noise, and demonstrate superior semantic coherence with strong spatial structure awareness, even under severe driving motion.

Table 10. **Runtime performance on an NVIDIA H20 GPU.** We report trajectory planning time, and per-frame video generation time.

Method	Resolution	Params	Traj. (s)	Frame (s)
Epona	1024 × 512	~ 1.9B	0.42	1.06
DriveLaW (Ours)	1024 × 512	~ 2.0B	0.71	0.21

performed on the nuScenes validation set, with FID and FVD as evaluation metrics.

Table 11. Effect of enabling noise reinjection on driving video generation quality.

Setting	FID ↓	FVD ↓
w/o Noise Reinjection	6.1	102.1
w/ Noise Reinjection (Ours)	4.6	81.3

As shown in Tab. 11, removing noise reinjection results in a noticeable degradation in temporal coherence and a slight decline in spatial fidelity. By selectively perturbing high-

frequency regions before each denoising step, our method compels the generator to actively regenerate fine details, thereby improving both sharpness and temporal stability while reducing artifacts.

C. More Qualitative Results

In this section, we present additional qualitative examples to further illustrate the capabilities of DriveLaW in diverse driving scenarios and planning tasks.

C.1. Video Generation on nuScenes

We evaluate DriveLaW on the nuScenes validation set across a wide variety of real-world driving scenarios. As shown in Fig. 5, our results demonstrate that the model maintains temporal coherence, fine-grained spatial detail, and robust performance across diverse visual conditions.

C.2. Planning Results Visualization

As shown in Fig. 6, we present representative cases from the Navtest splits, highlighting DriveLaW’s ability to predict future trajectories while ensuring safety and smoothness.

C.3. Supplementary Video Demonstrations

To facilitate clearer understanding, we provide 6 MP4 demo videos in the supplementary material, including 4 normal-driving scenarios and 2 rainy-weather scenarios. These examples help reviewers and readers visually assess temporal consistency, spatial detail, and the practical utility of planning outputs in diverse and challenging driving conditions.

D. Limitations and Future Work

While DriveLaW demonstrates strong performance in video generation and trajectory planning, we acknowledge several limitations that present opportunities for future research.

D.1. Motion Artifacts in High-Compression VAE.

To achieve efficient inference, DriveLaW employs a high-compression Video-VAE with a $32 \times 32 \times 8$ downsampling factor. Our experiments reveal that such aggressive compression introduces noticeable artifacts during reconstruction, particularly in high-motion scenarios. These artifacts propagate to the video generation stage, manifesting as visual distortions during rapid ego-motion or dynamic agent interactions. Although our proposed noise reinjection mechanism mitigates this issue to some extent (Tab. 11), it does not fundamentally resolve the underlying limitation. We plan to address this through architectural improvements and advanced training strategies in future work.

D.2. Inference Latency.

Despite our optimizations (*e.g.*, high-compression VAE, resolution scaling, and hybrid decoding), DriveLaW’s inference speed remains slower than end-to-end planning models that bypass explicit video generation. This gap stems from the inherent computational demands of diffusion-based video generation.

D.3. Scalability and Future Outlook.

Notwithstanding these limitations, DriveLaW’s primary advantage lies in its *scalability* with rapid advances in video generation technology. As foundational video models continue to improve in quality, speed, and efficiency, DriveLaW’s performance will advance commensurately without requiring architectural redesign. Furthermore, the paradigm enables the research community to leverage powerful pre-trained video generators to rapidly develop generalizable planners with minimal domain-specific training. With anticipated improvements in inference acceleration techniques (*e.g.*, distillation, quantization, and dedicated hardware), we envision the DriveLaW paradigm becoming viable for on-board deployment in the near future.



(a) Conventional urban driving scenarios.



(b) Complex urban driving scenarios.



(c) Night driving scenarios.

Figure 5. **Qualitative examples of DriveLaW video generation on the nuScenes dataset.** (a) Conventional urban driving scenarios, showing stable lane keeping and interactions with surrounding traffic. (b) Complex urban driving scenarios involving dense multi-agent interactions, turning maneuvers, and occlusions. (c) Night driving scenarios, demonstrating the model’s robustness to low-light conditions while preserving temporal consistency and fine details.



(a) Go straight

(b) Turning

(c) Intersection

Figure 6. Qualitative results on the Navtest benchmark.