Position: Video as the New Language for Real-World Decision Making

Sherry Yang [1 2] **Jacob Walker** [1] **Jack Parker-Holder** [1] **Yilun Du** [3] **Jake Bruce** [1] **Andre Barreto** [1]
Pieter Abbeel [2] **Dale Schuurmans** [1]

Abstract

Both text and video data are abundant on the internet and support large-scale self-supervised learning through next token or frame prediction. However, they have not been equally leveraged: language models have had significant real-world impact, whereas video generation has remained largely limited to media entertainment. Yet video data captures important information about the physical world that is difficult to express in language. To address this gap, we discuss an under-appreciated opportunity to extend video generation to solve tasks in the real world. We observe how, akin to language, video can serve as a unified interface that can absorb internet knowledge and represent diverse tasks. Moreover, we demonstrate how, like language models, video generation can serve as planners, agents, compute engines, and environment simulators through techniques such as in-context learning, planning and reinforcement learning. We identify major impact opportunities in domains such as robotics, self-driving, and science, supported by recent work that demonstrates how such advanced capabilities in video generation are plausibly within reach. Lastly, we identify key challenges in video generation that mitigate progress. Addressing these challenges will enable video generation models to demonstrate unique value alongside language models in a wider array of AI applications.

1 Introduction

There has been tremendous progress in training large language models (LLMs) from internet text datasets in the past few years (Team et al., 2023; Achiam et al., 2023). The impressive performance of LLMs on a wide variety of tasks makes it tempting to reduce the artificial intelligence agenda to scaling up these systems. However, this is not sufficient. Firstly, the quantity of publicly available text data is becoming a bottleneck to further scaling (Villalobos et al., 2022). Secondly, and perhaps more importantly, natural language alone might not be enough to describe all intelligent behavior (Searle, 1980; Dennett, 1993; Minsky, 1988) or capture all information about the physical world we live in (e.g., imagine teaching someone how to tie a knot using words only). While language is a powerful tool to describe higher-level abstractions, it is not always sufficient to capture the physical world in all its wealth of detail.

Thankfully, there are abundant video data on the internet (e.g., over ten thousand years of consecutive video watching from YouTube alone) encapsulating a wealth of information imbued with knowledge of the world. Nevertheless, today's machine learning models trained on internet text or video data have demonstrated remarkably different capabilities. LLMs have advanced to tackling intricate tasks that require sophisticated reasoning (Huang & Chang, 2022), tool use (Mialon et al., 2023), and decision making (Yang et al., 2023c). In contrast, video generation models have been less explored, primarily focusing on creating entertainment videos for human consumption (Ho et al., 2022a; Singer et al., 2022; Bar-Tal et al., 2024). Given the paradigm shift unfolding in language modeling, it is important to ask whether we can elevate video generation models to the level of autonomous agents, simulation environments, and computational engines similar to language models so that applications requiring visual modalities such as robotics, self-driving, and science can more directly benefit from internet visual knowledge and pretrained video models.

In this paper, we take the position that *video generation will be to the physical world as language modeling is to the digital world.* To arrive at this position, we first identify key components that have enabled language models to solve many real-world tasks: (1) a *unified representation* (i.e., text) that can absorb broad information from the internet, (2) a *unified interface* (i.e., text generation) through which diverse tasks can be expressed as generative modeling, and (3) language models' ability to *interact* with external environments (e.g., humans, tools, and other models) by taking actions and optimizing decisions based on external feedback through techniques such as reinforcement learning from human feedback (Ouyang et al., 2022), planning (Huang et al., 2022), search (Yao et al., 2023), and optimization (Rafailov

[1]Google DeepMind [2]UC Berkeley [3]MIT. Correspondence to: Sherry Yang <sherryy@{google.com}{berkeley.edu}>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

et al., 2023).

Motivated by these three aspects of language models, we observe that (1) video can serve as a unified representation to absorb broad information about the physical world, (2) diverse tasks in computer vision, embodied AI, and science can be expressed or supported by a video generation model, and (3) video generation as a pretraining objective introduces internet-scale supervision for large vision models, behavior models, and world models, which in tern enables actions to be extracted, environment interactions to be simulated, and decisions to be optimized.

To further illustrate how video generation can have a profound impact on real-world applications, we provide an in depth analysis on recent work that utilizes video generation as task solvers, answers to questions, policies/agents, and environment simulators through techniques such as instruction tuning, in-context learning, planning, and reinforcement learning (RL) in settings such as games, robotics, self-driving, and science. Lastly, we identify major difficulties around video generation, and suggest plausible solutions to address these challenges to unleash the full potential of video generation in the real world.

2 Preliminaries

We provide a brief overview of video generation models and how they have been used in domain-specific settings through conditional generation.

2.1 Conditional Video Generation

We denote a video clip as a sequence of image frames $\mathbf{x} = (x_0, ..., x_t)$. An image on its own can be treated as a special video with a single frame, $\mathbf{x} = (x_0,)$. Conditional video generation models the conditional probability $p(\mathbf{x}|c)$ where c is the conditioning variable. The conditional probability $p(\mathbf{x}|c)$ has commonly been factorized by an autoregressive model (Razavi et al., 2019), a diffusion model (Ho et al., 2022a), or a masked transformer model (Chang et al., 2022). Depending on the factorization, sampling from $p(\mathbf{x}|c)$ corresponds to either predicting images (patches) sequentially or predicting all frames $(x_0, ..., x_t)$ together, iteratively.

2.2 Task-Specific Specialization

Depending on what is in the conditioning variable c, conditional video generation can serve different purposes. Below, we enumerate common examples of c and their use cases.

- $p(\mathbf{x}|c = \text{text})$. This corresponds to text-to-video models commonly used for generative media (Kondratyuk et al., 2023; Blattmann et al., 2023b), where the text is often some creative description of the desired video (e.g., "A teddy bear painting a portrait" in Singer et al. (2022)). Text-to-video has mostly been applied to generating movies (Zhu et al., 2023) and animations (He et al., 2023; Guo et al., 2023).
- $p(\mathbf{x}|c = \{x_0, \text{text}\})$. This corresponds to generating video rollouts starting from a given image x_0 while incorporating the text description. This type of conditioning has been applied to generate scene-specific visual interactions (Yang et al., 2023b) and visual plans for robot executions (Du et al., 2023b). When $\mathbf{x}$ only contains a future image x_t, $p(x_t|c = \{x_0, \text{text}\})$ can predict visual goals for robot manipulation (Black et al., 2023; Yu et al., 2023). This approach to goal synthesis is largely inspired by the vast literature on stylized image generation and inpainting (Efros & Freeman, 2023; Wang et al., 2023a).
- $p(\mathbf{x}|c = \{\bar{\mathbf{x}}, \text{text}\})$. When $\mathbf{x}$ and $\bar{\mathbf{x}}$ have the same underlying content, this corresponds to text-guided video editing and stylization (Loeschcke et al., 2022; Yang et al., 2023a), which has been applied to generate self-driving videos in different weather conditions (Hu et al., 2023). Note that $\bar{\mathbf{x}}$ can also be completely different from $\mathbf{x}$, in which case $\bar{\mathbf{x}}$ can serve as a visual prompt to elicit certain patterns in the output videos (Bai et al., 2023).
- $p(x_{i+1}|c = \{x_i, \text{action}\})$. This corresponds to learning a visual dynamics model where the action can be robot controls (Yang et al., 2023b), keyboard inputs (Hafner et al., 2020), or other motion information (Li et al., 2023) that causes change in the visual space. If we replace x_{i+1} with x_{i+t} for some $t > 1$, we have a temporally-abstract dynamics model (Sutton et al., 1999). In this case we can also replace x_i with any sub-sequence of $(x_i, x_{i+1}, ..., x_{i+t-1})$.

These specializations of conditional video generation suggest that there may exist a general framework under which broad video data can be absorbed and diverse tasks can be expressed using video generation.

3 Unified Representation and Task Interface

In this section, we first describe how video is a *unified representation* that can capture various types of information from the internet to form broad knowledge. We then discuss how diverse tasks from computer vision and embodied AI can be formulated as a conditional video generation problem, providing the foundation for real-world decision making with video generation. Details of the models used to generate the examples can be found in Appendix A. Additional generated videos can be found in Appendix B.

3.1 Video as a Unified Representation of Information

While internet text data has provided much value to the digital/intellectual world with large language models, text is more suitable for capturing high-level abstractions as opposed to low-level details of the physical world. Below, we list a few types of information that is hard to express as text but can be easily captured by video.

- **Visual and Spatial Information**: This includes visual details such as colors, shapes, textures, lighting effects, and spacial details such as how objects are arranged in space, their relative positions, distances, orientations, and 3D information. Such information naturally exist in image/video format as opposed to text format.

- **Physics and Dynamics**: This includes details about how objects and environments interact with each other physically, such as collisions, manipulations, and other movements influenced by physical laws. While text can describe movements at a high-level (e.g., "a car driving down the street"), it is often insufficient to capture low-level details such as the torque and friction applied to the vehicle. Videos can implicitly capture this information.
- **Behavior and Action Information**: This includes information such as human behaviors and agent actions, characterizing the low-level details of performing tasks such as how to assemble a piece of furniture. Text again can mostly capture high-level descriptions of how to perform a task as opposed to the detailed information such as precise motions and movements.

Why Video? One may wonder that, even if text is not sufficient to capture the above information, why video? To answer this question, we observe that video, in addition to existing at an internet scale, is interpretable to humans (similar to text) so that debugging, interaction, and safety speculation can be easily conducted. Moreover, video is a flexible representation that can characterize information at different spacial and temporal resolutions, e.g., atoms moving at angstrom scale (10^{-10}m) (Kashin et al., 2021) and light traveling at a trillion frames per second (Faccio & Velten, 2018).

3.2 Video Generation as a Unified Task Interface

In addition to a unified representation that can absorb broad information, we have seen from language modeling that we need a unified task interface through which diverse tasks can be expressed using a single objective (e.g., next token prediction); also, it is the alignment between representation of information (e.g., text) and task interface (e.g., text generation) that enables transfer of broad knowledge to task-specific decisions. In this section, we show how diverse vision tasks, as well as a broader set of question answering, reasoning, and problem solving, can all be expressed as a video generation task.

Classical Computer Vision Tasks. In natural language processing, many tasks (e.g., machine translation, text summarization, question answering, sentiment analysis, named entity recognition, part-of-speech tagging, text classification, dialogue systems) have traditionally been considered as different tasks but now have all been unified under the umbrella of language modeling. This has allowed greater generalization and knowledge sharing across tasks. Similarly, computer vision also has a broad set of tasks spanning across semantic segmentation, depth estimation, surface normal estimation, pose estimation, edge detection, and object tracking. Recent work has shown that it is possible to convert diverse vision tasks into a video generation task as shown in Figure 1 (Bai et al., 2023; Bar et al., 2022; Wang et al., 2023b), and that this unified approach to solving vision tasks scale favorably with model size, data size, and context length (Bai et al., 2023).

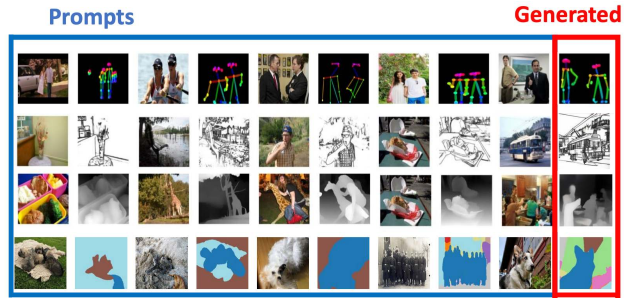

Figure 1: **Vision Tasks as Video Generation.** Figure 8 from Bai et al. (2023) (simplified to show partial prompts), where diverse computer vision tasks such as joint/edge detection, depth estimation, and segmentation can be converted into a single next-frame prediction task.

Converting vision tasks into a video generation task generally involves the following steps: (1) structure the input and output of a task (e.g., segmentation maps, depth maps) into a unified image/video space, (2) reorder image frames so that an input image is followed by the expected output image of a specific task (e.g., a regular input image followed by a depth map), and (3) leverage in-context learning by providing example input-output pairs as input to the conditional video generation model to specify the desired task.

Video as Answers. In traditional visual question answering (VQA) (Antol et al., 2015), the expected answers are in text. With the development in video generation, a novel task would be to treat video as answers, e.g., a video would be generated in response to "how to make an origami airplane" (Souček et al., 2023; Yang et al., 2023b). Similar to how language models can generate customized response to human inquiries in text, video models can also generation customized answers to how-to questions with great low-level details. Such video response can be more preferable to humans than textual response (Yadav et al., 2011). In Figure 2, we illustrate videos generated by a text-to-video model in response to a set of how-to inquiries. Additionally, one may consider conditioning generation on an initial frame to synthesize video answers in user-specific scenes. Despite such a grand promise, videos synthesized by today's text-to-video models are generally too short/simple, not containing enough information to fully answer users' questions.

The problem of synthesizing video frames to answer users' questions has similarities to planning with language models (Valmeekam et al., 2023), except that both the state and low-level action spaces are now pixels as opposed to text.

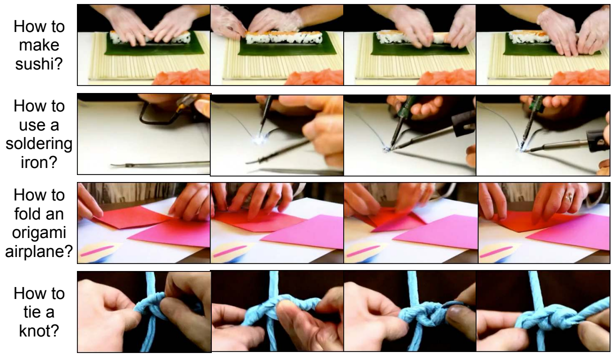

Figure 2: **Generated How-to Videos.** Video generation models can synthesize key frames corresponding to human hands performing intricate tasks. However, the generated frames are too generic and do not capture enough details to fully answer users' questions.

One may utilize language models or vision language models to break down high-level goals (e.g., "how to make sushi") into specific subgoals (e.g., "first, put rice on rolling mat") and synthesize plans for each subgoal while validating the plausibility of synthesized plans (Du et al., 2023c).

Visual Reasoning and Chain-of-Thought. With a unified representation of information and a unified task interface, reasoning has emerged in language modeling where a model can elicit relevant information as intermediate steps towards solving more complex problems (Wei et al., 2022). Similarly, with video as a unified representation and task interface, video generation has also exhibited early signs of visual reasoning by predicting masked regions of an image, as shown in Figure 3 (Bai et al., 2023). It would be interesting to see if next frame prediction can be used to solve more complex geometry problems similar to Trinh et al. (2024) by generating videos with the right set of auxiliary lines.

Building on the idea of leveraging next-frame prediction for visual reasoning and solving geometry problems, we can further characterize the reasoning *process* (Himakunthala et al., 2023) and algorithms (Yang et al., 2022b) using videos. Specifically, Yang et al. (2022b) characterized the execution state of a Breadth First Search (BFS) algorithm using videos. In this context, learning to generate a video corresponds to learning to search, as illustrated in Figure 4 (see also Silver et al. (2017)). While the examples in Figure 3 and Figure 4 might seem contrived, they serve as early indicators that video generation as a pretraining task may elicit reasoning-like behaviors similar to language models, revealing opportunities in leveraging video generation to solve complex reasoning and algorithmic tasks.

Figure 3: **Visual Reasoning as Next–Frame Generation.** Figure 13 from Bai et al. (2023) shows next-frame prediction can solve visual reasoning tasks such those in IQ tests.

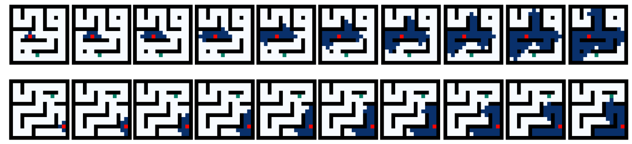

Figure 4: **BFS as Video Generation.** Figure 14 from Yang et al. (2022b) shows two sets of intermediate frames generated by a video model emulating the BFS search procedure. The red and green cells represent the start and goal locations. The white and black cells represent empty spaces and obstacles. Blue cells represent the cells that would have been visited by running the BFS algorithm.

3.3 Video as a Unified State-Action Space

We have seen that video generation can absorb broad knowledge and characterize diverse vision tasks. In this section, we further support this observation by providing concrete examples in embodied AI of using video as a unified representation and task interface.

One of the long-standing challenges in embodied AI has been data fragmentation, where datasets collected by one robot performing one set of tasks is hardly useful for learning on a different robot or on a different set of tasks (Padalkar et al., 2023). The major difficulty in knowledge sharing across robots and tasks lies in that each type of robot and task has distinct state-action spaces. To address this difficulty, Du et al. (2023b) advocate the use of the pixel space as a unified state-action space across tasks and environments. Under this framework, embodied planning can be cast as a conditional video generation problem, thereby benefiting from internet pretrained video generation models. An additional module such as an inverse dynamics model (Du et al., 2023b), a goal-conditioned policy (Black et al., 2023; Kang et al., 2023; Du et al., 2023c), an optical flow network (Ko et al., 2023), or dense grid point (Wen et al., 2023) can then be employed to recover the low-level robot controls from high-level video plans. We illustrate video plans generated by previous work in Figure 5 (top). Most existing work trains one video generation model per robot, which diminishes the potential benefit of using video as a unified state-action space for embodied learning. We provide additional generated video plans from training a video generation model on the Open X-Embodiment dataset (Padalkar et al., 2023) with a diverse set of robots and tasks in Figure 5 (bottom). Both the previous and newly generated video plans look highly realistic and successfully complete the specified tasks. Note that video as a unified interface can also implicitly align multiple viewpoints (robot wrist view in Figure 5) through action-conditioned video generation. Given the initial frame (wrist-view or third-person) and a sequence of actions, the generated wrist view video is encouraged to be consistent with the generated third-person view conditioned on the same sequence of actions.

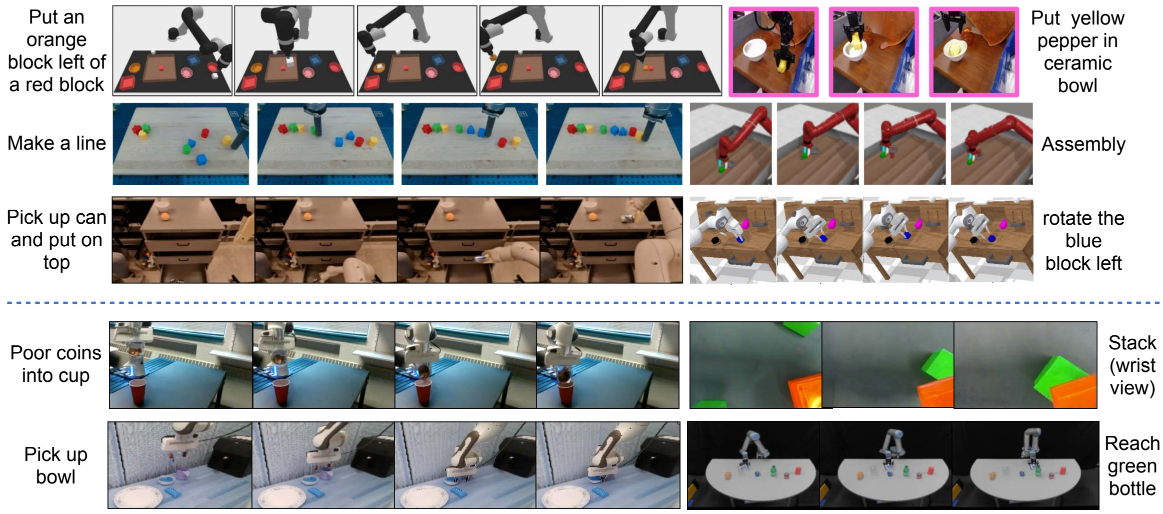

Figure 5: **Generated Video Plans for Robots.** [Top] Video plans generated by existing work (Figure 3 in Du et al. (2023b), Figure 3 in Black et al. (2023), Figure 3 in Du et al. (2023c), Figure 5 in Ko et al. (2023), Figure 14 in Yang et al. (2023b), Figure 7 in Kang et al. (2023)). [Bottom] Video plans generated by a single video generation model trained on the Open X-Embodiment (Padalkar et al., 2023).

4 Video Generation as Simulation

While video generation on its own can already solve many tasks as described in the previous section, another important opportunity in video generation is to simulate visual observations of various systems and processes, so that control inputs to a system can be optimized according to simulation results. This is especially useful for applications where abundant video data can be collected but the underlying dynamics are hard to be explicitly expressed (e.g., cloud movement, interaction with soft objects). In this section, we begin by studying such visual generative simulators in game settings, where we may have ground truth game engines to verify qualities of learned simulators and iterate on effective generation of new experiences. We then provide examples of simulating real-world processes such as robot interactions, autonomous driving, and atomic-level interactions. Details of the generative models used to generate the examples can be found in Appendix A. Additional generative simulation results can be found in Appendix B.

4.1 Generative Game Environments

Games have been used as a testbed for AI algorithms for decades (Yannakakis & Togelius, 2018). For instance, the Arcade Learning Environment (Bellemare et al., 2013) enabled the development of deep Q-learning, the first AI agent to reach human level in playing Atari games (Mnih et al., 2015). In a similar vein, we can consider games as a means to test the quality of generative simulators by comparing against ground truth simulations from the game engine. In the future, we may even be able to use generative models to surpass what is possible with existing human-designed simulated environments. In this section we discuss these possibilities, ranging from simulating a single complex environment to generating entirely new ones.

Figure 6: **Generated Game Trajectories in Minecraft.** Both actions and observations are generated using an autoregressive model trained on Minecraft data. In the top row, the inventory is opened. The middle row shows use of a pickaxe to break a stone block. The bottom row predicts movement throughout the environment.

Video Models for Planning. Action-conditioned video generation can possibly simulate the dynamics of complex environments. As a proof of concept, we trained a transformer-based architecture, autoregressive in time, that predicts future agent actions and observations in the Minecraft video game. The predictions are conditioned on the episode history, that is, the sequence of observations and actions up to a point in time. We used the "contractor data" from Baker et al. (2022), which consists of trajectories collected while humans interacted with the game. Both observations and actions are quantized tokens, reducing model-based rollout to next token prediction. Note that in this case the model serves both as a world model and a policy: given a sequence of alternating observations and actions ending in an action, the model can infer the next observation (world model), and given an analogous sequence ending in an observation, the model can infer the next action to take

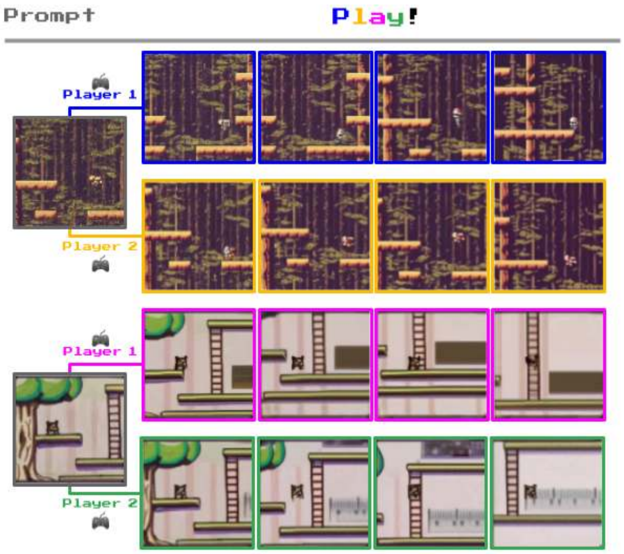

Figure 7: **Generated Interactive Game Environments**: Two synthetic image prompts passed to the model from (Bruce et al., 2024), which converts them into interactive environments. From there, it is possible to generate diverse trajectories by taking different latent actions, shown here as Player 1 and 2.

(policy). Figure 6 shows a few generated trajectories from this model. The model is capable of generating actions and transitions corresponding to sophisticated strategies (e.g., using a pickaxe to break a stone block).

With such a policy and dynamics backbone, model-based reinforcement learning algorithms—such as Dyna (Sutton, 1991), Dreamer (Hafner et al., 2020), and MuZero (Schrittwieser et al., 2019; Antonoglou et al., 2022)—could be employed to improve the policy. When used for planning in reinforcement learning, a world model should only capture the information from the video—or, more generally, from the stream of observations—that influences the reward. If details like the background, texture, and colors do not have any impact on rewards, not modeling them can give rise to a model that is computationally faster to use and statistically easier to learn. However, if one lacks information about the reward when learning the model, trying to capture as much information as possible from the stream of observations can be a good strategy.

Generating Novel Game Environments. Procedurally generating novel game contents and levels is an active area of research in the game AI community (Summerville et al., 2018), which has been shown to be useful to both training and evaluation of RL agents (Risi & Togelius, 2020; Justesen et al., 2018; Cobbe et al., 2020). There have been attempts to leverage generative models for game design by directly predicting frames (Bamford & Lucas, 2020) or modifying backgrounds to generate new game levels (Kim et al., 2020). However, these works rely on privileged simulation data, and have only been attempted at a small scale, limiting the potential to generate entirely new game environments.

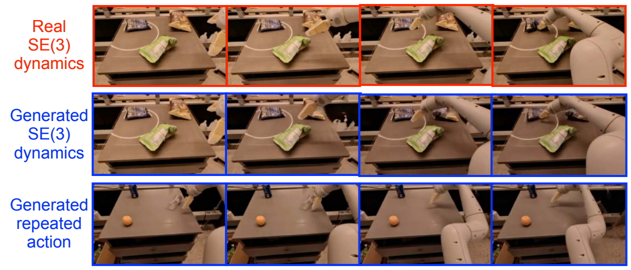

Figure 8: **Generative Simulation of SE(3) Robot Actions.** Real execution of a robot policy (top), simulated execution of the same policy (middle), and simulated execution of repeating the same action (bottom). The simulated rollout generally agrees with the ground truth rollout, but hallucination can happen as the bottle disappears (bottom row).

Recent work has shown it is possible to leverage unlabelled internet-scale gameplay data to learn *latent* actions and then train an action-controllable video model (Bruce et al., 2024). This makes it possible to generate an endless possibility of diverse interactive environments from a prompt image. Figure 7 shows generated game trajectories controlled by human players selecting latent actions given two novel starting frames. While this work remains exploratory, one could imagine a future where it is possible to also integrate learned reward models (Chan et al., 2023; Du et al., 2023a; Escontrela et al., 2023) to train RL agents in fully generative game environments.

4.2 Robotics and Self-Driving.

Simulating the SE(3) Action Space One of the long standing challenges in robot learning is around sim-to-real transfer (Rusu et al., 2017), where policies trained in a simulator fails to transfer to execution on a real robot.Yang et al. (2023b) demonstrated that it is possible to learn an action-conditioned next-frame prediction model on real-robot video data from the Language Table environment (Lynch et al., 2023) with a simple Cartesion action space. In Figure 8, we illustrate that next-frame prediction can predict the visual effect of the more general end-effector actions in the SE(3) space (Blanco-Claraco, 2021).

One immediate use case of a generative SE(3) simulator is to evaluate robot policies, which is particularly useful given the safety considerations associated with real-robot evaluation. Aside from evaluation, Yang et al. (2023b) has trained an RL policy using rollouts from a generative simulator in the Language Table environment. An interesting next step would be to learn a policy from both simulated rollouts and a real environment using a Dyna-style algorithm (Sutton, 1991). Under this setting, real-world videos would be collected as the policy were being executed, which would serve as additional demonstration and feedback for the generative simulator. Lastly, generative simulators can enable effective training of multi-task and multi-environment policies through video rollouts in diverse environments. This was not possible previously, as a policy generally only has access to

Figure 9: **Generative Simulation for Self-Driving.** With internet knowledge, we can simulate different driving conditions at particular locations, such as "rain on Golden Gate Bridge" (top), "dawn in Yosemite" (middle), and "snow on the way to Yosemite" (bottom).

a single real-world environment at a time. Video generation as environment simulation also enables usage of suboptimal data for planning and RL. This would require an additional reward function to evaluate the quality of video generation / simulation. Previous work has shown that vision-language models (VLMs) can serve effective reward functions (Du et al., 2023c; Venuto et al., 2024).

Domain Randomization. Another benefit of generative simulators that is broadly applicable to robotics, navigation, and self-driving is their ability to introduce natural randomness to the training environment to improve real-world transfer of policies trained in simulation. Without generative models, this is achieved through domain randomization by hard-coding rendering rules (Tobin et al., 2017), which is tedious and results in limited environment variations and unrealistic rendering effects. With a generative simulator, recent work has shown that different driving conditions (e.g., sunny, foggy, snowy, rainy, at night) can be introduced into the simulator (Hu et al., 2023). Furthermore, combined with internet-scale knowledge, we can simulate driving conditions at specific locations such as simulating driving in the rain on the Golden Gate Bridge, as shown in Figure 9, which enables training self-driving policies with diverse locations and weather conditions.

4.3 Science and Engineering

Video can serve as a unified representation across a wide range of science and engineering domains, impacting research fields such as medical imaging, computerized image processing, and computational fluid dynamics (Steinman, 2002). Video generation has also been applied to many scientific and engineering domains such as traffic prediction (Kim & Lee, 2021) and cloud movement prediction (Xiong et al., 2018). In situations where visual information can be easily captured by cameras but the underlying dynamical systems are difficult to identify (e.g., cloud movements, atom movements under electron microscopes), video generation models conditioned on the control input can be an effective visual simulator, which can then in tern be used to derive better control inputs. In Figure 10, we illustrate the transition dynamics of silicon atoms on a

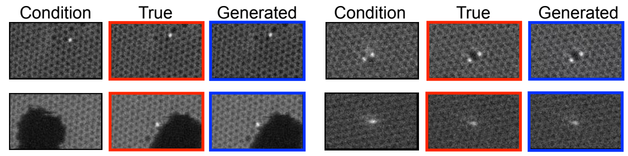

Figure 10: **Atomic-Level Next-Frame Prediction.** The conditional frame, true next frame, and generated next frame reflecting the visual dynamics of silicon atoms on graphene sheets stimulated by electron beams of an electron microscope. Generative models are capable of modeling the visual dynamics with high fidelity.

single layer of carbon atoms, when stimulated by the electron beam of a scanning transmission electron microscope (STEM) using the STEM data collected from Schwarzer et al. (2023). We can see that the generative simulator is capable of characterizing the movement of the silicon atom in the pixel space.

Employing a highly realistic visual simulator in response to control inputs can mitigate the issue of limited hardware access in scientific research endeavors that requires operating specialized equipment, such as electron microscopes. However, leveraging a visual generative simulator for control input optimization requires further investigation to ensure its validity and effectiveness.

In addition to closing the sim-to-real gap in simulating scientific processes, another benefit of generative simulators is that they have a fixed computational overhead which can be beneficial when traditional computational methods are intractable. For instance, simulating calorimeter showers requires computing pairwise interactions between electrons, the complexity of which quickly becomes impractical when the number of electrons are large (Mikuni & Nachman, 2022). Videos of electron showers, on the other hand, have a fixed computational overhead in proportion to the resolution at which showers are being modeled.

5 Challenges

While video generation has great potential, some major challenges for their application still remain. We outline these challenges and potential solutions in this section.

5.1 Dataset Limitations

Limited Coverage. In language modeling, the distribution of language data for solving specific downstream tasks is generally within the distribution of internet text data. However, this is not the case for video. Videos posted on the internet are geared towards human interest, which are not necessarily the video data useful for downstream tasks. For example, models for computational fluid dynamics would likely require many long videos focusing on the movement of fluids such as water; such videos lasting hours would not be very interesting to humans and are thus scarce on the internet. Similarly, it is unusual to find a particular type of robot (e.g., a Franka Emika Panda robot) performing a particular task (e.g., folding clothes) on the internet. This calls for better facilitation to collect and distribute domain specific video data. The Open-X Embodiment dataset for robotics is one such example (Padalkar et al., 2023).

Limited Labels. Another challenge in video modeling is the lack of annotated videos. For example, the MineDojo dataset (Fan et al., 2022) has over 300 thousand hours of humans playing the game Minecraft, but the dataset only has language transcriptions but no game action labels, making it difficult to train policies or environment models using this dataset. Similarly, in the largest open-source robotics dataset (Padalkar et al., 2023), many robot trajectories do not have language annotations on the tasks being performed, or only have generic labels such as "interact with any object". In order to label more video data, prior work has utilized image/video captioning models to provide additional text labels which can be further used to train text-to-image/video models (Betker et al., 2023; Blattmann et al., 2023a). This is similar to video pretraining (VPT) (Baker et al., 2022) except that VPT labels video with action data as opposed to text data. Another possibility is to leverage latent actions/skills inferred from videos (Edwards et al., 2019; Rybkin et al., 2018; Ye et al., 2022), with the largest scale example being Bruce et al. (2024). In Figure 13 in Appendix B, we show examples of the latent actions. Despite the consistency of learned latent actions, it remains an open question as to whether this approach could scale to more complex and diverse dynamics.

5.2 Model Heterogeneity

Unlike how language models have converged on an autoregressive architecture, video generation has yet to settle on the best approach. Autoregressive models, diffusion models, and masked models each have their own advantages and drawbacks.

Diffusion Models. Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2022a) (such as the model used in Section 3.3) have two major advantages. First, they can easily model continuous output spaces without requiring tokenization, which can lead to better generation quality. Second, multiple frames can be sampled in parallel. However, sampling speed in diffusion models is still fairly slow, limiting its applications in real-time simulation. In addition, it is unclear how to generate long video sequences with diffusion models. Diffusion models are also known to be sensitive to hyperparameters such as noise schedules (Croitoru et al., 2023), making training and scaling difficult.

Autoregressive Models. Autoregressive models with a tokenized output space (such as the model mentioned in Section 4.1) are relatively easier to train than diffusion models. Tokenization also allows video generation to be integrated with text or discrete action generation, opening up more applications that require multi-modal generation (Team et al., 2023). Additionally, autoregressive models scale well with context length (Dai et al., 2019; Yan et al., 2023; Bai et al., 2023), allowing them to potentially model very long sequences of frames. However, autoregressive decoding is computationally expensive as each token has to be predicted sequentially. Furthermore, autogregressively bootstraped videos may suffer from the drifting effect (Weng et al., 2023).

Masked Models. Models based on masked reconstruction (such as the model used for generating novel game environments in Section 4.1) can leverage some of the benefits of diffusion and mitigate some of the issues of token-autoregressive modelling by sampling batches of image tokens in parallel (Chang et al., 2022). This allows images composed of thousands of tokens to be sampled with only dozens of model invocations as in Bruce et al. (2024). However, this approach introduces challenges such as sampling bias introduced by the independence assumptions within individual sampling steps.

Better Future Models. Potential solutions to model heterogeneity may require combining the advantages of different models , such as combing autoregressive and masked models (Yan et al., 2023) or combining autoregressive and diffusion models (Weng et al., 2023; Peebles & Xie, 2023; Brooks et al., 2024). In addition, video data might contain redundant information both spatially and temporally. Future models could consider learning latent spaces to reduce the redundency. Better video generation models should also address the current challenges in generation speed and long-term consistency across existing models.

5.3 Hallucination

Hallucination in video generation is common across different types of models. For instance, objects can randomly emerge or disappear (see Figure 8 bottom row and Appendix B.5). This could be due to the loss weight on objects often being not as high as the loss weight on backgrounds since objects are often small. Another type of common hallucination involves implausible dynamics, e.g., a cup "jump" into a robot hand as opposed to a robot grasping a cup. This could be due to videos with coarse temporal frequency not capturing the exact motion-critical frames. Furthermore, generative models that simultaneously model behaviors and dynamics may not distinguish visual changes caused by actions or dynamics (Yang et al., 2022a). Another type of hallucination occurs in the form of violation of causality. Incorporating causal inference algorithms and temporal coherence mechanisms into the video generation models can potentially mitigate such hallucination. By understanding and modeling the causal relationships between objects and actions within a scene, these models could better predict the effects of interactions over time. Hallucination can also occur when a user input is unrealistic given a particular scene,

Figure 11: **Generation from an Unrealistic Instruction.** The input image to the video generation model is a table top with a robot hand. The language instruction is "wash hand". The video model is able to generate egocentric motions to move away from the table top to a kitchen sink in an attempt to fulfill the language instruction realistically.

e.g., "wash hands" is given to a table-top robot. Nevertheless, we haven seen that a video generation model attempts to generate realistic videos by utilizing egocentric motions to fulfill unrealistic user input as shown in Figure 11. Methods such as reinforcement learning with external feedback can be applied to further reduce hallucination in video generation models.

5.4 Limited Generalization

Generating videos from arbitrary image and text input has been difficult. This is especially true for domains that are not well represented by the training data, which, due to limited data coverage challenge discussed in Section 5.1, is quite common in practice. Take diffusion model as an example, it is a common to train on lower resolution videos followed by spatial super-resolution to prevent overfitting (Ho et al., 2022a; Bar-Tal et al., 2024; Xing et al., 2023). We hypothesize that high-resolution images/videos have too much high-frequency information invisible to human eyes, and the focus on which leads to a lack of generalization.

6 Discussion

Integration of Video and Language Generative Models. Neither video or language generation alone is sufficient for real-world decision making. In fact, video and language data contain information along complementary dimensions that need to come together for decision making. Specifically, language contain higher-level abstract information that allows reasoning, search, and planning to be conducted at a conceptual level (and hence more efficiently), meanwhile video contains low-level details that allow abstract plans to be converted into concrete low-level actions (e.g., robot controls). In real-world decision making, the goal is often to derive a sequence of low-level actions from some high-level task description. This derivation process often requires both language and video generation (e.g., use language models to breakdown a high-level task description into step-by-step instructions, and use video generation to turn language instructions into concrete execution plans (Du et al., 2023c)).

Future Directions on Improving Video Generation for Decision Making. We summarize a few concrete future directions in improving video generation models for decision making.

- **Incorporating feedback and self improvement.** One way to improve video generation is to incorporate feedback. This includes human preferences for generated videos similar to RLHF for language models. Perhaps more importantly, generated videos can be converted to robot actions, the execution of which grounds the generated video in the real world by providing real-world feedback. One can also leverage a multimodal model to provide feedback on whether and why generated video is realistic, so that a video generation model can utilize such feedback to self improve.
- **Evaluation.** One key step in improving generation is developing proper evaluation metrics. One approach to evaluating video generation is to convert generated video into real-world actions, executing such actions, and observe the difference between the imagined video and real-world executions.
- **Reducing hallucination during sampling.** Aside from scaling up model and data, one can also reduce hallucination at sampling time by selecting samples that tend to have a higher likelihood. For autoregressive video generation models, this would be similar to beam search. For video diffusion models, this involves using the score function to select samples.

7 Conclusion

We have taken the position that video generation is to physical world as language modeling to the digital world. We have supported this position by showing how, similar to language models, video can represent broad information and tasks. We have further described prior work and new perspectives on applications of video generation combined with reasoning, in-context learning, search, planning, and reinforcement learning to solve real-world tasks. Challenges like hallucination and generalization notwithstanding, video generation models have the potential to become autonomous agents, planners, environment simulators, and compute engines, and to eventually serve as the artificial brain to think and act in the physical world.

8 Impact Statement

This paper argues that video generation holds significant promise for real-world tasks, akin to the accomplishments of language models in digital and intellectual domains. While video generation has profound societal consequences such as impacting privacy and intellectual property, this paper itself focuses on leveraging video generation as task solvers, agents, and environments. Thus, we do not feel that specific ethical aspects and future societal consequences must be highlighted here.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., and Parikh, D. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

Antonoglou, I., Schrittwieser, J., Ozair, S., Hubert, T. K., and Silver, D. Planning in stochastic environments with a learned model. In *International Conference on Learning Representations*. ICLR, 2022.

Bai, Y., Geng, X., Mangalam, K., Bar, A., Yuille, A., Darrell, T., Malik, J., and Efros, A. A. Sequential modeling enables scalable learning for large vision models. *arXiv preprint arXiv:2312.00785*, 2023.

Baker, B., Akkaya, I., Zhokov, P., Huizinga, J., Tang, J., Ecoffet, A., Houghton, B., Sampedro, R., and Clune, J. Video pretraining (vpt): Learning to act by watching unlabeled online videos. *Advances in Neural Information Processing Systems*, 35:24639–24654, 2022.

Bamford, C. and Lucas, S. M. Neural game engine: Accurate learning of generalizable forward models from pixels. In *2020 IEEE Conference on Games (CoG)*, pp. 81–88, 2020. doi: 10.1109/CoG47356.2020.9231688.

Bar, A., Gandelsman, Y., Darrell, T., Globerson, A., and Efros, A. Visual prompting via image inpainting. *Advances in Neural Information Processing Systems*, 35: 25005–25017, 2022.

Bar-Tal, O., Chefer, H., Tov, O., Herrmann, C., Paiss, R., Zada, S., Ephrat, A., Hur, J., Li, Y., Michaeli, T., et al. Lumiere: A space-time diffusion model for video generation. *arXiv preprint arXiv:2401.12945*, 2024.

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y., et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2:3, 2023.

Black, K., Nakamoto, M., Atreya, P., Walke, H., Finn, C., Kumar, A., and Levine, S. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*, 2023.

Blanco-Claraco, J. L. A tutorial on se(3) transformation parameterizations and on-manifold optimization. *arXiv preprint arXiv:2103.15980*, 2021.

Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023a.

Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S. W., Fidler, S., and Kreis, K. Align your latents: High-resolution video synthesis with latent diffusion models. *arXiv preprint arXiv:2304.08818*, 2023b.

Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., Ng, C., Wang, R., and Ramesh, A. Video generation models as world simulators. 2024.

Bruce, J., Dennis, M., Edwards, A., Parker-Holder, J., Shi, Y., Hughes, E., Lai, M., Mavalankar, A., Steigerwald, R., Apps, C., Aytar, Y., Bechtle, S., Behbahani, F., Chan, S., Heess, N., Gonzalez, L., Osindero, S., Ozair, S., Reed, S., Zhang, J., Zolna, K., Clune, J., de Freitas, N., Singh, S., and Rocktäschel, T. Genie: Generative interactive environments. *arXiv preprint arXiv:2402.15391*, 2024.

Chan, H., Mnih, V., Behbahani, F., Laskin, M., Wang, L., Pardo, F., Gazeau, M., Sahni, H., Horgan, D., Baumli, K., Schroecker, Y., Spencer, S., Steigerwald, R., Quan, J., Comanici, G., Flennerhag, S., Neitz, A., Zhang, L. M., Schaul, T., Singh, S., Lyle, C., Rocktäschel, T., Parker-Holder, J., and Holsheimer, K. Vision-language models as a source of rewards. In *Second Agent Learning in Open-Endedness Workshop*, 2023.

Chang, H., Zhang, H., Jiang, L., Liu, C., and Freeman, W. T. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11315–11325, 2022.

Cobbe, K., Hesse, C., Hilton, J., and Schulman, J. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, pp. 2048–2056. PMLR, 2020.

Croitoru, F.-A., Hondru, V., Ionescu, R. T., and Shah, M. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., and Salakhutdinov, R. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019.

Dennett, D. C. *Consciousness explained*. Penguin uk, 1993.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Du, Y., Konyushkova, K., Denil, M., Raju, A., Landon, J., Hill, F., de Freitas, N., and Cabi, S. Vision-language models as success detectors. In *Proceedings of The 2nd Conference on Lifelong Learning Agents*, pp. 120–136, 2023a.

Du, Y., Yang, M., Dai, B., Dai, H., Nachum, O., Tenenbaum, J. B., Schuurmans, D., and Abbeel, P. Learning universal policies via text-guided video generation. *arXiv preprint arXiv:2302.00111*, 2023b.

Du, Y., Yang, M., Florence, P., Xia, F., Wahid, A., Ichter, B., Sermanet, P., Yu, T., Abbeel, P., Tenenbaum, J. B., et al. Video language planning. *arXiv preprint arXiv:2310.10625*, 2023c.

Edwards, A., Sahni, H., Schroecker, Y., and Isbell, C. Imitating latent policies from observation. In *International conference on machine learning*, pp. 1755–1763. PMLR, 2019.

Efros, A. A. and Freeman, W. T. Image quilting for texture synthesis and transfer. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pp. 571–576. 2023.

Escontrela, A., Adeniji, A., Yan, W., Jain, A., Peng, X. B., Goldberg, K., Lee, Y., Hafner, D., and Abbeel, P. Video prediction models as rewards for reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Faccio, D. and Velten, A. A trillion frames per second: the techniques and applications of light-in-flight photography. *Reports on Progress in Physics*, 81(10):105901, 2018.

Fan, L., Wang, G., Jiang, Y., Mandlekar, A., Yang, Y., Zhu, H., Tang, A., Huang, D.-A., Zhu, Y., and Anandkumar, A. Minedojo: Building open-ended embodied agents with internet-scale knowledge. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

Guo, Y., Yang, C., Rao, A., Wang, Y., Qiao, Y., Lin, D., and Dai, B. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

Gupta, A., Tian, S., Zhang, Y., Wu, J., Martín-Martín, R., and Fei-Fei, L. Maskvit: Masked visual pre-training for video prediction. *arXiv preprint arXiv:2206.11894*, 2022.

Hafner, D., Lillicrap, T., Norouzi, M., and Ba, J. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

He, Y., Xia, M., Chen, H., Cun, X., Gong, Y., Xing, J., Zhang, Y., Wang, X., Weng, C., Shan, Y., et al. Animate-a-story: Storytelling with retrieval-augmented video generation. *arXiv preprint arXiv:2307.06940*, 2023.

Himakunthala, V., Ouyang, A., Rose, D., He, R., Mei, A., Lu, Y., Sonar, C., Saxon, M., and Wang, W. Let's think frame by frame with vip: A video infilling and prediction dataset for evaluating video chain-of-thought. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 204–219, 2023.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D. P., Poole, B., Norouzi, M., Fleet, D. J., et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a.

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022b.

Hu, A., Russell, L., Yeo, H., Murez, Z., Fedoseev, G., Kendall, A., Shotton, J., and Corrado, G. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

Huang, J. and Chang, K. C.-C. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*, 2022.

Huang, W., Abbeel, P., Pathak, D., and Mordatch, I. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International Conference on Machine Learning*, pp. 9118–9147. PMLR, 2022.

Justesen, N., Torrado, R. R., Bontrager, P., Khalifa, A., Togelius, J., and Risi, S. Illuminating generalization in deep reinforcement learning through procedural level generation. *CoRR*, abs/1806.10729, 2018.

Kang, X., Ye, W., and Kuo, Y.-L. Imagined subgoals for hierarchical goal-conditioned policies. In *CoRL 2023 Workshop on Learning Effective Abstractions for Planning (LEAP)*, 2023.

Kashin, A. S., Boiko, D. A., and Ananikov, V. P. Neural network analysis of electron microscopy video data reveals the temperature-driven microphase dynamics in the ions/water system. *Small*, 17(24):2007726, 2021.

Kim, H. and Lee, K. Air traffic prediction as a video prediction problem using convolutional lstm and autoencoder. *Aerospace*, 8(10):301, 2021.

Kim, S. W., Zhou, Y., Philion, J., Torralba, A., and Fidler, S. Learning to Simulate Dynamic Environments with GameGAN. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020.

Ko, P.-C., Mao, J., Du, Y., Sun, S.-H., and Tenenbaum, J. B. Learning to act from actionless videos through dense correspondences, 2023.

Kondratyuk, D., Yu, L., Gu, X., Lezama, J., Huang, J., Hornung, R., Adam, H., Akbari, H., Alon, Y., Birodkar, V., et al. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023.

Li, Z., Tucker, R., Snavely, N., and Holynski, A. Generative image dynamics. *arXiv preprint arXiv:2309.07906*, 2023.

Loeschcke, S., Belongie, S., and Benaim, S. Text-driven stylization of video objects. In *European Conference on Computer Vision*, pp. 594–609. Springer, 2022.

Lynch, C., Wahid, A., Tompson, J., Ding, T., Betker, J., Baruch, R., Armstrong, T., and Florence, P. Interactive language: Talking to robots in real time. *IEEE Robotics and Automation Letters*, 2023.

Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Rozière, B., Schick, T., Dwivedi-Yu, J., Celikyilmaz, A., et al. Augmented language models: a survey. *arXiv preprint arXiv:2302.07842*, 2023.

Mikuni, V. and Nachman, B. Score-based generative models for calorimeter shower simulation. *Physical Review D*, 106(9):092009, 2022.

Minsky, M. *Society of mind*. Simon and Schuster, 1988.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *nature*, 518(7540): 529–533, 2015.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Padalkar, A., Pooley, A., Jain, A., Bewley, A., Herzog, A., Irpan, A., Khazatsky, A., Rai, A., Singh, A., Brohan, A., et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

Razavi, A., Van den Oord, A., and Vinyals, O. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Risi, S. and Togelius, J. Increasing generality in machine learning through procedural content generation. *Nature Machine Intelligence*, 2, 08 2020. doi: 10.1038/s42256-020-0208-z.

Rusu, A. A., Večerík, M., Rothörl, T., Heess, N., Pascanu, R., and Hadsell, R. Sim-to-real robot learning from pixels with progressive nets. In *Conference on robot learning*, pp. 262–270. PMLR, 2017.

Rybkin, O., Pertsch, K., Derpanis, K. G., Daniilidis, K., and Jaegle, A. Learning what you can do before doing anything. *arXiv preprint arXiv:1806.09655*, 2018.

Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., Lillicrap, T. P., and Silver, D. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588:604 – 609, 2019. URL https://api.semanticscholar.org/CorpusID:208158225.

Schwarzer, M., Farebrother, J., Greaves, J., Roccapriore, K., Cubuk, E., Agarwal, R., Courville, A., Bellemare, M., Kalinin, S., Mordatch, I., et al. Learning silicon dopant transitions in graphene using scanning transmission electron microscopy. In *AI for Accelerated Materials Design-NeurIPS 2023 Workshop*, 2023.

Searle, J. R. Minds, brains, and programs. *Behavioral and brain sciences*, 3(3):417–424, 1980.

Silver, D., van Hasselt, H., Hessel, M., Schaul, T., Guez, A., Harley, T., Dulac-Arnold, G., Reichert, D. P., Rabinowitz, N. C., Barreto, A., and Degris, T. The predictron: End-to-end learning and planning. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3191–3199. PMLR, 2017.

Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Souček, T., Damen, D., Wray, M., Laptev, I., and Sivic, J. Genhowto: Learning to generate actions and state transformations from instructional videos. *arXiv preprint arXiv:2312.07322*, 2023.

Steinman, D. A. Image-based computational fluid dynamics modeling in realistic arterial geometries. *Annals of biomedical engineering*, 30:483–497, 2002.

Summerville, A., Snodgrass, S., Guzdial, M., Holmgård, C., Hoover, A. K., Isaksen, A., Nealen, A., and Togelius, J. Procedural content generation via machine learning (PCGML). *IEEE Trans. Games*, 10(3):257–270, 2018.

Sutton, R. S. Dyna, an integrated architecture for learning, planning, and reacting. *ACM Sigart Bulletin*, 2(4):160–163, 1991.

Sutton, R. S., Precup, D., and Singh, S. Between MDPs and semi-MDPs: a framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112: 181–211, August 1999. doi: http://dx.doi.org/10.1016/S0004-3702(99)00052-1.

Team, G., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., and Abbeel, P. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 23–30. IEEE, 2017.

Trinh, T. H., Wu, Y., Le, Q. V., He, H., and Luong, T. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.

Valmeekam, K., Marquez, M., Olmo, A., Sreedharan, S., and Kambhampati, S. Planbench: An extensible benchmark for evaluating large language models on planning and reasoning about change. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

Van Den Oord, A., Vinyals, O., et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Venuto, D., Islam, S. N., Klissarov, M., Precup, D., Yang, S., and Anand, A. Code as reward: Empowering reinforcement learning with vlms. *arXiv preprint arXiv:2402.04764*, 2024.

Villalobos, P., Sevilla, J., Heim, L., Besiroglu, T., Hobbhahn, M., and Ho, A. Will we run out of data? an analysis of the limits of scaling datasets in machine learning. *arXiv preprint arXiv:2211.04325*, 2022.

Wang, S., Saharia, C., Montgomery, C., Pont-Tuset, J., Noy, S., Pellegrini, S., Onoe, Y., Laszlo, S., Fleet, D. J., Soricut, R., et al. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18359–18369, 2023a.

Wang, X., Wang, W., Cao, Y., Shen, C., and Huang, T. Images speak in images: A generalist painter for in-context visual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6830–6839, 2023b.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35: 24824–24837, 2022.

Wen, C., Lin, X., So, J., Chen, K., Dou, Q., Gao, Y., and Abbeel, P. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2023.

Weng, W., Feng, R., Wang, Y., Dai, Q., Wang, C., Yin, D., Zhao, Z., Qiu, K., Bao, J., Yuan, Y., Luo, C., Zhang, Y., and Xiong, Z. Art•v: Auto-regressive text-to-video generation with diffusion models, 2023.

Xing, Z., Feng, Q., Chen, H., Dai, Q., Hu, H., Xu, H., Wu, Z., and Jiang, Y.-G. A survey on video diffusion models. *arXiv preprint arXiv:2310.10647*, 2023.

Xiong, W., Luo, W., Ma, L., Liu, W., and Luo, J. Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2364–2373, 2018.

Yadav, A., Phillips, M. M., Lundeberg, M. A., Koehler, M. J., Hilden, K., and Dirkin, K. H. If a picture is worth a thousand words is video worth a million? differences in affective and cognitive processing of video and text cases. *Journal of Computing in Higher Education*, 23: 15–37, 2011.

Yan, W., Hafner, D., James, S., and Abbeel, P. Temporally consistent transformers for video generation. In *International Conference on Machine Learning*, pp. 39062–39098. PMLR, 2023.

Yang, M., Schuurmans, D., Abbeel, P., and Nachum, O. Dichotomy of control: Separating what you can control from what you cannot. *arXiv preprint arXiv:2210.13435*, 2022a.

Yang, M., Du, Y., Dai, B., Schuurmans, D., Tenenbaum, J. B., and Abbeel, P. Probabilistic adaptation of text-to-video models. *arXiv preprint arXiv:2306.01872*, 2023a.

Yang, M., Du, Y., Ghasemipour, K., Tompson, J., Schuurmans, D., and Abbeel, P. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 2023b.

Yang, M. S., Schuurmans, D., Abbeel, P., and Nachum, O. Chain of thought imitation with procedure cloning. *Advances in Neural Information Processing Systems*, 35: 36366–36381, 2022b.

Yang, S., Nachum, O., Du, Y., Wei, J., Abbeel, P., and Schuurmans, D. Foundation models for decision making: Problems, methods, and opportunities. *arXiv preprint arXiv:2303.04129*, 2023c.

Yannakakis, G. N. and Togelius, J. *Artificial Intelligence and Games*. Springer, 2018. https://gameaibook.org.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. *arXiv preprint arXiv:2305.10601*, 2023.

Ye, W., Zhang, Y., Abbeel, P., and Gao, Y. Become a proficient player with limited data through watching pure videos. In *The Eleventh International Conference on Learning Representations*, 2022.

Yu, T., Xiao, T., Stone, A., Tompson, J., Brohan, A., Wang, S., Singh, J., Tan, C., Peralta, J., Ichter, B., et al. Scaling robot learning with semantically imagined experience. *arXiv preprint arXiv:2302.11550*, 2023.

Zhu, J., Yang, H., He, H., Wang, W., Tuo, Z., Cheng, W.-H., Gao, L., Song, J., and Fu, J. Moviefactory: Automatic movie creation from text using large generative models for language and images. *arXiv preprint arXiv:2306.07257*, 2023.

Appendix

A Details of Models Used to Generate Examples in the Main Text

A.1 Autoregressive Model

The model in Section 4.1 is autoregressive in time but uses a masked model (Chang et al., 2022) for each frame in a manner similar to TECO (Yan et al., 2023). For a given trajectory of pixel observation x_t with corresponding action a_t, we model the interleaved sequence $z_0, a_0, z_1, a_1, ...$ where we encode pixel observations x_t into tokens z_t via VQVAE (Van Den Oord et al., 2017) in combination with a vision transformer (Dosovitskiy et al., 2021). We tokenize the actions according to VPT (Baker et al., 2022). We utilize Transformer-XL (Dai et al., 2019) to encode the temporal trajectory $z_0, a_0, z_1, a_1, ...$ with temporally aligned outputs $h_{z_0}, h_{a_0}, h_{z_1}, h_{a_1}, ...$ For steps where the last input was an observation, i.e. h_{z_t}, we utilize the context h_{z_t} as conditioning input to an autoregressive transformer head to predict a_t. If the last input was an action, the context h_{a_0} is conditioning to a masked transformer head to model z_t. Our MaskGIT implementation uses 8 steps with a cosine masking schedule. To further enhance the performance of our interleaved transformer, we initialize the memory using a *past encoder*, an identical transformer separately trained on the interleaved sequence $..., x_{-2}, a_{-2}, x_{-1}, a_{-1}$ utilizing inputs without any discretization.

A.2 Diffusion Model

The diffusion model used to generate examples in Figure 2, Figure 5, Figure 8, Figure 9, and Figure 10 uses the same 3D U-Net architecture as Ho et al. (2022b;a) with interleaved 3D attention and convolution layers in the spatial downsampling pass and spatial upsampling pass. Skip connections are applied to the downsampling pass activations. The model uses pixel-space diffusion as opposed to latent-space diffusion. Following conventions in video diffusion as described in Section 5, the lower resolution video generation model operates at resolution [24, 40], followed by two spacial super-resolution models with target resolution [48, 80] and [192, 320]. Classifier-free guidance (Ho & Salimans, 2022) was applied for text or action conditioning. For frame conditioning, we input the conditioning frame into both the conditional and unconditional model used for classifier-free guidance. To simulate the SE(3) dynamics shown in Figure 8, we employ action discretization similar to Yang et al. (2023b) and Padalkar et al. (2023).

A.3 Masked Model

The masked dynamics model in (Bruce et al., 2024) that generated the novel game environments in Section 4.1 is a controllable video continuation model, producing outputs autoregressively at the frame level, conditioned on unsupervised latent variables that represent the transitions. The latent variables are composed of a discrete set of VQ-VAE codes (Van Den Oord et al., 2017) $\tilde{a}_{1:T-1}$ that are conditioned on frames $x_{1:T}$ and optimized to help predict $\hat{x}_{2:T}$ with a causal transformer. The dynamics model is a transformer with interleaved temporal and spatial attention (Gupta et al., 2022) trained using a masked reconstruction objective, following (Chang et al., 2022). Video tokens are masked with independent random Bernoulli masks at an average rate of 75%, and the dynamics model is trained to predict the missing tokens by minimizing a cross-entropy objective.

At inference time, tokens are generated in parallel following MaskGIT (Chang et al., 2022). Beginning with unmasked context tokens for frames $x_{1:t-1}$ and a fully masked frame x_t, a series of iterative steps are performed, where each step computes the logits for all of the tokens conditioned on $x_{1:t}$ and $\tilde{a}_{1:t}$, a candidate token for each remaining masked position is sampled, and the highest-probability samples are locked in for future steps. In (Bruce et al., 2024) each image is composed of 920 tokens, and they are all eventually sampled over the course of 25 MaskGIT steps.

The model is trained entirely unsupervised on large video datasets; see 13 for example trajectories from a 10.7B parameter model demonstrating that the unsupervised latent action objective results in consistent control variables across a variety of visual prompts.

B Additional Generated Videos

B.1 Additional Game Simulations

Figure 12: **Additional Generated Game Trajectories in Minecraft.**. Using the model in Section 4.1, we demonstrate additional rollouts from the model. We find that the model is able to handle ego-centric motion quite well. However, temporal consistency can sometimes be an issue as shown in the third row. The agent opens up an inventory mid-clip, and then the chest in front disappears.

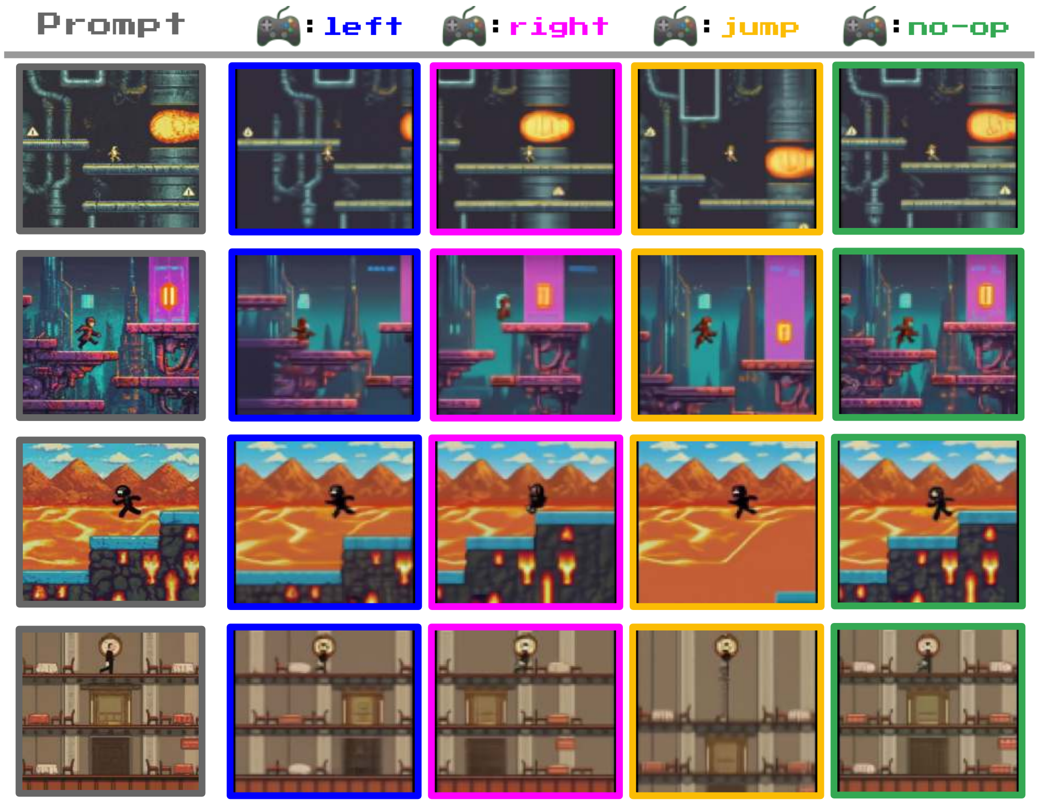

Figure 13: **Additional Simulated Game Dynamics.** Generated frames from (Bruce et al., 2024). Each frame is generated from an initial synthetic prompt image from a text-to-image model, and an unsupervised latent action. Despite the unsupervised nature of the latent actions, their semantics are relatively consistent across initial frames. One limitation of the model is its tendency to generate relatively plain continuations outside the boundaries of the initial image, demonstrated most clearly in the **jump** column of the bottom row.

B.2 Additional Generation for How-to Videos

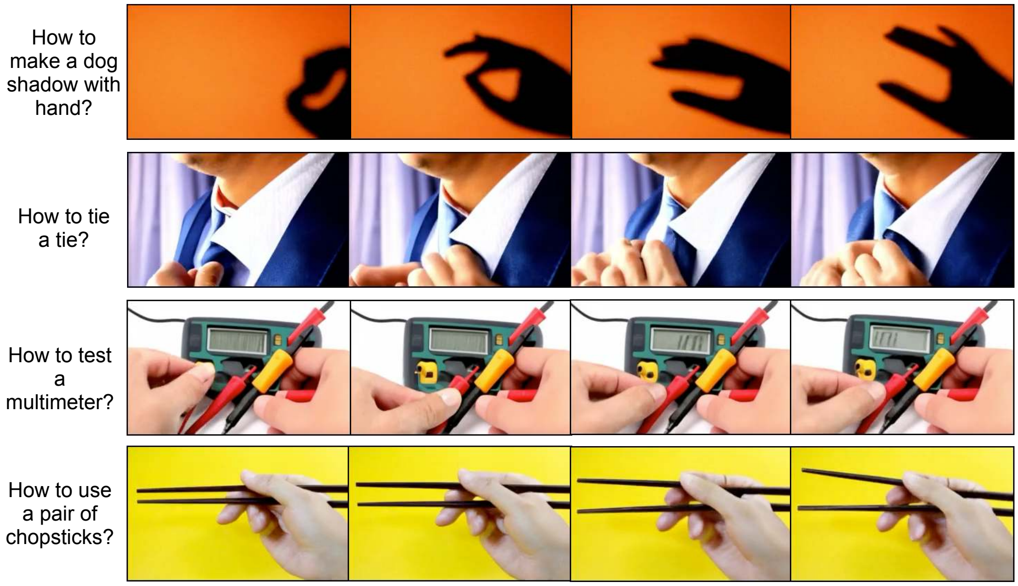

Figure 14: **Additional Generated How-to Videos.** Some generated frames can synthesize key frames in response to human inquiries (first and last row), but some other generated frames are too generic and do not capture enough details to fully answer users' questions.

B.3 Additional Self-Driving Simulations

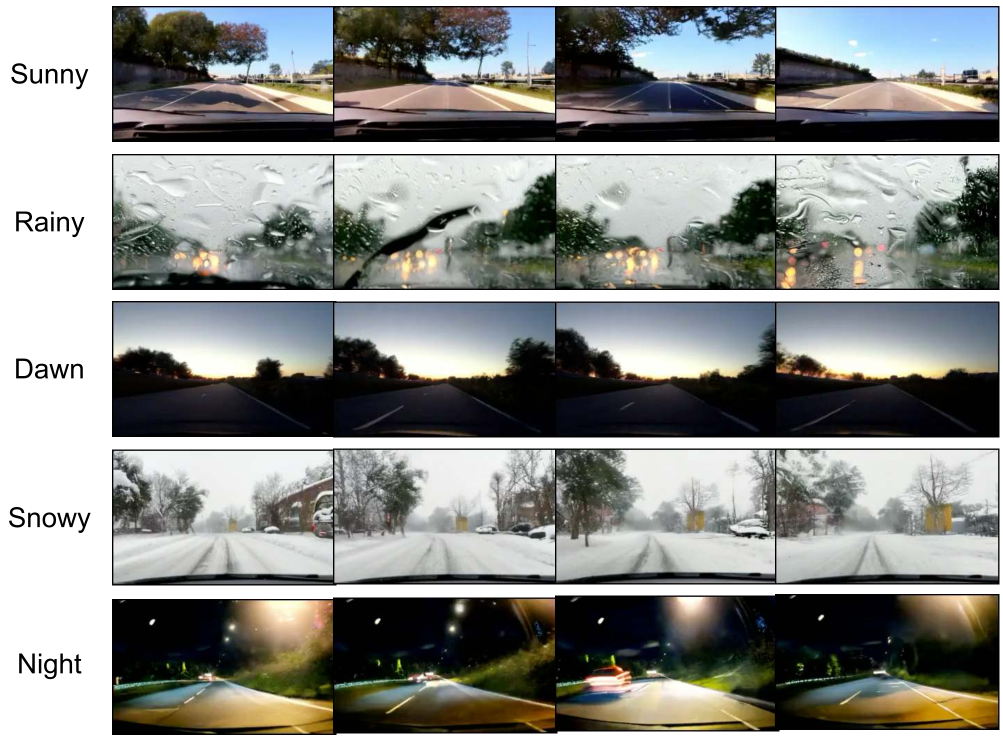

Figure 15: **Additional Generative Simulation for Driving.** Generative simulators can generate driving in different weather conditions and time of the day, such as sunny, rainy, snowy, night, and dawn.

B.4 Additional Robot SE(3) Simulations

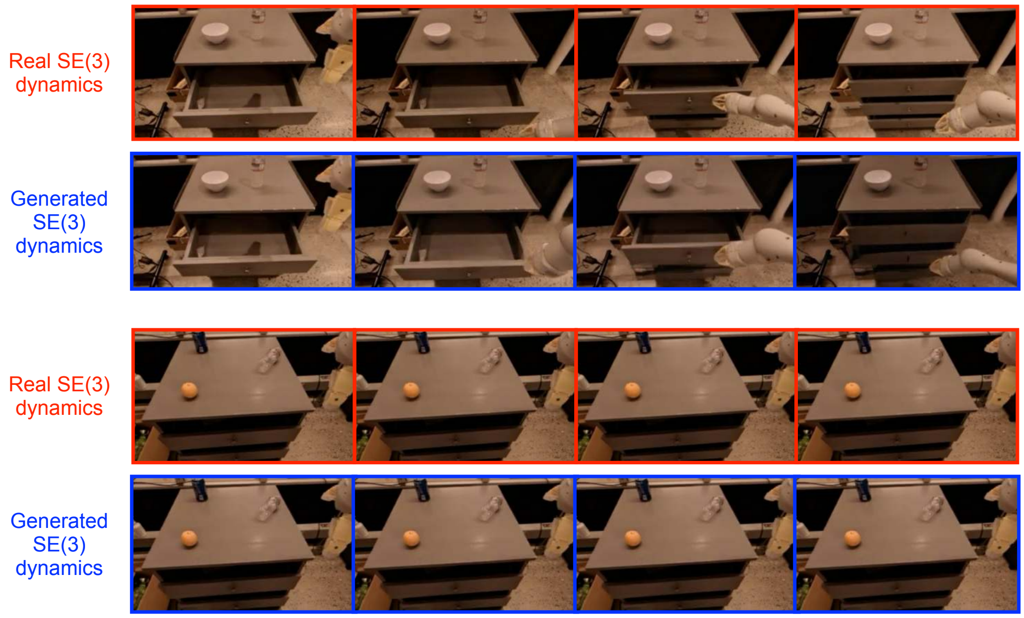

Figure 16: **Additional Generative Simulation of SE(3) Robot Actions.** Real execution of a robot policy (red) and simulated execution of the same policy (blue). The simulated rollout generally agrees with the ground truth rollout.

B.5 Examples of Hallucination

Figure 17: **Examples of Hallucination from All Three Types of Models.** The problem of hallucination persists across different types of video generation models. On the first row, the video generated by the autoregressive model shows that the chest disappears after the inventory is closed. On the second row, the video generated by the diffusion model shows the orange that the orange disappears after being put in the draw. On the bottom row, the video generated by the masked model shows that the cloud suddenly stops at the boundary.