ReSim: Reliable World Simulation for Autonomous Driving

Jiazhi Yang[1,3] **Kashyap Chitta**[4,7] **Shenyuan Gao**[8] **Long Chen**[5] **Yuqian Shao**[6]
Xiaosong Jia[6] **Hongyang Li**[2] **Andreas Geiger**[7] **Xiangyu Yue**[1†] **Li Chen**[2,3†]

[1]The Chinese University of Hong Kong [2]The University of Hong Kong
[3]OpenDriveLab at Shanghai AI Lab [4]NVIDIA Research [5]Xiaomi EV
[6]Shanghai Jiao Tong University [7]University of Tübingen, Tübingen AI Center [8]HKUST

https://opendrivelab.com/ReSim

Abstract

How can we reliably simulate future driving scenarios under a wide range of ego driving behaviors? Recent driving world models, developed exclusively on real-world driving data with expert trajectories, struggle to represent hazardous or non-expert behaviors that are rare in training corpus. This limitation restricts their applicability to tasks such as policy evaluation. In this work, we address this challenge by enriching real-world human demonstrations with diverse non-expert data collected from a driving simulator (e.g., CARLA), and building a controllable world model trained on this heterogeneous corpus. Starting with a video generator featuring a diffusion transformer architecture, we devise several strategies to effectively integrate conditioning signals and improve prediction controllability and fidelity. The resulting model, **ReSim**, enables **Re**liable **Sim**ulation of diverse open-world driving scenarios under various actions, including hazardous non-expert ones. To close the gap between high-fidelity simulation and applications that require reward signals to judge different actions, we introduce a Video2Reward module that estimates a reward from ReSim's simulated future. Our ReSim paradigm achieves up to 44% higher visual fidelity, improves controllability for both expert and non-expert actions by over 50%, and boosts planning and policy selection performance on NAVSIM by 2% and 25%, respectively.

1 Introduction

Learning a world model capable of predicting plausible future outcomes is now envisioned as a key milestone in achieving autonomy [1, 2, 3]. Over the past decade, researchers have leveraged visual world models to learn compact representations [4, 5], guide test-time planning [6, 7, 8], and develop reinforcement learning agents [9, 10] across various domains [11, 12, 13]. Unlike general-purpose video generators, which prioritize visual fidelity and generalization, world models simulate futures with precise control over ego actions.

In the autonomous driving domain, recent driving world models have also made rapid improvements in visual fidelity and generalization by scaling to massive driving datasets [14, 15, 16] and integrating frontier video generation techniques [17, 18]. However, the ability to accurately follow actions, which is an essential requirement for precise reward estimation and effective planning [19, 20], remains challenging [21]. As real-world data continues to grow, a critical question emerges: *Is real-world human data alone sufficient to guarantee simulation reliability?* A notable limitation of real-world data is that it predominantly consists of safe *expert* demonstrations, where the state-action space is inherently restricted by safety and regulation concerns [22, 23]. Consequently, safety-critical or

[†]Corresponding authors. Primary contact to Jiazhi Yang at: jzyang@link.cuhk.edu.hk

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

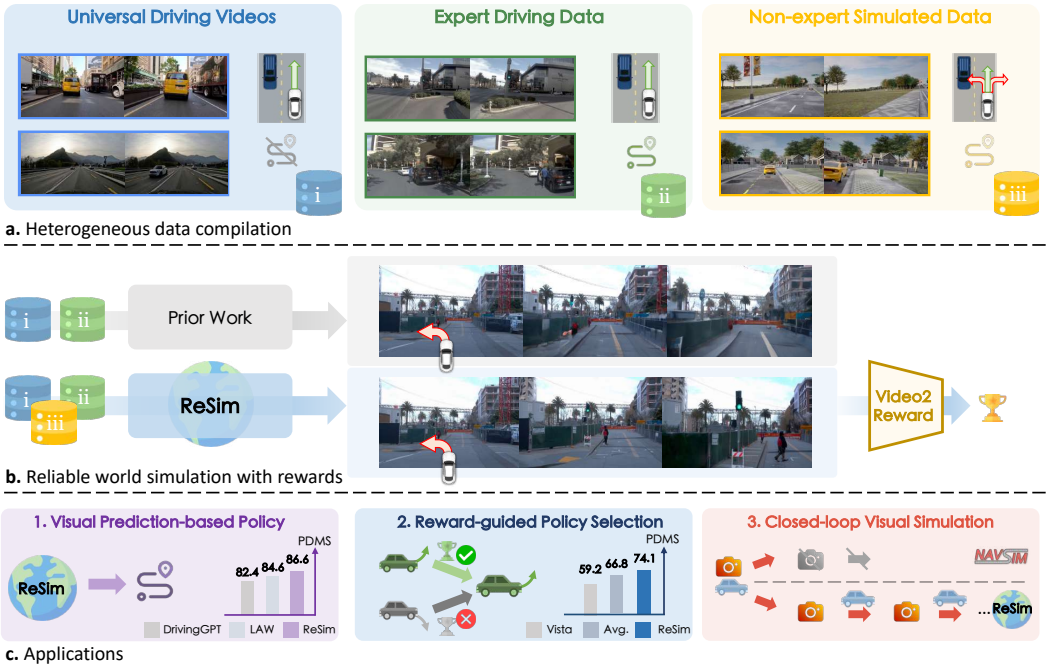

Figure 1: **Overview of ReSim.** **(a)** Heterogeneous driving data includes (i,ii) experts' safe driving logs, and (iii) potentially dangerous (non-expert) driving behaviors from simulations. **(b)** Prior driving world models are trained on expert data solely, leading to consistently safe yet inaccurate imaginations; in ReSim, we leverage all sources of data to simulate reliable and realistic futures, and build a robust reward model that generalizes to open-world scenarios within the simulator. **(c)** The high-fidelity prediction, accurate action-following, and reward estimation abilities of ReSim facilitate driving applications related to both policy deployment and simulation.

hazardous events (*e.g.*, collisions, off-road deviations) are significantly underrepresented [24, 25, 26]. This imbalance leads to severe hallucinations when the world model is exposed to unseen *non-expert* actions in certain states, undermining its robustness and reliability [27, 28].

To address the problem, we present **ReSim**, a reliable driving world model that can be steered by various actions, including out-of-distribution ones, while achieving high-fidelity simulation results. Our approach first enriches real-world human driving logs with non-expert data gathered from a driving simulator [29], where agents can execute a broader spectrum of actions without safety concerns. The resulting training corpus illustrated in Fig. 1(a) covers a wide spectrum of scenarios and actions (including non-expert ones), and further supports the simulation reliability for our world model. Built upon a scalable text-to-video generator [30], ReSim applies a multi-stage training pipeline to integrate visual and action conditions. We devise an unbalanced noise sampling strategy along with a dynamics consistency loss to emphasize the learning of motion coherence, especially when being applied to non-expert actions with significant visual changes. As showcased in Fig. 1(b), prior works like Vista [16] fail to follow the specified steering action, while ReSim accurately simulates off-road behaviors. Moreover, making the world model beneficial for real-world driving often requires reward estimation [1, 10, 31, 8], which judges the quality of different actions to guide decisions. Therefore, we develop a Video2Reward model to convert simulated video outputs from ReSim into scalar rewards in real-world scenarios.

Based on the above explorations, we further demonstrate applications for supporting real-world autonomous driving in various aspects, as depicted in Fig. 1(c). In scenarios where action conditioning is absent, the simulated future of ReSim can serve as a visual plan from which an executable ego trajectory can be derived. On the NAVSIM planning benchmark [23], with front-view sensory videos only, our video prediction-based policy achieves an improvement of +2.0 compared to state-of-the-art world model-based planners [32] and +2.6 compared to an end-to-end baseline [33] with supervised learning. Additionally, the integration of ReSim and the Video2Reward model offers a solution for selecting the trajectory with the highest estimated reward among those generated by candidate policies, thereby justifying and guiding the final decision. In our experiments, this policy selection

process leads to a performance boost of 55.3% in comparison to the weak candidate policies. More intuitively, our system offers a synthetic environment where we can validate the behavior of a learned driving policy by running it within the imagination of ReSim in a closed-loop manner.

Contributions. **(1)** While prior works either use simulated or real-world data separately to develop driving world models, we demonstrate that integrating both sources can alleviate the shortage of unsafe driving behaviors in real-world data, and can improve the model's action controllability in real-world scenarios. **(2)** We present ReSim, a controllable world model that reliably simulates high-fidelity future outcomes by precisely executing diverse action inputs, together with a comprehensive training recipe including an improved loss formulation and noise sampling strategy for incorporating condition inputs and capturing scenario dynamics. Rewards can be derived from the simulated futures via a Video2Reward model. **(3)** We applying ReSim to facilitate driving in real-world scenarios, and validate its effectiveness via benchmarking on a wide array of datasets and tasks, where it exhibits evident improvements over previous counterparts.

2 Reliable Driving World Simulation

We outline the ReSim framework as follows. In Sec. 2.1, we introduce the heterogeneous training data with a wide range of scenarios and actions. We instantiate ReSim on a diffusion transformer architecture with careful modifications to capture dynamic driving scenarios and enable accurate action conditioning in Sec. 2.2. We propose to derive rewards from the simulated results of ReSim via a Video2Reward model in Sec. 2.3. Furthermore, we demonstrate the applicability of our method in real-world driving applications in Sec. 2.4. More implementation details are in Appendix Sec. B.

2.1 Heterogeneous Data Compilation

Existing driving world models are typically developed on public autonomous driving datasets with expert trajectories [34, 35] and web videos [14]. Similarly, in this work, we compile these two sources, specifically NAVSIM [23] and OpenDV [14], within our training data. NAVSIM contains rigorously labeled actions for action control learning, while the large-scale OpenDV dataset supports generalization of the world model. However, as shown in Fig. 1(a), both sources are dominated by human behaviors. The lack of non-expert actions limits prior world models' ability to emulate non-expert behaviors and their corresponding outcomes, such as collisions. This issue further hinders the world models from effectively identifying an inferior driving policy and providing reliable rewards.

To address this limitation, we leverage a driving simulator, *i.e.*, CARLA [29], to gather data in synthetic environments, enabling exploration without the costs and risks associated with the physical world. Notably, although simulated data has been adopted for world models in synthetic environments [2, 10, 36, 37], such a source is overlooked in driving world models that operate in real scenarios [15, 14, 16, 38]. World models trained in simulation alone struggle to generalize to real world due to the significant visual gap between two worlds. Our data collection is conducted in CARLA with randomly sampled routes from Bench2Drive settings [39]. Two types of agents are deployed within the environments. One uses a well-established driving policy, PDM-Lite [40, 41], to collect expert executions, while the other adopts an exploration strategy whereby both the steering angle and the speed are randomly sampled from predefined sets to generate non-expert behavior data, which is underrepresented in human data yet a crucial component for training reliable world models. As a result, the numbers of video samples for each dataset are 4M for OpenDV, 85K for NAVSIM and 88K for CARLA. Each video sample is 4.9s long with a frequency of 10Hz, where the first 9 frames are visual context and the last 40 frames are prediction targets during training. See more data collection details in appendix B.1.

2.2 Controllable World Model

Basics. ReSim is built on CogVideoX [30], a high-capacity diffusion transformer originally conditioned only on text. To enable visual-context and trajectory conditioning, we replace the denoiser's historical latent inputs with their clean counterparts (following Vista [16]) and project future ego waypoints via a learnable encoder into the transformer's input space alongside video

latents. The model is supervised by the following video diffusion loss:

$$\mathcal{L}_{\text{diffusion}} = \mathbb{E}_{\boldsymbol{x},\epsilon,t}\Big[\|\boldsymbol{x}^{k:} - D_\theta(\boldsymbol{x_t}; t, \boldsymbol{h}, \boldsymbol{c}[, \boldsymbol{a}])^{k:}\|^2\Big], \tag{1}$$

where $\boldsymbol{x}$ is the clean video latent, and $\boldsymbol{x_t}$ is the noised video latent constructed by imposing a randomly sampled noise ϵ on $\boldsymbol{x}$ at a diffusion timestep t. The diffusion transformer D_θ is conditioned on latents of historical frames $\boldsymbol{h}$, a high-level text command $\boldsymbol{c}$ (*e.g.*, "Turn left"), and a fine-grained action $\boldsymbol{a}$ which is a sequence of future ego waypoints. To focus on forecasting the future, the diffusion loss is applied to the latent from the k-th frame onward only, excluding the observed history.

Dynamics Consistency Loss. So far, the standard video diffusion loss (Eq. (1)) supervises each video frame independently, which overlooks temporal correlations in videos, resulting in inferior spatiotemporal coherence and realism [42, 43, 16]. To address this, we introduce a dynamics consistency loss to additionally supervise the "latent motion", the discrepancy of video latent across different timestep ranges. This loss forces predicted motion to match the ground truth. We compute this loss over multiple intervals to capture both short-term and long-term dynamics. The intuition behind this is that some agent behaviors, *e.g.*, yielding, are hard to capture in the short term. To stabilize the magnitude of the loss value, we further normalize this loss by a factor of s, which is the average value of absolute motion disparity for each interval. This loss is formulated as:

$$\mathcal{L}_{\text{dynamics}} = \mathbb{E}_{\boldsymbol{x},\epsilon,t}\Big[\sum_{j=1}^{K}\sum_{i=1}^{N-j}\frac{1}{s}\|(\boldsymbol{d}^{i+j} - \boldsymbol{d}^{i}) - (\boldsymbol{x}^{i+j} - \boldsymbol{x}^{i})\|^2\Big], \tag{2}$$

where $\boldsymbol{x}$ and $\boldsymbol{d}$ are the ground-truth and model-predicted video latent respectively, and i indexes the frame of the video latent. K is the maximum timestep intervals considered for latent motion, which is set to 4 in our experiments. N is the number of frames of video latent. The total loss for training the world model is the combination of video diffusion loss and dynamics consistency loss: $\mathcal{L} = \mathcal{L}_{\text{diffusion}} + \lambda\mathcal{L}_{\text{dynamics}}$, where λ is set to 0.1 empirically. Note that both $\mathcal{L}_{\text{diffusion}}$ and $\mathcal{L}_{\text{dynamics}}$ are applied to the video latent compressed by the video VAE [30]. Therefore, the indices and number of frames in Eq. (1) and Eq. (2) correspond to the video latent representation instead of the raw video.

Unbalanced Noise Sampling. The behavior of a diffusion model is largely influenced by how we sample noise during training [44, 45], which controls how much noise is injected into the input data for the denoiser to recover [46, 47]. When applying commonly-used uniform noise sampling as in [30], we empirically find that our world model underperforms on complex driving dynamics, especially when we consider rare and non-expert behaviors. The issue behind this is that uniform timestep sampling lets models take a "shortcut" on low-noise diffusion timesteps where the model can recover the injected video noise by simply averaging information in adjacent frames, instead of learning critical motion details, which degrades the dynamics fidelity in generated driving videos [48]. To force the model to capture complex agent–environment interactions, we bias sampling toward higher-noise steps. We increase the frequency of drawing timesteps in [500, 1000] from $1/2$ to $2/3$, thereby amplifying input corruption and compelling richer dynamics learning.

Progressive Multi-stage Learning. We adapt CogVideoX [30], originally pretrained with text-only conditioning, into an controllable driving world model via a three-stage curriculum. **1)** We first endow it with the ability to predict futures that follow historical visual context and text commands, by training on OpenDV [14]. **2)** Next, we incorporate NAVSIM [23] and CARLA [29] with annotated actions for joint training with OpenDV. Action conditions, i.e., future ego waypoints, are encoded through a learnable transformer. Notably, NAVSIM trajectories are randomly masked ($p = 0.5$) to support both action-conditioned and free prediction, while CARLA waypoints remain intact to guide hazardous maneuvers that cannot be directly inferred from visual context. To prioritize structural dynamics over high-frequency details while improving training efficiency, we downsample inputs to 256×448, freeze the diffusion backbone, and fine-tune only the trajectory encoder and a LoRA adapter [49]. **3)** After the effective adaptation of action conditions, we finally resume the model training on 512×896 resolution with full fine-tuning, producing a model that generates 4s of 10Hz video conditioned on nine frames at 10Hz, an optional command, and a 4s, 2Hz waypoint sequence.

2.3 Reward Estimation from Video

To accomplish a feedback loop, world models need to estimate a reward to assess the predicted futures [1, 10], which is largely overlooked in prior driving world models [38, 14, 15]. Among the

few attempts, the lack of explicit goal states [19] in open-world driving and the complexity of outdoor scenarios make manual reward crafting challenging [31]. To overcome this, our key insight is to use the widely adopted simulator CARLA [29] as a rich source to learn rewards from via the unified video interface, as depicted in Fig. 2. Such a formulation offers several notable advantages. First, the driving simulator allows flexible exploration and can produce extensive data with environmental feedback to learn from. This includes not only successful driving experiences, but also non-expert mistakes and edge cases, covering a wide distribution of reward ranges. Second, contrary to constructing rewards manually with 3D perception models [31], the video interface does not require highly crafted 3D priors such as camera poses, and thus can benefit from a broad range of frontier vision models with strong cross-domain generalization [50, 51].

In detail, our Video2Reward model (V2R) is established on a frozen DINOv2 backbone [50] with an additional lightweight prediction head. Supervised by the CARLA infraction score [52, 29] that comprehensively penalizes multiple factors such as collisions and excessively low speeds, V2R learns to estimate the reward from video sequences. During inference, we send a planned trajectory produced by any policy to ReSim to simulate future video, which is then sent to V2R to estimate the reward of that trajectory. Due to the highly generalizable visual features of the DINOv2 backbone and the realistic prediction of ReSim, V2R is readily applicable to real-world driving scenarios, effectively assessing the quality of diverse behaviors.

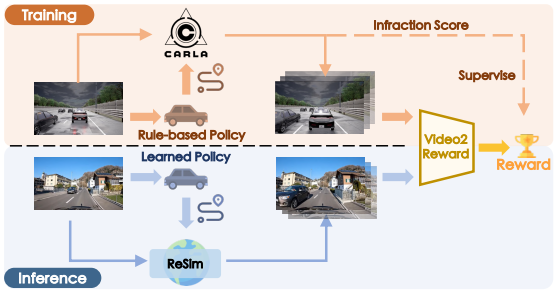

Figure 2: **Video2Reward model (V2R). Top**: V2R is supervised by infraction score of both safe and hazardous data from simulation, deriving the reward from a driving video. **Bottom**: In real-world inference, the predicted video of ReSim in reaction to a proposed action is fed into V2R to estimate the action's reward.

2.4 Applications

Video Prediction-based Policy. From the future prediction capability learned from massive human driving videos at scale, ReSim implicitly learns how the ego vehicle should behave and can be converted into a video prediction-based policy, akin to recent approaches in robotics [53, 3, 54]. As opposed to solely imitating the ego trajectory, predicting future observations allows for utilizing a broader source of unlabeled video data while leveraging richer supervision, including the intention of surrounding agents that are not captured in sparse trajectory-based outputs. To serve as a policy for deployment, ReSim takes historical visual observations and a high-level command as input and imagines the unseen future images, without conditioning on actions (which should be the output of this task). After visual imagination, the predicted future frames of ReSim are fed into an inverse dynamics model (IDM) that converts it into a future trajectory of the ego vehicle. Illustrative samples are shown in Fig. 3, where critical events for ego planning are highlighted in dashed boxes.

Figure 3: **Video prediction-based policy.** ReSim conditions on the history context (left) to synthesize a plausible visual plan (middle), which is then translated into an ego trajectory via an IDM (right).

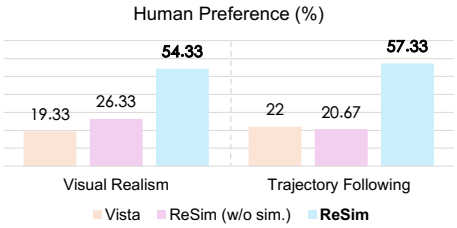

Figure 4: **Human evaluation of non-expert action controllability.** ReSim gets the most votes in both realism and trajectory following.

Table 1: **Action-conditioned prediction accuracy on Waymo (zero-shot).** ReSim surpasses baselines by a large margin under both action conditions. w/o sim.: No simulation data for training.

Method	Action-free ↓	Expert Action ↓
GT Future	0.58	0.58
Vista [16]	5.68	1.89
Ours w/o sim.	1.47	1.18
Ours	**1.13**	**0.86**

Reward-guided Policy Selection. Complex driving environments often necessitate the maintenance of multiple candidate proposals to ensure planning robustness across various scenarios [55]. While it is straightforward to obtain multiple trajectory proposals from different policies for the same scenario, it raises the question of how to reconcile these diverse outputs. To address this, we propose to apply our method to score each trajectory with a reward and select the one with the highest reward for execution. Concretely, each candidate trajectory is rendered into a short predictive video using ReSim, and the resulting video is then passed through Video2Reward model to obtain its reward. Guided by the estimated reward, the trajectory selection process results in a steered policy with significant improvement over individual policy candidates, by leveraging their advantages in different situations.

Closed-loop Visual Simulation. Vision-based driving agents are primarily evaluated in an open-loop manner, either on static datasets against pre-recorded trajectories [34, 35] or simulation-based benchmarks that consider local interactions [23]. Both these evaluation types confine agents to safe and human-driven scenarios. More seriously, they overlook error accumulations over extended rollouts and fail to reflect the closed-loop performance as in real-world driving, where agents would be continuously exposed to new states after taking actions. Owing to its precise action controllability and high visual fidelity, we can leverage ReSim to simulate visual states in a closed-loop manner. In each iteration, ReSim executes the predicted action of the driving agent to generate the next visual state, which is then input to the agent to make decisions for the next iteration.

3 Experiments

In this section, we first evaluate ReSim's simulation reliability, specifically relating to its action controllability, video prediction fidelity, and reasonableness of the reward formulation (Sec. 3.1). Next, we validate ReSim's applicability to real-world driving tasks (Sec. 3.2). Finally, we present ablation studies on data and methodological designs to verify their effectiveness (Sec. 3.3).

3.1 Results of Simulation Reliability

Results of Action Controllability. We verify the action controllability of ReSim on the unseen Waymo Open dataset [35]. For action-free and expert action conditioning, we follow the protocol of Vista [16] and use the *Trajectory Difference* metric to assess how closely the world model's predicted future aligns with the input trajectory. As reported in Tab. 1, ReSim improves the results by 80% and 54% for both conditioning modes compared to Vista. Moreover, removing the simulated data from training (ReSim w/o sim.) results in a performance decrease for both conditioning modes. This evaluation is conducted on a random subset of the Waymo validation set with 540 samples.

For non-expert action conditioning, we conduct a human preference study among samples generated by different methods conditioned on non-expert actions. As reported in Fig. 4, ReSim outperforms baselines by a large margin for both visual realism and trajectory following. We also make qualitative comparisons between different methods in Fig. 5, where ReSim yields more reliable and realistic results that align with the non-expert trajectory input. Moreover, the learned action controllability can be transferred to unseen datasets in a zero-shot manner, as showcased in Fig. 6.

Comparison of Video Prediction Fidelity. The fidelity of video prediction is a key indicator of a driving world model's ability to simulate realistic scenarios. As presented in Tab. 2, we evaluate the

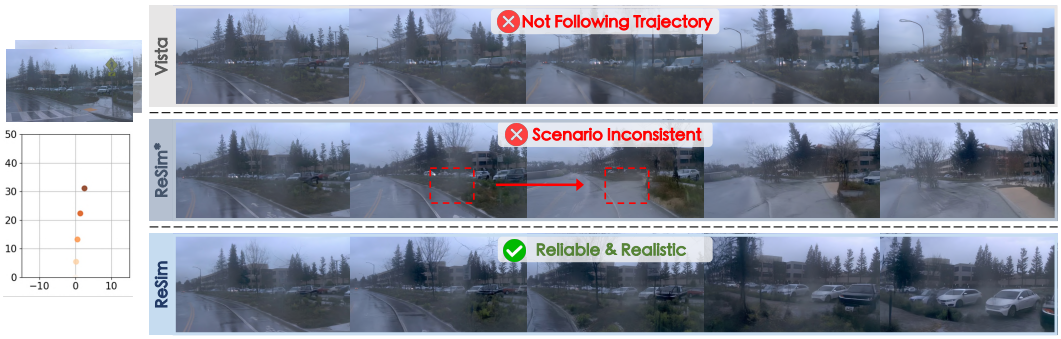

Figure 5: **Qualitative comparisons of non-expert action controllability.** ReSim reliably simulates hazardous outcomes from the non-expert action, while other methods either fail to follow the specified trajectory or compromise the scenario's consistency. *: without simulated data in training.

Figure 6: **Zero-shot action controllability.** ReSim can reliably follow both expert and non-expert actions in various scenarios from zero-shot datasets.

performance of various driving world models with FID [56] and FVD [57] metrics on nuScenes [34] validation set. The evaluation protocol follows Vista [16], using only context frames as conditions without imposing explicit action control. Notably, without training on any nuScenes data samples, ReSim yields significantly better results in a zero-shot manner compared to in-distribution models. We also provide qualitative comparisons for long-term future prediction in Appendix Sec. C, where Vista's prediction becomes over-saturated and loses semantics of the scene, while ReSim remains predicting visually rich future states in 30s.

Results of Reward Estimation. To evaluate the effectiveness of our reward formulations, we measure the ability of each reward model to distinguish "expert" from "non-expert" trajectories via a reward correlation metric. Specifically, for both CARLA [29] and NAVSIM [23], we randomly sample successful episodes with expert trajectories and accompany each with a randomly drawn trajectory from other samples that is potentially unsafe and assumed as non-expert. Evaluation is conducted with 250 pairs of com-

Table 2: **Comparison of prediction fidelity on nuScenes validation set without action condition**. Without seeing any nuScenes samples during training, ReSim outperforms previous in-distribution driving world models.

Method	Zero-shot	FID ↓	FVD ↓
DriveGAN [58]	✗	27.8	390.8
DriveDreamer [38]	✗	14.9	340.8
DriveDreamer-2 [17]	✗	25.0	105.1
WoVoGen [59]	✗	27.6	417.7
Drive-WM [31]	✗	15.8	122.7
GenAD [14]	✗	15.4	184.0
GEM [18]	✗	10.5	158.5
Vista [16]	✗	6.9	89.4
Ours	✓	**5.2**	**50.4**

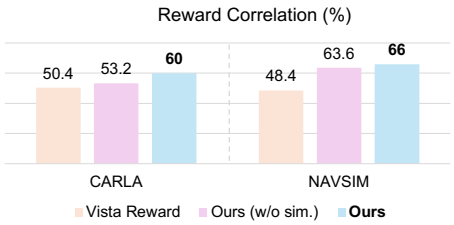

Figure 7: **Reward correlation.** Our method of composing ReSim and Video2Reward model yields more accurate rewards compared to baselines in both datasets.

Table 3: **Reward-guided policy selection.** Our reward formulation leads to a guided policy with +26% PDMS compared to candidate policies, outperforming other guidance and averaged ensemble.

Method	Steered	PDMS ↑
Transfuser [33]	✗	47.7
LTF [33]	✗	47.2
PDMS (Oracle)	✓	94.2
Average	✓	66.8
Vista [16]	✓	59.2
Ours w/o sim.	✓	69.7
Ours	✓	**74.1**

Table 4: **Planning results on NAVSIM** `navtest`**.** ReSim outperforms both world-model (WM)-based planners and end-to-end (E2E) methods by a large margin, without accessing extra information.

Method	Type	Multiple Sensors	Ego Status	Past Traj.	Extra Anno.	PDMS ↑
VO planner [60]	E2E Plan	✗	✗	✗	✗	78.4
UniAD [61]	E2E Plan	✓	✓	✗	✓	83.4
Transfuser [33]	E2E Plan	✓	✓	✗	✓	84.0
DrivingGPT [62]	WM + E2E Plan	✗	✗	✓	✗	82.4
LAW [32]	WM + E2E Plan	✓	✗	✗	✗	84.6
GT Future (Oracle)	Ground-truth + IDM	✗	✗	✗	✗	90.8
Ours	WM + IDM	✗	✗	✗	✗	**86.6**

parative samples for the reward model to judge. Reward models are expected to assign higher scores to expert trajectories compared to non-expert ones for the same scenario. Results in Fig. 7 validate the advantage of our method, which surpasses its counterparts in both simulated and real-world datasets.

3.2 Results of Applications

Video Prediction-based Policy. We evaluate the performance of our method on the `navtest` split of NAVSIM [23] benchmark. Specifically, we separately train an Inverse Dynamics Model (IDM) on the NAVSIM training set to convert the predicted video sequence of ReSim to an executable ego trajectory. As reported in Tab. 4, coupling ReSim and the lightweight IDM produces a video prediction-based policy that outperforms both end-to-end baselines (UniAD and Transfuser) and world model counterparts (DrivingGPT) by a non-trivial margin. Notably, our method only requires the history observations and a high-level command as input, without accessing multiple sensors, ego status, past trajectory, or extra annotations like other methods. The Visual Odometry (VO) planner shares the same architecture as our IDM yet performs poorly, underscoring ReSim's guidance.

Reward-guided Policy Selection. We compare different strategies for selecting an action from two candidate policies, *i.e.*, Transfuser and LTF [33]. The evaluation is conducted on a subset of NAVSIM, by selecting 300 challenging scenarios where one of the candidate policies fails while the other succeeds according to PDMS metric. As shown in Tab. 3, when applied separately, Transfuser and LTF achieve PDMS of 47.7 and 47.2, respectively. A uniform average ensemble lifts performance to 66.8, while the Vista reward only reaches 59.2. Instead, applying our reward strategy by composing ReSim and Video2Reward achieves a PDMS of 74.1, which is the closest score compared to the oracle selection according to ground-truth PDMS, and is higher than all baselines including our alternative (ours w/o sim.) that removes simulated data from the training of ReSim.

Closed-loop Visual Simulation. As showcased in Fig. 8, we leverage ReSim to iteratively simulate visual feedback for a running policy starting from two NAVSIM [23] scenarios. At each iteration, ReSim simulates an entire 4s future simultaneously by executing the action (*i.e.*, future trajectory for 4s) output by the policy. The newly generated frames are then fed into the policy for the subsequent

Figure 8: **Closed-loop visual simulation example.** A policy with front view only runs within the imaginary world generated by ReSim. The policy is adapted from XVO [60].

Figure 9: **Effect of unbalanced noise sampling.** Training with unbalanced noise sampling yields improved motion and scenario consistency.

Figure 10: **Effect of dynamics consistency loss (DCL).** Applying DCL with $K = 4$ (in Eq. (2)) works best.

decision. We opt for a lightweight Visual Odometry-based planner adopted from XVO [60] as the policy, since it only takes front-view video as input. Attributed to the generative rollout, ReSim position the policy into states that are never encountered in a pre-recorded dataset.

3.3 Ablation Study

Effect of Simulated Data. Throughout our experiments, we demonstrate that training with simulated data improves results across multiple tasks. For action controllability, removing CARLA simulation data leads to inferior results for both expert (Tab. 1) and non-expert actions (Fig. 4). Without simulated data, the synthesized future may be inconsistent in the scenario's structure when conditioned on non-expert actions, as showcased in Fig. 5. Simulated data also contributes to more accurate reward estimates as shown in Fig. 7, which further benefits reward-guided policy selection (Tab. 3).

Effect of Unbalanced Noise Sampling. As shown in Fig. 9, applying unbalanced noise sampling during training makes the predicted future more consistent in terms of agents' motion and scenario layout, compared to the baseline with uniform noise sampling.

Effect of Dynamics Consistency Loss. We visualize the effect of applying our proposed dynamic consistency loss in Fig. 10. The qualitative results verify that incorporating the loss and extending the maximum interval K for latent motion extraction (in Eq. (2)) yield more coherent future predictions.

4 Conclusion and Outlook

In this paper, we present ReSim, a reliable driving world model that excels in simulating a diverse range of actions in open-world scenarios. We incorporate non-expert data with hazardous actions from an established driving simulator to enrich real-world human driving data that primarily consists of safe behaviors. We also integrate several new training strategies, including a dynamics consistency loss, unbalanced noise sampling, and multi-stage learning. To facilitate driving applications beyond visual simulation, a Video2Reward model is devised to estimate the reward from the simulated future. Extensive experiments demonstrate the effectiveness and versatility of our ReSim system.

Limitation and Future Works. We envision our work as an early glimpse at open-world simulation with reward feedback, a cornerstone in establishing robust intelligence in the unstructured physical world. However, our system is still bottlenecked by inference efficiency due to iterative denoising, and

how to train agents within the synthesized world produced by ReSim is yet to be discovered. Future work focused on enhancing the efficiency, developing reinforced agents with the world model, and constructing fair closed-loop planning benchmarks would propel us closer to this goal. A discussion of limitations and broader impact of our work is included in Appendix Sec. D.

Acknowledgments

This study is supported by National Natural Science Foundation of China (62206172) and Shanghai Committee of Science and Technology (23YF1462000). It is also supported in part by Centre for Perceptual and Interactive Intelligence (CPII) Ltd. (a CUHK-led InnoCentre under the InnoHK initiative of the Innovation and Technology Commission of the Hong Kong Special Administrative Region Government.), National Natural Science Foundation of China (Grant No. 62306261), the Shun Hing Institute of Advanced Engineering (SHIAE, Grant No. 8115074), and the EXC (number 2064/1 – project number: 390727645). This work is also partially supported by Hong Kong RGC Strategic Topics Grant STG1/E-403/24-N, and CUHK-CUHK(SZ)-GDST Joint Collaboration Fund YSP26-4760949.

We also thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Kashyap Chitta. Our gratitude goes to Naiyan Wang, Shiyi Lan, Chonghao Sima, Haochen Tian, Yihang Qiu, Tianyu Li, Yunsong Zhou, and Qingwen Bu for valuable advice and discussions. We appreciate Huijie Wang's assistance in conducting the user study and constructing the project webpage.

References

[1] Yann LeCun. A path towards autonomous machine intelligence. *Open Review*, 62, 2022. 1, 2, 4, 24

[2] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *NeurIPS*, 2018. 1, 3, 24, 25, 28

[3] Sherry Yang, Jacob Walker, Jack Parker-Holder, Yilun Du, Jake Bruce, Andre Barreto, Pieter Abbeel, and Dale Schuurmans. Video as the new language for real-world decision making. In *ICML*, 2024. 1, 5, 24

[4] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, 2023. 1

[5] Penghao Wu, Li Chen, Hongyang Li, Xiaosong Jia, Junchi Yan, and Yu Qiao. Policy pre-training for autonomous driving via self-supervised geometric modeling. In *ICLR*, 2023. 1

[6] Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. In *ICML*, 2022. 1

[7] Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024. 1, 24

[8] Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *CVPR*, 2025. 1, 2, 24

[9] Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. In *NeurIPS*, 2024. 1, 24

[10] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to Control: Learning behaviors by latent imagination. In *ICLR*, 2020. 1, 2, 3, 4, 24, 28

[11] Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *ICLR*, 2024. 1, 24, 26

[12] Jing Yu Koh, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. Pathdreamer: A world model for indoor navigation. In *ICCV*, 2021. 1, 24

[13] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *ICLR*, 2024. 1, 24, 25

[14] Jiazhi Yang, Shenyuan Gao, Yihang Qiu, Li Chen, Tianyu Li, Bo Dai, Kashyap Chitta, Penghao Wu, Jia Zeng, Ping Luo, et al. Generalized predictive model for autonomous driving. In *CVPR*, 2024. 1, 3, 4, 7, 24, 25, 27, 28, 29

[15] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. GAIA-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023. 1, 3, 4, 24

[16] Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability. In *NeurIPS*, 2024. 1, 2, 3, 4, 6, 7, 8, 24, 27, 28, 29

[17] Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Xinze Chen, Guan Huang, Xiaoyi Bao, and Xingang Wang. DriveDreamer-2: LLM-enhanced world models for diverse driving video generation. *arXiv preprint arXiv:2403.06845*, 2024. 1, 7, 24

[18] Mariam Hassan, Sebastian Stapf, Ahmad Rahimi, Pedro Rezende, Yasaman Haghighi, David Brüggemann, Isinsu Katircioglu, Lin Zhang, Xiaoran Chen, Suman Saha, et al. GEM: A generalizable ego-vision multimodal world model for fine-grained ego-motion, object dynamics, and scene composition control. In *CVPR*, 2025. 1, 7, 24

[19] Stephen Tian, Chelsea Finn, and Jiajun Wu. A control-centric benchmark for video prediction. In *ICLR*, 2023. 1, 5

[20] Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. AdaWorld: Learning adaptable world models with latent actions. In *ICML*, 2025. 1

[21] Hidehisa Arai, Keishi Ishihara, Tsubasa Takahashi, and Yu Yamaguchi. ACT-Bench: Towards action controllable world models for autonomous driving. *arXiv preprint arXiv:2412.05337*, 2024. 1

[22] Zhiqi Li, Zhiding Yu, Shiyi Lan, Jiahan Li, Jan Kautz, Tong Lu, and Jose M Alvarez. Is ego status all you need for open-loop end-to-end autonomous driving? In *CVPR*, 2024. 1

[23] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, et al. NAVSIM: Data-driven non-reactive autonomous vehicle simulation and benchmarking. In *NeurIPS Datasets and Benchmarks*, 2024. 1, 2, 3, 4, 6, 7, 8, 25, 27, 29

[24] Dian Chen, Vladlen Koltun, and Philipp Krähenbühl. Learning to drive from a world on rails. In *ICCV*, 2021. 2

[25] Brian Tefft. Rates of motor vehicle crashes, injuries and deaths in relation to driver age, united states, 2014-2015. *AAA Foundation for Traffic Safety.*, 2017. 2

[26] Han Lu, Xiaosong Jia, Yichen Xie, Wenlong Liao, Xiaokang Yang, and Junchi Yan. ActiveAD: Planning-oriented active learning for end-to-end autonomous driving. *arXiv preprint arXiv:2403.02877*, 2024. 2

[27] Bingyi Kang, Yang Yue, Rui Lu, Zhijie Lin, Yang Zhao, Kaixin Wang, Gao Huang, and Jiashi Feng. How far is video generation from world model: A physical law perspective. *arXiv preprint arXiv:2411.02385*, 2024. 2

[28] Yunsong Zhou, Michael Simon, Zhenghao Mark Peng, Sicheng Mo, Hongzi Zhu, Minyi Guo, and Bolei Zhou. SimGen: Simulator-conditioned driving scene generation. In *NeurIPS*, 2024. 2

[29] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *CoRL*, 2017. 2, 3, 4, 5, 7, 29

[30] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. CogVideoX: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025. 2, 3, 4, 25, 26, 27, 29

[31] Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. Driving into the Future: Multiview visual forecasting and planning with world model for autonomous driving. In *CVPR*, 2024. 2, 5, 7, 24, 28

[32] Yingyan Li, Lue Fan, Jiawei He, Yuqi Wang, Yuntao Chen, Zhaoxiang Zhang, and Tieniu Tan. Enhancing end-to-end autonomous driving with latent world model. In *ICLR*, 2025. 2, 8

[33] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. TransFuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE TPAMI*, 2023. 2, 8

[34] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yuxin Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 3, 6, 7, 25, 29

[35] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020. 3, 6, 25, 27, 29

[36] Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *ICLR*, 2021. 3, 24

[37] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023. 3, 24

[38] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. DriveDreamer: Towards real-world-driven world models for autonomous driving. In *ECCV*, 2024. 3, 4, 7, 24

[39] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2Drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. In *NeurIPS Datasets and Benchmarks*, 2024. 3, 25, 29

[40] Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Jens Beißwenger, Ping Luo, Andreas Geiger, and Hongyang Li. DriveLM: Driving with graph visual question answering. In *ECCV*, 2024. 3

[41] Jens Beißwenger. PDM-Lite: A rule-based planner for carla leaderboard 2.0. https://github.com/OpenDriveLab/DriveLM/blob/DriveLM-CARLA/pdm_lite/docs/report.pdf, 2024. 3, 25

[42] Hyeonho Jeong, Chun-Hao Paul Huang, Jong Chul Ye, Niloy Mitra, and Duygu Ceylan. Track4Gen: Teaching video diffusion models to track points improves video generation. In *CVPR*, 2025. 4

[43] Shijie Wang, Samaneh Azadi, Rohit Girdhar, Saketh Rambhatla, Chen Sun, and Xi Yin. MotiF: Making text count in image animation with motion focal loss. In *CVPR*, 2025. 4

[44] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable Video Diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 4, 24, 25, 28

[45] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024. 4

[46] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 4

[47] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *NeurIPS*, 2022. 4

[48] Willi Menapace, Aliaksandr Siarohin, Ivan Skorokhodov, Ekaterina Deyneka, Tsai-Shien Chen, Anil Kag, Yuwei Fang, Aleksei Stoliar, Elisa Ricci, Jian Ren, et al. Snap Video: Scaled spatiotemporal transformers for text-to-video synthesis. In *CVPR*, 2024. 4

[49] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. 4

[50] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *TMLR*, 2024. 5, 26, 29

[51] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 5

[52] CARLA autonomous driving leaderboard. https://leaderboard.carla.org/, 2022. 5, 27

[53] Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *NeurIPS*, 2023. 5

[54] Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, et al. GR-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024. 5

[55] Haoran Song, Wenchao Ding, Yuxuan Chen, Shaojie Shen, Michael Yu Wang, and Qifeng Chen. PiP: Planning-informed trajectory prediction for autonomous driving. In *ECCV*, 2020. 6

[56] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017. 7

[57] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards Accurate Generative Models of Videos: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 7

[58] Seung Wook Kim, Jonah Philion, Antonio Torralba, and Sanja Fidler. DriveGAN: Towards a controllable high-quality neural simulation. In *CVPR*, 2021. 7, 24

[59] Jiachen Lu, Ze Huang, Jiahui Zhang, Zeyu Yang, and Li Zhang. WoVoGen: World volume-aware diffusion for controllable multi-camera driving scene generation. In *ECCV*, 2024. 7

[60] Lei Lai, Zhongkai Shangguan, Jimuyang Zhang, and Eshed Ohn-Bar. XVO: Generalized visual odometry via cross-modal self-training. In *ICCV*, 2023. 8, 9, 27, 29

[61] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, et al. Planning-oriented autonomous driving. In *CVPR*, 2023. 8, 27, 29

[62] Yuntao Chen, Yuqi Wang, and Zhaoxiang Zhang. DrivingGPT: Unifying driving world modeling and planning with multi-modal autoregressive transformers. *arXiv preprint arXiv:2412.18607*, 2024. 8

[63] Jialong Wu, Haoyu Ma, Chaoyi Deng, and Mingsheng Long. Pre-training contextualized world models with in-the-wild videos for reinforcement learning. In *NeurIPS*, 2023. 24

[64] Frederik Ebert, Chelsea Finn, Sudeep Dasari, Annie Xie, Alex Lee, and Sergey Levine. Visual Foresight: Model-based deep reinforcement learning for vision-based robotic control. *arXiv preprint arXiv:1812.00568*, 2018. 24

[65] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *ICRA*, 2017. 24

[66] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *ICML*, 2019. 24

[67] Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *ICML*, 2024. 24

[68] Russell Mendonca, Shikhar Bahl, and Deepak Pathak. Structured world models from human videos. In *RSS*, 2023. 24

[69] Yilun Du, Sherry Yang, Pete Florence, Fei Xia, Ayzaan Wahid, brian ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B. Tenenbaum, Leslie Pack Kaelbling, Andy Zeng, and Jonathan Tompson. Video language planning. In *ICLR*, 2024. 24

[70] Yutao Zhu, Xiaosong Jia, Xinyu Yang, and Junchi Yan. FlatFusion: Delving into details of sparse transformer-based camera-lidar fusion for autonomous driving. *arXiv preprint arXiv:2408.06832*, 2024. 24

[71] Cunxin Fan, Xiaosong Jia, Yihang Sun, Yixiao Wang, Jianglan Wei, Ziyang Gong, Xiangyu Zhao, Masayoshi Tomizuka, Xue Yang, Junchi Yan, et al. Interleave-VLA: Enhancing robot manipulation with interleaved image-text instructions. *arXiv preprint arXiv:2505.02152*, 2025. 24

[72] Dian Chen and Philipp Krähenbühl. Learning from all vehicles. In *CVPR*, 2022. 24

[73] Henry X Liu and Shuo Feng. Curse of rarity for autonomous vehicles. *Nature Communications*, 2024. 24

[74] Penghao Wu, Xiaosong Jia, Li Chen, Junchi Yan, Hongyang Li, and Yu Qiao. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. In *NeurIPS*, 2022. 24

[75] Xiaosong Jia, Penghao Wu, Li Chen, Jiangwei Xie, Conghui He, Junchi Yan, and Hongyang Li. Think Twice before Driving: Towards scalable decoders for end-to-end autonomous driving. In *CVPR*, 2023. 24

[76] Xiaosong Jia, Yulu Gao, Li Chen, Junchi Yan, Patrick Langechuan Liu, and Hongyang Li. DriveAdapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving. In *ICCV*, 2023. 24

[77] Xiaosong Jia, Junqi You, Zhiyuan Zhang, and Junchi Yan. DriveTransformer: Unified transformer for scalable end-to-end autonomous driving. In *ICLR*, 2025. 24

[78] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *JAIR*, 2013. 24

[79] Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, Nicolas Heess, and Yuval Tassa. Dm_control: Software and tasks for continuous control. *Software Impacts*, 2020. 24

[80] Michał Kempka, Marek Wydmuch, Grzegorz Runc, Jakub Toczek, and Wojciech Jaśkowski. ViZDoom: A doom-based ai research platform for visual reinforcement learning. In *CIG*, 2016. 24

[81] Xiaosong Jia, Liting Sun, Masayoshi Tomizuka, and Wei Zhan. IDE-Net: Interactive driving event and pattern extraction from human data. *RA-L*, 2021. 24

[82] Xiaosong Jia, Liting Sun, Hang Zhao, Masayoshi Tomizuka, and Wei Zhan. Multi-agent trajectory prediction by combining egocentric and allocentric views. In *CoRL*, 2022. 24

[83] Xiaosong Jia, Li Chen, Penghao Wu, Jia Zeng, Junchi Yan, Hongyang Li, and Yu Qiao. Towards capturing the temporal dynamics for trajectory prediction: a coarse-to-fine approach. In *CoRL*, 2023. 24

[84] Xiaosong Jia, Penghao Wu, Li Chen, Yu Liu, Hongyang Li, and Junchi Yan. HDGT: Heterogeneous driving graph transformer for multi-agent trajectory prediction via scene encoding. *IEEE TPAMI*, 2023. 24

[85] Xiaosong Jia, Shaoshuai Shi, Zijun Chen, Li Jiang, Wenlong Liao, Tao He, and Junchi Yan. AMP: Autoregressive motion prediction revisited with next token prediction for autonomous driving. *arXiv preprint arXiv:2403.13331*, 2024. 24

[86] Anthony Hu, Gianluca Corrado, Nicolas Griffiths, Zachary Murez, Corina Gurau, Hudson Yeo, Alex Kendall, Roberto Cipolla, and Jamie Shotton. Model-based imitation learning for urban driving. In *NeurIPS*, 2022. 24

[87] Qifeng Li, Xiaosong Jia, Shaobo Wang, and Junchi Yan. Think2Drive: Efficient Reinforcement Learning by Thinking in Latent World Model for Quasi-Realistic Autonomous Driving (in CARLA-v2). In *ECCV*, 2024. 24

[88] Zhenjie Yang, Xiaosong Jia, Qifeng Li, Xue Yang, Maoqing Yao, and Junchi Yan. Raw2Drive: Reinforcement learning with aligned world models for end-to-end autonomous driving (in carla v2). *arXiv preprint arXiv:2505.16394*, 2025. 24

[89] Yumeng Zhang, Shi Gong, Kaixin Xiong, Xiaoqing Ye, Xiao Tan, Fan Wang, Jizhou Huang, Hua Wu, and Haifeng Wang. BEVWorld: A multimodal world model for autonomous driving via unified bev latent space. *arXiv preprint arXiv:2407.05679*, 2024. 24

[90] Daniel Bogdoll, Yitian Yang, and J Marius Zöllner. MUVO: A multimodal generative world model for autonomous driving with geometric representations. *arXiv preprint arXiv:2311.11762*, 2023. 24

[91] Zetong Yang, Li Chen, Yanan Sun, and Hongyang Li. Visual point cloud forecasting enables scalable autonomous driving. In *CVPR*, 2024. 24

[92] Lunjun Zhang, Yuwen Xiong, Ze Yang, Sergio Casas, Rui Hu, and Raquel Urtasun. Learning unsupervised world models for autonomous driving via discrete diffusion. In *ICLR*, 2024. 24

[93] Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. OccWorld: Learning a 3D occupancy world model for autonomous driving. In *ECCV*, 2024. 24

[94] Sicheng Zuo, Wenzhao Zheng, Yuanhui Huang, Jie Zhou, and Jiwen Lu. GaussianWorld: Gaussian world model for streaming 3D occupancy prediction. *arXiv preprint arXiv:2412.10373*, 2024. 24

[95] Junqi You, Xiaosong Jia, Zhiyuan Zhang, Yutao Zhu, and Junchi Yan. Bench2Drive-R: Turning real world data into reactive closed-loop autonomous driving benchmark by generative model. *arXiv preprint arXiv:2412.09647*, 2024. 24

[96] Zhenjie Yang, Xiaosong Jia, Hongyang Li, and Junchi Yan. LLM4Drive: A survey of large language models for autonomous driving. *arXiv preprint arXiv:2311.01043*, 2023. 24

[97] Zhenjie Yang, Yilin Chai, Xiaosong Jia, Qifeng Li, Yuqian Shao, Xuekai Zhu, Haisheng Su, and Junchi Yan. DriveMoE: Mixture-of-experts for vision-language-action model in end-to-end autonomous driving. *arXiv preprint arXiv:2505.16278*, 2025. 24

[98] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 24, 25

[99] Yuqi Wang, Ke Cheng, Jiawei He, Qitai Wang, Hengchen Dai, Yuntao Chen, Fei Xia, and Zhaoxiang Zhang. DrivingDojo Dataset: Advancing interactive and knowledge-enriched driving world model. In *NeurIPS Datasets and Benchmarks*, 2024. 24, 25

[100] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. BEVFormer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *ECCV*, 2022. 24

[101] Bencheng Liao, Shaoyu Chen, Xinggang Wang, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Chang Huang. MapTR: Structured modeling and learning for online vectorized hd map construction. In *ICLR*, 2023. 24

[102] Xinzhu Ma, Wanli Ouyang, Andrea Simonelli, and Elisa Ricci. 3D object detection from images for autonomous driving: a survey. *IEEE TPAMI*, 2023. 24

[103] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024. 25

[104] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 25

[105] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. AnimateDiff: Animate your personalized text-to-image diffusion models without specific tuning. In *ICLR*, 2024. 25

[106] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent Video Diffusion Models for High-Fidelity Long Video Generation. *arXiv preprint arXiv:2211.13221*, 2022. 25

[107] Shoufa Chen, Mengmeng Xu, Jiawei Ren, Yuren Cong, Sen He, Yanping Xie, Animesh Sinha, Ping Luo, Tao Xiang, and Juan-Manuel Perez-Rua. GenTron: Diffusion transformers for image and video generation. In *CVPR*, 2024. 25

[108] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 25

[109] Itseez. Open source computer vision library. https://github.com/itseez/opencv, 2015. 25

[110] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020. 26

[111] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *JMLR*, 2022. 26

[112] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshops*, 2021. 26

[113] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. *arXiv preprint arXiv:1711.05101*, 2017. 27

[114] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. VideoGPT: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021. 27

[115] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 27

[116] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. ST-P3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *ECCV*, 2022. 27

[117] Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE TPAMI*, 2024. 28

[118] Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024. 28

[119] Yuanzhi Zhu, Hanshu Yan, Huan Yang, Kai Zhang, and Junnan Li. Accelerating video diffusion models via distribution matching. *arXiv preprint arXiv:2412.05899*, 2024. 28

[120] Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast causal video generators. *arXiv preprint arXiv:2412.07772*, 2024. 28

[121] Mitchell Goff, Greg Hogan, George Hotz, Armand du Parc Locmaria, Kacper Raczy, Harald Schäfer, Adeeb Shihadeh, Weixing Zhang, and Yassine Yousfi. Learning to drive from a world model. *arXiv preprint arXiv:2504.19077*, 2025. 28

[122] Marco Cusumano-Towner, David Hafner, Alex Hertzberg, Brody Huval, Aleksei Petrenko, Eugene Vinitsky, Erik Wijmans, Taylor Killian, Stuart Bowers, Ozan Sener, et al. Robust autonomy emerges from self-play. *arXiv preprint arXiv:2502.03349*, 2025. 29

[123] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. VAD: Vectorized scene representation for efficient autonomous driving. In *ICCV*, 2023. 29

[124] Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric M. Wolff, Alex H. Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. nuPlan: A closed-loop ml-based planning benchmark for autonomous vehicles. In *CVPR Workshops*, 2021. 29

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately summarize the contributions and reflect our results in Sec. 3.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Brief description is in Sec. 4 and a standalone limitation section is provided in Appendix D.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We put more implementation details in Appendix B. We will publicly release our code, model, and dataset.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We will release all code and models.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See experimental details in Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Our experiments are conducted at an extensive data scale on multiple benchmarks as specified in Sec. 3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See the implementation details in Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: This research conforms to the Code of Ethics in all aspects.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See broader impacts in Appendix D.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: See Appendix E.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: All assets, including code and models, will be released with well-organized documentation and instructions. We provide descriptions in Appendix E.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: See our setting of human evaluation in Sec. 3.1.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper conducts a human evaluation (survey) with human subjects and does not have potential risks incurred by participants.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

Appendix

In the Appendix, we first outline related works in Sec. A. We then demonstrate implementation details for data and models in Sec. B to supplement Sec. 2 in the main paper. Additional results are included in Sec. C to supplement Sec. 3 in the main paper. We discuss the limitations and broader impact of our work in Sec. D, and list the license of all assets in Sec. E.

A Related Work

A.1 World Model

World models are considered as the abstraction of the open world, and having this kind of common sense greatly helps to learn new skills effectively, thus leading to high-level intelligence [1]. Under the definition of world models following the Dreamer series [10, 36, 37], they represent the transition of environmental dynamics, taking the past states or observations and policy's actions as input, and generating the next (latent) state together with an estimation of the reward. They also feature long-term prediction with continuous rollouts [2].

Abundant literature has explored world models in traditional policy learning tasks, especially utilizing the look-ahead property to learn efficient representations [63], conduct sampling-based planning [64, 65, 66], and enable model-based reinforcement learning [2, 10, 36, 37].

Taking a step further, researchers in applications have successfully employed world models in simulated games [67, 9, 11, 10, 36, 37], navigation [8, 12], and robotics [13, 68, 69, 3, 70, 71]. However, to learn and apply a world model requires extensive exploration and interaction with the environment, leading to the above advancements mostly being developed in simulation or constrained environments. It is infeasible to obtain diverse hazardous driving movements in the real world [72, 73]. In this work, for the first time, we address this challenge by leveraging heterogeneous data and transferring rewards learned from simulation to diverse real-world scenarios.

A.2 Predictive Model for Driving Scenes

Driving Scenes are significantly unstructured, dynamic, and complex [74, 75, 76, 77], compared to standard policy learning environments such as Atari [78], DM Control [79], ViZDoom [80], *etc*. In order to effectively encode observations and facilitate restoring future environments, a wide span of representations have been explored to build the world state, including the vectorized representations [81, 82, 83, 84, 85], bird's-eye-view (BEV) representation [86, 87, 88, 89], point clouds [90, 91, 92], 3D occupancy [93, 94], images [58, 38, 17, 18, 95], and language [96, 97]. Meanwhile, these works mainly focus on public driving datasets, which are still limited in scales to achieve strong generalization ability. Inspired by the rapid growth of visual generative models [98, 44] and the increased data volume captured by cameras with low costs [15, 14, 99], recent world models that imagine future states in image sequence (*i.e.*, video) yield encouraging results in visual fidelity and generalization [15, 16, 18].

Unfortunately, prior methods still struggle to fulfill the mission of faithful simulation. Due to the insufficient learning of scenario dynamics, their imagination quality significantly degrades in challenging cases and long-horizon predictions [15, 16]. They also fall short in simulating negative consequences, such as car crashes, in response to bad ego actions, since they are mainly established on human driving logs, which are biased toward safe executions. Furthermore, the core problem for driving world models, how to deduce the reward for a given action and apply the world model for real-world driving problems, is largely understudied. In particular, with high-dimensional observations and complex relationships between agents and the environment, specifying rewards for open-world driving scenarios is challenging compared to goal-conditioned reward specifications [7, 65, 8]. Among the previous works, Wang *et al.* [31] propose to construct rule-based rewards with off-the-shelf 3D perception models [100, 101], yet these models are sensitive to sensor configurations like camera poses thus hard to generalize [102]. Uncertainty-based rewards in Vista [16] struggle to consider specific types of behaviors such as off-route actions. Our work meticulously investigates these challenges to facilitate planning and simulation.

A.3 Video Generation

In recent years, deep generative models have made remarkable strides in both image generation [103, 98] and video generation [104, 44, 105, 106]. Recent studies [30, 107] introduce the diffusion transformer architecture [108] to video generation and achieve impressive spatiotemporal consistency. However, existing video generation models trained with large-scale web data are not directly applicable as driving world models due to their imperfect prediction of driving scenarios and lack of action controllability [14]. We bridge the gap with novelly designed model structures and training protocols.

B Implementation Details

B.1 Dataset

Our guiding observation is that each data corpus has distinct characteristics and limitations in terms of scenario diversity, planning labels feasibility, and the degree of danger, as depicted in Fig. 1(a). Based on that, we propose compiling our training data from diverse sources to integrate their complementary features to cover a wide scope of scenarios and ego actions. We specify each type of data source as follows.

Universal Driving Videos. Building a world model that generalizes to arbitrary scenarios requires learning from massive data with a wide coverage [13, 14, 99]. Therefore, we leverage the OpenDV dataset [14], which is the largest public driving video dataset, to pillar the scenario generalization of our world model. OpenDV dataset includes 1700 hours of uncalibrated front-view driving videos captured worldwide with a wide coverage of scenarios and camera configurations. The uncalibrated nature of this dataset allows the learned model to seamlessly adapt to new camera settings. We pseudo-labeled the dataset with high-level driving commands, including "`Turning left`", "`Moving forward`", and "`Turning right`", by estimating the flow via the OpenCV toolkit [109]. During training, we assign a high sampling rate ($5\times$) to video sequences with turning actions based on the driving command, as these cases are generally more challenging to learn than the forward movement. As a result, we collect 4M video clips from OpenDV datasets.

Expert Driving Data. Despite the large data volume and high diversity of online driving videos, these videos do not provide detailed annotations for ego actions, *e.g.*, ego trajectories, which are critical for learning world models with required action conditions [2]. The absence of such action annotations calls for the need to incorporate expert driving datasets that are rigorously curated and labeled. Therefore, we include a public driving dataset NAVSIM [23] into our compilation. We intentionally exclude commonly used nuScenes [34] and Waymo [35] datasets from training, and leverage them for held-out evaluation. Specifically, 85K data samples from `navtrain` split of NAVSIM [23] are included in training.

Explorable simulated data. Both online driving videos and expert driving datasets are produced by human drivers. The lack of suboptimal data would hinder the world model's ability to emulate non-expert behaviors and corresponding outcomes, *e.g.*, collisions. We randomly sample from 220 predefined routes in the Bench2Drive benchmark [39], varying the weather and time of day to enhance scenario diversity. We deploy two agents to explore the simulated environment while collecting data: One uses a well-established driving policy, PDM-Lite [41], to collect data from successful executions. Another agent for collecting non-expert data is implemented by rule-based explorations to cover a larger action space. This agent randomly samples a control configuration for steering angle and throttle and a behavior pattern from a predetermined set to execute. The total number of successful and hazardous execution cases is 88K, with each type accounting for roughly half the amount.

To be more specific about the 'non-expert' agent, it follows a structured "expert-takeover" process. First, the expert PDM-Lite policy drives for the initial period (1s). Then, control is switched to one of the following exploratory strategies to generate diverse, non-expert actions for 4s: 1) *Slight Turns*: The vehicle steers slightly left or right towards a randomly chosen angle between 10-20 degrees and then continues forward. 2) *Hard Turns*: The vehicle steers slightly left or right towards a randomly chosen angle between 10-20 degrees and then continues forward. 3) *Forced Lane Changes*: The vehicle executes a hard lane change to the left or right. 4) *Tailgating*: The vehicle disables its brakes

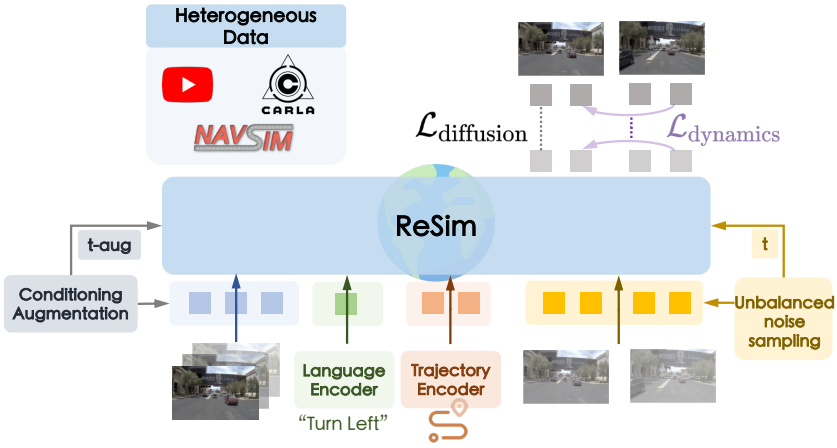

Figure S.11: **Overview of ReSim world model.** Learning from heterogeneous data compilation (Sec. 2.1), ReSim features designs specified in Main Sec. 2.2.

Table S.5: **Optimization configurations for different learning stages.** Traj. Enc.: Trajectory Encoder, Res.: Resolution, BS: Batch Size, LR: Learning Rate.

Stages	DiT	LoRA	Traj. Enc.	Dataset	Res.	BS	LR	Steps
1)	Trainable	-	-	OpenDV	512×896	80	$1e^{-5}$	20K
2)	Frozen	Trainable	Trainable	OpenDV, NAVSIM, CARLA	256×448	160	$5e^{-5}$	80K
3)	Trainable	Trainable	Trainable	OpenDV, NAVSIM, CARLA	512×896	80	$5e^{-5}$	50K

and applies full throttle to induce a rear-end collision with a vehicle ahead. Each strategy also includes randomization of its internal parameters (e.g., turning angle, speed) to encourage action diversity.

B.2 Model and Training

ReSim World Model. The architecture of ReSim is adapted from CogVideoX [30], consisting of a 2B diffusion transformer (DiT) as denoising backbone, a T5 encoder [110] for language encoding, a 3D Causal VAE that compresses raw videos into a compact latent space. Alongside language conditions for high-level driving command, we additionally devise a lightweight trajectory encoder, composed of two attention blocks and a linear head, to integrate the action condition into the DiT input. The overall architecture is depicted in Fig. S.11 with some of our designs highlighted. Besides our key innovations stated in Sec. 2, we also apply a conditioning augmentation strategy following [111, 11] to corrupt the video latent of historical observations to mitigate the error accumulation issue of long-term rollout. Similar to [11], the diffusion timesteps for historical context (t-aug) and future prediction (t) are separately sampled during training, while t-aug is always set to 0 during inference. This strategy improves the robustness of ReSim for multi-round prediction.

To enable classifier-free guidance for sampling [112], we randomly drop the textual command with a probability of $p = 0.5$. Similarly, we also drop the conditional ego trajectory at $p = 0.5$ for NAVSIM samples. However, we retain the ego trajectory for all CARLA samples without dropout, since their abnormal and hazardous behaviors cannot be accurately inferred from historical observations only and require explicit trajectory as guidance. Moreover, exposing the model to unconditioned hazardous behaviors could interfere with the learning of expert patterns from NAVSIM. Detailed learning configurations for different stages are included in Tab. S.5. All training stages are conducted on 40 A100 GPUs, and the total training duration is around 14 days.

Video2Reward Model. Video2Reward model consists of a pretrained DINOv2 [50] as backbone, and a prediction head that outputs a scalar reward. For each video sequence, all video frames are first processed separately via the image-based DINOv2 backbone. All image features are then passed to the prediction head, which aggregates all features via two consecutive spatial-temporal attention blocks and further predicts a scalar reward via an MLP.

Learning from our collected CARLA data only, Video2Reward model is supervised by the Infraction Score recorded from the CARLA simulator for each sample, which is a comprehensive evaluation of the ego driving performance [52] and penalizes behaviors such as collisions, traffic light violations, off-road deviations, and unreasonable low speed. It is trained for 20 epochs on a random subset of 35K samples from our CARLA data. We use the AdamW optimizer [113] with a learning rate of 1×10^{-3}. All video sequences are resized to 224×224 as input to this model.

Inverse Dynamics Model. Inverse dynamics model (IDM) estimates the ego trajectory from a video clip [114, 16]. Throughout our experiments, there are two parts that require the use of IDM, *i.e.*, the *Trajectory Difference* evaluation of expert action controllability (Sec. 3.1) and the application of video prediction-based policy (Sec. 3.2). These two IDMs are trained separately on different datasets, yet share the same architecture with a visual odometry backbone from XVO [60] and a lightweight attention head that outputs the ego trajectory with 8 waypoints in 2Hz.

For the *Trajectory Difference* of expert action controllability, the IDM transform model's action-control prediction into an estimated trajectory, and then we measure how closely the estimated trajectory matches the ground truth according to their L2 distance. A lower distance signifies a better action controllability of the driving world model. This IDM is trained on Waymo training set [35] for 40 epochs with a learning rate of 1×10^{-4}. For video prediction-based policy, the IDM transforms ReSim's action-free prediction (without command and ego future trajectory as condition) into an executable trajectory for planning. The IDM is trained on `navtrain` split of NAVSIM for 100 epochs. The learning rate for first 50 epochs is 1×10^{-4} and decreases to 1×10^{-5} for the last 50 epochs.

Visual Odometry(VO)-based Planner. The VO-based planner is utilized as a baseline for video prediction-based policy as in Tab. 4, and an agent that drives within the simulated world of ReSim for closed-loop visual simulation as in Fig. 8. It shares similar architecture and training to the aforementioned NAVSIM IDM. The only difference is that, instead of ingesting the whole video sequence containing both history and future frames as NAVSIM IDM, the VO-based planner takes historical frames as input only, without any explicit clue of the future observations.

B.3 Sampling

With ReSim, each short-term future video is simulated by sampling with the DDIM sampler [115] for 50 steps. The simulated outcome is a 4s video sequence in 10Hz with a resolution of 512×896. The input conditions include 9 frames of historical observations in 10Hz, an optional high-level command, and an optional ego trajectory with 8 future waypoints in 2Hz. The high-level command is in one of "`Turning left`", "`Moving forward`", and "`Turning right`", and is classified either by estimated flow for OpenDV dataset [14] or ego trajectory for action-annotated datasets like NAVSIM [23] following common practice in [61, 116]. We always apply a prefix prompt, "`This video depicts a realistic view from the driver's perspective of a car driving on the road.`", concatenated with the textual command for both training and sampling. Empirically, this prefix helps guide the model to generate driving scenarios. Following CogVideoX [30], we apply a decreasing classifier-free guidance strategy with guidance scale starting from 7.5 and gradually decreasing to 1. To synthesize a longer future beyond the training horizon (4s), we can leverage the last 9 frames from the newly generated sequence as the context for next-round prediction iteratively. Simulating a 4-second video sequence takes two minutes on a single Nvidia A100 GPU.

B.4 Human Evaluation

The human evaluation for non-expert action controllability (Sec. 3.1) is conducted with 15 participants and 40 questions for each participant, resulting in 600 answers in total. As showcased in Fig. S.12, each participant is requested to choose their preferred one among the synthesized video of three candidate models for each evaluation aspect. The candidate models are Vista [16], ReSim w/o simulated data, and ReSim (ours), and the evaluation aspects are Visual Realism and Trajectory Following. The association of different models and their generations is anonymous to participants.

Figure S.12: **Example of human evaluation.** Participants are presented with synthesized videos of three anonymous candidate models. The order of different models' generations is shuffled for each testing scenario.

C Additional Results

C.1 Action Controllability

We provide additional visualizations for zero-shot action controllability in Fig. S.13 and Fig. S.14 for nuScenes and Waymo samples, respectively. Both datasets are unseen during training. Qualitative results demonstrate that ReSim can be flexibly controlled by both ground-truth trajectory (expert action) and randomly associated trajectory (non-expert actions).

C.2 Ablation Study for Simulated Data

As shown in Fig. S.15, jointly training with simulated data improves the controllability of ReSim in open-world scenarios. Samples are from OpenDV validation set [14] with randomly associated trajectories from other labeled datasets.

C.3 Action-free Prediction

We show the action-free prediction ability of ReSim in Fig. S.16. When conditioned on historical frames only without action inputs, ReSim synthesizes a possible future outcome, that might differ from the ground-truth due to the multi-modality of driving scenarios [117].

C.4 Long-horizon Prediction

We compare ReSim with Vista [16] on long-horizon prediction in Fig. S.17. Starting from the same scenario, ReSim can emulate a more visually rich future in a longer horizon. This generation process does not use any action conditions, and both models perform multi-round rollouts that iteratively condition on the previously generated sequence to extend the prediction horizon.

C.5 Failure Mode

Although ReSim exhibits improved fidelity and controllability over previous methods, it still faces challenges as in Fig. S.18. We discuss the limitation in Sec. D (Societal Impact).

D Limitations and Broader Impact

Inference Efficiency. Despite the improved fidelity and controllability of our proposed ReSim, its real-world application is still potentially bottlenecked by the inference efficiency since diffusion models typically require multiple rounds of denoising process to ensure the generation quality [31, 44, 16]. To improve the inference latency, one potential solution is to reduce the number of denoising steps during the sampling phase. Recent advances in robotics [118] have proven that even with a single forward pass of the generative denoising network, the produced representation would greatly benefit downstream planning performance. Another approach is to distill a large yet slow diffusion model into a smaller one, which can be real-time deployed [119, 120].

World Model for Policy Training. Besides the onboard deployment of the heavy world model, another promising direction is to apply the world model as an dynamic environment to train policies [2, 10, 121]. This is beneficial as we can then deploy the policy to the autonomy directly, instead of

the world model, upon the training convergence of the policy model. Inspired by the tremendous success of large-scale policy learning within the abstract simulator without visual signals [122], the proposed ReSim offers a great opportunity to reproduce and go beyond the human-level robustness in the regime of vision-based driving [61, 123] by scaling up ReSim's visual simulation. We will follow this research direction in future work.

Closed-loop Benchmark. As illustrated in the results in Sec. 3.2, ReSim can reactively expose the policy to new states beyond the human driving logs when serving as a closed-loop visual simulator, in contrast to current predominant evaluation benchmarks for end-to-end autonomous driving [34, 35, 23]. However, since ReSim is trained on front-view observations only, common planning methods with multi-view camera inputs, such as UniAD [61] and VAD [123], cannot be readily applied in such simulation. Moreover, how to fairly benchmark different policies quantitatively using ReSim is still worth exploration.

Societal Impact. Though meticulously developed with state-of-the-art performance shown in the results, ReSim might still exhibit uncontrollable visual artifacts in generation due to the stochastic nature of the diffusion framework. It might also hallucinate in complex scenarios with multiple agents involved, and further pose risks for downstream applications. Despite the training on large-scale datasets, the uncurated data distribution, such as geographical regions, might lead to biased behavior of the learned model. We hope our work could shed light on the construction of open-world neural simulation for physical intelligence spanning both driving and robotics, by leveraging the visual richness of the physical world and the action flexibility of the simulated world collectively.

E License of Assets

Our training and evaluation are conducted on publicly licensed datasets and benchmarks [34, 124, 35, 23, 14]. To improve action diversity, we collected some data from the CARLA simulator [29] under the CC-BY License. The scenario configurations for the CARLA data follow Bench2Drive [39] under CC BY-NC-SA 4.0. ReSim is developed upon CogVideoX [30], with both code and model under the Apache License 2.0. We adopt public visual encoders, including DINOv2 [50] (under Apache License 2.0) and XVO [60] (under CC BY-NC-SA 4.0) for the construction of our Video2Reward and inverse dynamics model, respectively. Vista [16] is leveraged as a comparative baseline, which is under Apache License 2.0. We will release our code and models under the Apache License 2.0.

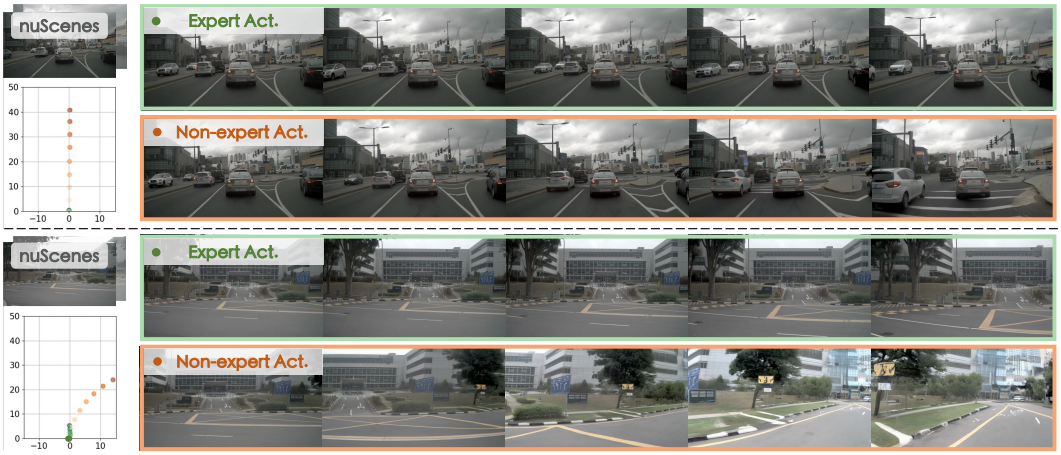

Figure S.13: **Visualizations for zero-shot action controllability on nuScenes.** The expert actions are recorded ground-truth from the driving log, while non-expert actions are randomly sampled from other scenarios. Best viewed zoomed in.

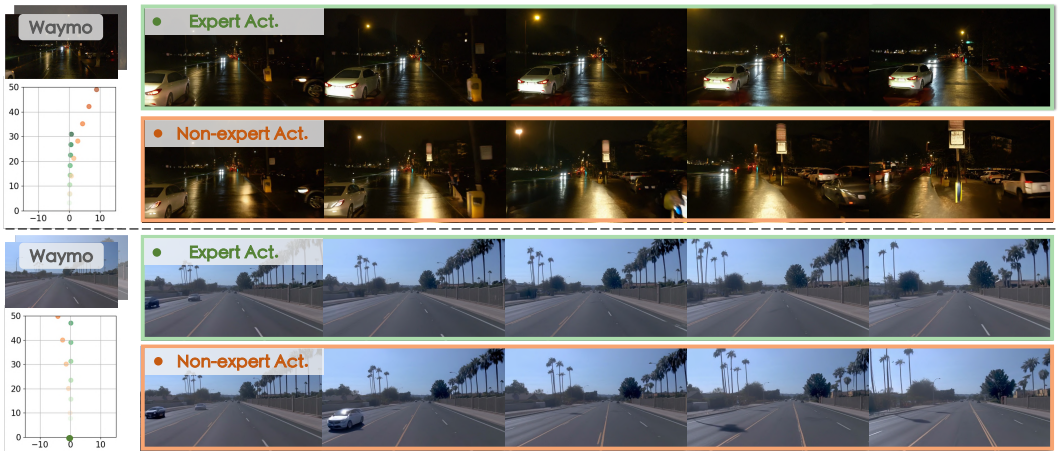

Figure S.14: **Visualizations for zero-shot action controllability on Waymo.** The expert actions are recorded ground-truth from the driving log, while non-expert actions are randomly sampled from other scenarios. Best viewed zoomed in.

Figure S.15: **Additional ablations for incorporating simulated data in training.** Simulated data improves controllability of ReSim for non-expert actions. Historical frames are not shown for brevity.

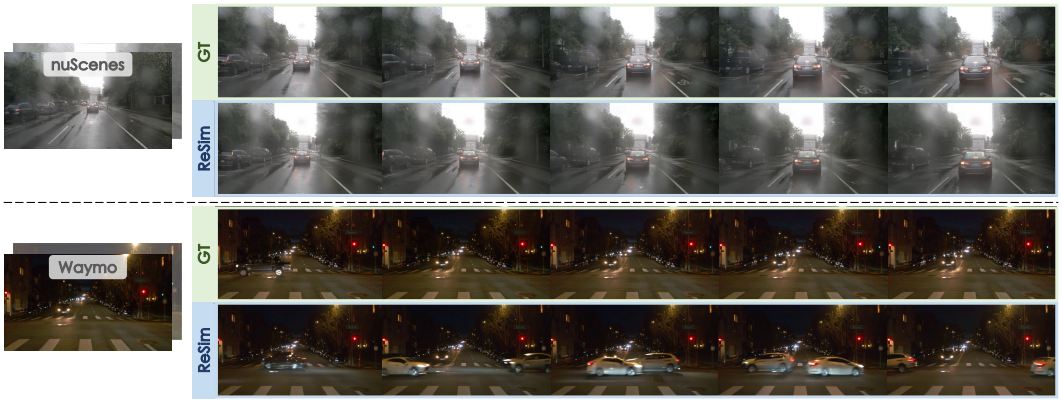

Figure S.16: **Visualizations for action-free future prediction.** ReSim can predict the future without action conditions by inferring from historical frames only.

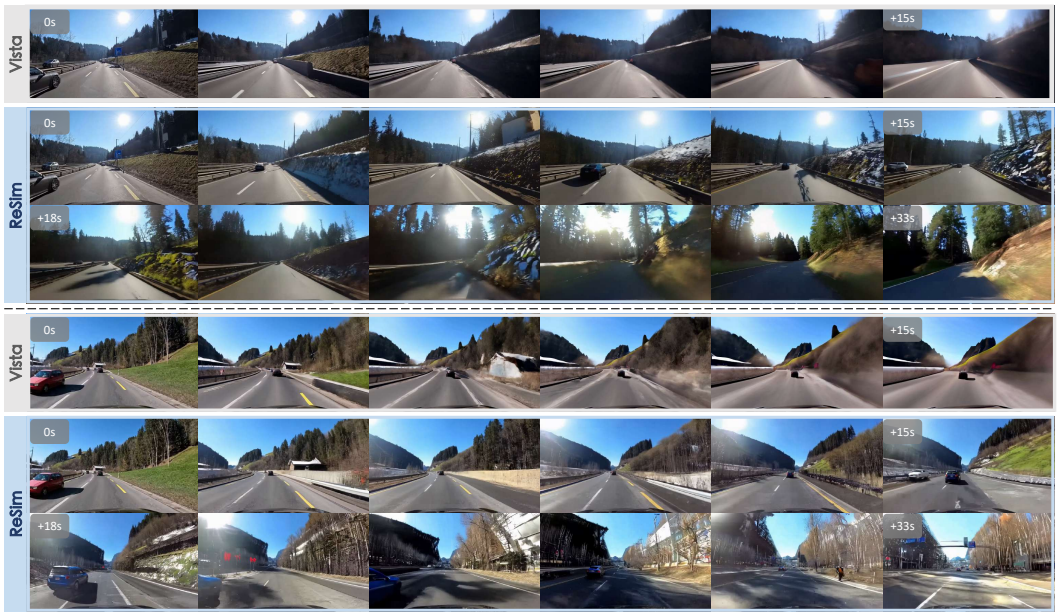

Figure S.17: **Long-term future prediction.** Compared to Vista whose prediction fidelity severely degrades in 15s, ReSim can predict consistent future states with rich details in more than 30s.

Figure S.18: **Failure modes.** ReSim still struggles in certain scenarios, such as falsely crossing the parapet, poor consistency for occluded objects, and producing visual artifacts for extreme cases. Best viewed zoomed in.