Force Prompting: Video Generation Models Can Learn and Generalize Physics-based Control Signals

Nate Gillman
Brown University

Charles Herrmann*
Google DeepMind

Michael Freeman
Brown University

Daksh Aggarwal
Brown University

Evan Luo
Brown University

Deqing Sun
Google DeepMind

Chen Sun*
Brown University

Abstract

Recent advances in video generation models have sparked interest in world models capable of simulating realistic environments. While navigation has been well-explored, physically meaningful interactions that mimic real-world forces remain largely understudied. In this work, we investigate using physical forces as a control signal for video generation and propose *force prompts* which enable users to interact with images through both localized point forces, such as poking a plant, and global wind force fields, such as wind blowing on fabric. We demonstrate that these force prompts can enable videos to respond realistically to physical control signals by leveraging the visual and motion prior in the original pretrained model, without using any 3D asset or physics simulator at inference. The primary challenge of force prompting is the difficulty in obtaining high quality paired force-video training data, both in the real world due to the difficulty of obtaining force signals, and in synthetic data due to limitations in the visual quality and domain diversity of physics simulators. Our key finding is that video generation models can *generalize* remarkably well when adapted to follow physical force conditioning from videos synthesized by Blender, even with limited demonstrations of few objects (e.g., flying flags, rolling balls). Our method can generate videos which simulate forces across diverse geometries, settings, and materials. We also try to understand the source of this generalization and perform ablations on the training data that reveal two key elements: visual diversity and the use of specific text keywords during training. Our approach is trained on only around 15k training examples for a single day on four A100 GPUs, and outperforms existing methods on force adherence and physics realism, bringing world models closer to real-world physics interactions. We release all datasets, code, model weights, and interactive video demos at our project page, https://force-prompting.github.io/.

1 Introduction

Humans develop an intuitive understanding of how objects respond to forces since infancy (Wilkening and Cacchione, 2010; Ullman et al., 2017): a gentle poke causes a plant to sway, while a breeze creates rippling patterns across fabric. Do video generation models, which encode powerful visual and motion priors through internet-scale pretraining, possess a similar level of intuitive physics understanding? And if so, how to elicit their capabilities to interact with force inputs? A positive answer to these questions would provide a more flexible and expressive interface for video content creation, enable interactive video generation with user input (e.g., generating a video game), and eventually lead to an intuitive world model for intelligent agents to plan and make decisions with.

*Equal advising. Correspondence to: nate_gillman@brown.edu, chensun@brown.edu. 39th Conference on Neural Information Processing Systems (NeurIPS 2025).

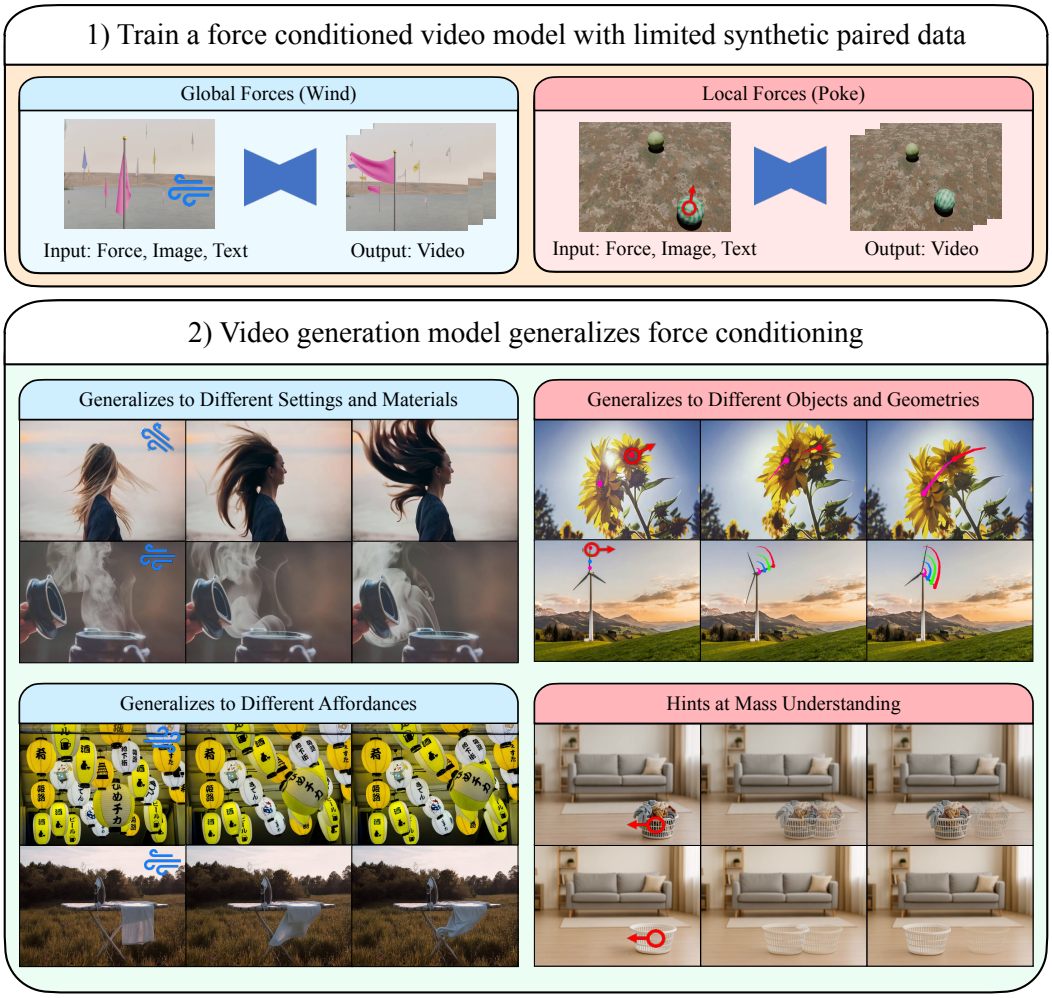


Figure 1: **Force prompting** allows users to apply either global or local forces to objects in an image and then generate the resultant video. Despite being trained on a limited set of synthetic videos (15k for global force and 23k for local force), we observe significant generalization to different settings, materials, objects, geometries, affordances, and some initial hints at mass understanding. Trajectory visualization or alpha overlay are incorporated to better illustrate movement for some examples.

We introduce **Force Prompting**, a step towards incorporating force-based control (direction and magnitude) into video generation models. We explore two distinct categories of force prompts: *local* force prompts, such as instantaneous pokes or pulls applied to specific regions, and *global* force prompts, such as sustained directional wind that affects the entire scene uniformly. Crucially, as manually collecting force annotations from natural videos is both costly and difficult, we instead leverage physics simulators (e.g., Blender) to hand-craft perfectly annotated training data. With our data creation pipeline, we specify a collection of objects along with the force conditions, and simulate the resulting dynamics to obtain the paired training videos. We hypothesize that such *sim2real* generalization is feasible because state-of-the-art video generation models already encode strong priors about visual dynamics, and our paired force-video data serves the role of eliciting their understanding of the physics-based control signals.

We implement Force Prompting by introducing additional force control as local or global vector fields on a video generation model (Yang et al., 2024) conditioned on initial frame and text. We also curate an evaluation benchmark of diverse objects and motion types to evaluate global and local force prompts. As illustrated in Figure 1, our main finding is that despite the synthetic visual appearance and few objects (flying flags and rolling balls) in our training data, video generation models can indeed learn to execute fine-grained force prompts, and exhibit surprisingly strong *generalization* behavior across diverse settings, object shapes and materials, geometry, and affordances. Through

extensive human evaluations, we demonstrate that Force Prompting exhibits superior adherence to physical instruction while maintaining realistic motion and visual quality, when compared to text-conditioned baselines. This validates our hypothesis that synthetic data can teach video generation models intuitive physics and control without damaging their video priors. We further show that simply extrapolating the future by treating forces as local trajectories is insufficient, and our approach significantly outperforms the state-of-the-art in trajectory-controlled video generation (Geng et al., 2024). Notably, Force Prompting can be trained in approximately a single day on four NVIDIA A100 GPUs. We also try to understand the cause of this strong generalization and perform a careful ablation on the training data. We find two elements that appear important to generalization: visual diversity in the training data with respect to the control signal and the usage of certain text keywords at training time, which appear to help elicit the understanding of force control signals.

In summary, our main contributions are as follows:

1. We introduce physical forces as conditioning signals for video generation through two models: one for localized point forces and another for global wind forces.
2. We find that video models can execute precise force prompts with broad generalization to different settings, objects, geometries, and affordances despite minimal training data (15K videos) and modest computational resources (one day on four A100 GPUs). We also attempt to understand the source of this generalization and perform careful ablations on the training data, finding two key elements: visual diversity with respect to the control signal, as well as the usage of text keywords at training time, which appear to help elicit understanding of force control signals.
3. We show that our force-conditioned model has some degree of mass understanding, where the same force can cause a lighter object to move farther than a heavier one.

We release all datasets, code, and models on our project page, https://force-prompting.github.io/.

2 Related Works

Video generation: In the last several years, video generation models have made rapid progress in visual quality and realistic dynamics (Singer et al., 2022; Ho et al., 2022; Blattmann et al., 2023; Girdhar et al., 2023; Bar-Tal et al., 2024; Brooks et al., 2024). In particular, Sora (Brooks et al., 2024) was one of the first video generation models to demonstrate truly compelling diverse real-world physical phenomena and directly advocated for the future use of using video generators as simulators for the physical world. In the last half year, significant progress has been made by open source models such as CogVideoX (Yang et al., 2024) and Wan 2.1 (Wang et al., 2025), even approaching the quality of closed-source models. While these models act as strong video priors, they primarily use text and images as input and lack precise control over general actions or other physical inputs.

Controllable video generation: As video models have rapidly progressed, so too has the accompanying field of controllability for these models with the majority of work in this domain focusing on either camera control (He et al., 2024; Zheng et al., 2024; Sun et al., 2024) or various paradigms of motion control (Yin et al., 2023; Chen et al., 2023; Wang et al., 2024; Shi et al., 2024; Niu et al., 2024; Wu et al., 2024; Geng et al., 2024; Li et al., 2024a; Namekata et al., 2024; Zhang et al., 2024b), such as drag-based, trajectory-based, and optical flow-based techniques. Many of the existing motion control models Yin et al. (2023); Chen et al. (2023); Zhang et al. (2024b) require the complete pre-specified trajectory, specifying the location of the pixel on every generated frame. This reliance on full temporal information makes it difficult to use these models for simulation or prediction tasks.

Motion Prompting (Geng et al., 2024), a concurrent work, uses spatio-temporally sparse trajectories as a conditioning signal, enabling users to specify motion over a few frames for video extrapolation. While this might superficially resemble force control, crucial distinctions exist. First, global phenomena like wind or fluid dynamics are naturally expressed as forces but are difficult or impossible to represent with trajectories. Second, applied forces fundamentally depend on an object's mass or material properties - a dependency absent when specifying motion or location (e.g., identical forces induce greater displacement in lighter objects). Third, specifying an object's location across a few frames is not equivalent to an applied force; the same observed motion could result from numerous alternative causes, such as camera movement or internal object changes. We compare to Motion Prompting and demonstrate significantly better adherence to the conditioning force.

Interactive world models: Paralleling the interest in video generation models, interactive world models (Ha and Schmidhuber, 2018) have gained significant attention. Despite extensive research in this area, investigations have predominantly concentrated on video game environments Valevski et al. (2024); Che et al. (2024); Bruce et al. (2024). While a few contemporary studies have begun exploring real-world applications (e.g., Bar et al. (2024); Agarwal et al. (2025)), none explores interactions besides camera control or text. In contrast, our work focuses on interaction through physical forces.

Physical simulators and hybrid approaches: Early work (Davis et al., 2015a,b) on generating video based on intuitive forces extracts modal bases of vibrating objects in 2D image space; these works, as well as their modern adaptations (Li et al., 2024b), represented motion as a series of vibrations with different frequencies and intensities, which works well for vibration-like motions but struggles to represent many types of motion, such as linear motion. This led to an alternative research direction explicitly incorporating physics solvers (Chen et al., 2022; Zhong et al., 2024; Le Cleac'h et al., 2023; Xie et al., 2024; Zhang et al., 2024a; Huang et al., 2024; Liu et al., 2024a; Lin et al., 2024; Aira et al., 2024). However, almost all of these techniques require the 3D geometry of the scenes. Recent work has focused on combining both physics simulators and generative models, trying to get the best of both worlds: accurate dynamics from the simulator and better appearance from generative models. For example, PhysGen (Liu et al., 2024b) uses a rigid-body physics solver to model object collisions and then renders these scenes through a video generator, and PhysMotion (Tan et al., 2024) uses a combination of a 3D physics solver and a video generation model. However, due to their usage of physics simulators, they are limited in the types of dynamics they can model. In contrast, we explore using the video generation as a simulator and do not use a physics simulator at inference time. We mention some concurrent works as well: (Li et al., 2025a) also explores the use of simulated videos to finetune generative models, but their focus is on modeling object freefall as opposed to learning physics-based control; Li et al. (2025b) explores action-conditioned video generation, but their model requires the use of a physics simulator at inference time; and Wang* et al. (2025) explores force-conditioned video generation, but their framework requires learning a 3D point cloud trajectory model from synthetic data, and then passing that 4D temporal volume into a point cloud-conditioned video generation model.

3 Method: Force Prompting

The goal of Force Prompting is to enable users to interact with images through physical forces. To this end, we explore two distinct force prompts paradigms: a global model that allows users to animate an entire scene with directional wind forces, and a local model that enables precise interaction through localized point forces applied to specific objects within the image. Our video generation method takes as input a triple (τ, ϕ, π), where τ is the text prompt, $\phi \in \mathbb{R}^{c \times h \times w}$ is the initial frame with height h width w and c channels, and π is the physics control signal which represents the force being applied: for the wind force model, this is simply a force vector (magnitude, angle) $\in \mathbb{R}^2$, and for the point force model, this is a force vector (magnitude, angle) $\in \mathbb{R}^2$ along with pixel coordinates $(x, y) \in \mathbb{R}^2$ specifying where to apply the force. The goal is to generate a video $v \in \mathbb{R}^{f \times c \times h \times w}$. While we train the global force and local force models with different synthetic datasets and encode the force inputs differently for each, both models share identical architectures and training procedures.

3.1 Synthetic training data

To construct our global wind force dataset, we use a physics simulator to generate videos of flags waving in the wind. And we construct our local point force dataset in two parts: for the first part, we use a physics simulator to generate videos of a ball rolling across the ground; and for the second part, we use a model (Zhang et al., 2024a) which integrates 3D Gaussians and a physics simulator to generate videos of a plant being poked. We provide more detail below.

Global force dataset: We use Blender to construct a dataset of flags waving in response to varying wind conditions. In order to generate a diverse dataset, we randomize multiple parameters for each video: flag quantity ($\mathrm{Unif}\{1, \ldots, 64\}$), flag color (from a set of 100), flag positions, camera placement, HDRIs (High Dynamic Range Images), which are 360-degree panoramic images used for lighting and background purposes (selected from 50 options on Polyhaven), wind direction in $[0, 360)$, and wind speed in $[0, 1]$, where 0 corresponds to no wind, and 1 corresponds to very strong wind. Each video captures the flags' transitions from stationary to wind-affected state. Our training dataset has 15k videos.

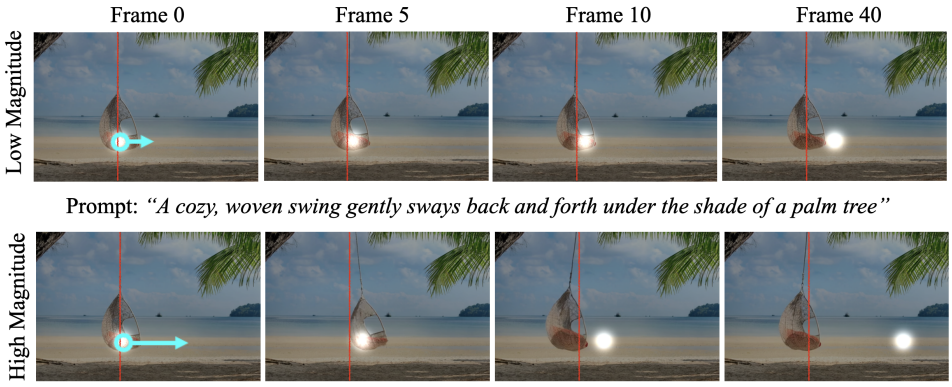


Figure 2: **Visualizing the point force control signal.** The magnitude of applied force is proportional to the gaussian blob's velocity in the control signal, producing proportionally stronger impulses. Stronger forces (bottom) generate faster-moving blobs and correspondingly larger physical responses than gentler forces (top). Note, red line added at the same location in each image for visualization. In our method, we enable the force prompt to dictate the object's trajectory, deliberately excluding such specifics from the text prompt.

Local force dataset: The first scenario in our dataset comprises 12k videos of balls, with one of them rolling in response to being pushed by an unseen point-wise force (the force actor is *not rendered*), and the other balls remaining stationary. We generate these videos using Blender with randomized parameters: ball quantity ($\text{Unif}\{2, 3, 4\}$), ball textures (soccer balls using a Polyhaven mesh [$p = 2/3$] , or bowling balls modeled as smooth spheres [$p = 1/3$]), ball colors (from a set of 108), ball positions, camera position, ground textures (from 42 Polyhaven options), target ball selection, force angle in $[0, 360)$, and force magnitude in [0,1]. We assign the bowling ball to be four times the mass of the soccer ball with the goal of teaching the model mass-based dynamics. The second scenario (11k videos) utilizes PhysDreamer (Zhang et al., 2024a), a *generative-simulator hybrid*, and features videos of a carnation swaying back and forth in response to being poked by an unseen force. We generate these videos with randomized camera position, contact points, force angles, and magnitudes. We use a mixed dataset with the goal of teaching the model that a point force can result in both simple linear motion, and complex oscillatory dynamics, depending on what type of object the force is applied to. In both scenarios, the force magnitude 0 corresponds to a very gentle poke, and the force magnitude 1 corresponds to a much stronger poke.

For both datasets, we project forces from 3D space onto the 2D pixel plane using the camera's parameters. This transformation maps force vectors and object positions from the physical world coordinate system to screen coordinates, allowing us to model forces within the image frame. We generate detailed text prompts using the GPT-4o API, creating a unique descriptions for each HRDI background and ground texture, plus a single shared prompt for all PhysDreamer carnation videos.

3.2 Local and Global Force Prompts

As the wind force is applied globally, and the point force is applied locally, we propose two different force encoding strategies.

Encoding strategy, global force: The wind force control signal is parameterized by a force $F \in [0, 1]$ and an angle $\theta \in [0, 360)$. The goal is to develop a tensor representation for the physics prompt π, which we denote by $\tilde{\pi} \in \mathbb{R}^{f \times c \times h \times w}$. Here, $f = 49$ is the number of frames, $c = 3$ is the number of color channels, and $h = 480$ and $w = 720$ are the height and width of the generated video. We define the first channel of $\tilde{\pi}$ to be $-1 + 2 \cdot F \in [-1, 1]$, the second channel to be $\cos\theta$, and the third angle to be $\sin\theta$. This defines a smooth map $[0, 1] \times [0, 360) \to \mathbb{R}^{f \times c \times h \times w}$ which encodes the angle and magnitude of the wind force field.

Encoding strategy, local force: The point force control signal π specifies a localized force, so it is parameterized by the pixel coordinates $(x, y) \in \{0, \ldots, w - 1\} \times \{0, \ldots, h - 1\}$ in addition to the force magnitude $F \in [0, 1]$ and angle $\theta \in [0, 360)$. At a high level, we define the tensor

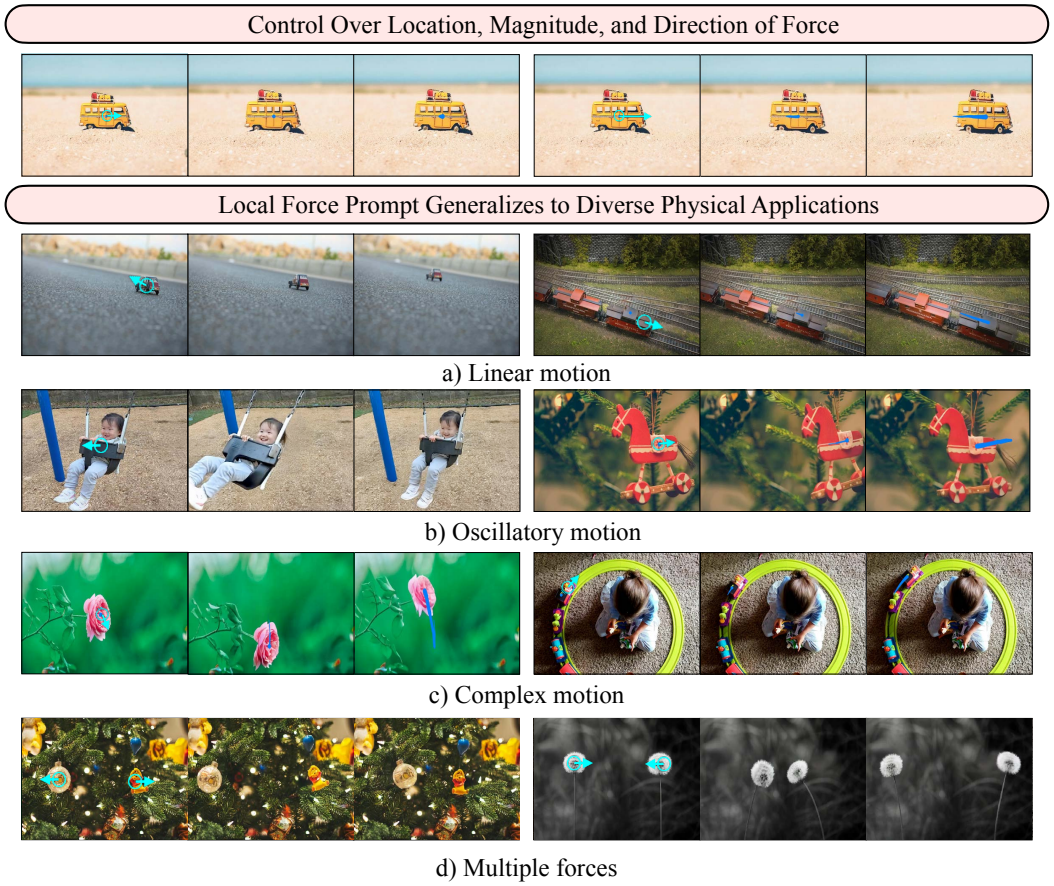


Figure 3: **Qualitative results for the Local Force (Poke) model**. *Top section:* For local forces, the control signal can specify both the location, magnitude, and direction of the force. *Bottom section:* despite the limited training data, the model generalizes to different types of motion. We add blue lines to visualize a time-lapse of some objects' movements.

representation $\tilde{\pi} \in \mathbb{R}^{f \times c \times h \times w}$ for the control signal π to be a sequence of frames where a Gaussian blob starts at the pixel location (x, y), and then moves in the direction θ at a constant velocity, for a total distance affinely proportional to the force F. Full mathematical details in Appendix A.3. This defines a continuous map $\{0, \ldots, w-1\} \times \{0, \ldots, h-1\}] \times [0, 1] \times [0, 360) \to \mathbb{R}^{f \times c \times h \times w}$ which encodes the coordinates where the force is applied, as well as the point force magnitude and angle, into a tensor representation $\tilde{\pi}$. We present a visual example of this in Figure 2. In the case of the local force, we note that the displacement of the Gaussian blob is nonzero when the force is $F = 0$, as our training dataset convention is that $F = 0$ indicates a small force.

We note that force values across the ball rolling and plant poking training videos are not calibrated to any absolute physical scale. Instead, they follow intuitive relative physics where smaller force values (approaching $F = 0$) correspond to gentle pokes resulting in minimal initial displacement, while larger force values produce stronger pokes with correspondingly greater initial displacement. We also wish to highlight that our force prompting models are fundamentally different from video generative models with trajectory-based control such as (Zhang et al., 2024b; Geng et al., 2024). This is because the gaussian blob which serves as the force indicator for the point force model is generally far away from the pixels that it affects, as demonstrated in the complex oscillatory motion of the swaying flower in Figure 2. Similarly, the wind force control signal under-specifies which points must move to which locations, as that control signal is global and causal.

Point Force Model	Linear Motion			Oscillatory Motion			Complex Motion		
	Force Adh.	*Real. Physics*	*Visual Qual.*	*Force Adh.*	*Real. Physics*	*Visual Qual.*	*Force Adh.*	*Real. Physics*	*Visual Qual.*
Text-only, zero-shot	72%	50%	48%	67%	48%	52%	73%	48%	49%
Text-only, fine-tuned	79%	53%	52%	62%	52%	58%	74%	55%	54%
Motion Prompting	91%	93%	100%	89%	76%	99%	86%	76%	98%

Global Force Model	Tethered Motion			Aerodynamic Motion			Fluid Dynamics		
	Force Adh.	*Real. Physics*	*Visual Qual.*	*Force Adh.*	*Real. Physics*	*Visual Qual.*	*Force Adh.*	*Real. Physics*	*Visual Qual.*
Text-only, zero-shot	91%	50%	54%	97%	48%	47%	84%	53%	47%
Text-only, fine-tuned	62%	48%	47%	57%	70%	50%	71%	58%	49%
Motion Prompting	93%	82%	100%	90%	75%	100%	90%	80%	95%

Table 1: **Comparison to baselines.** *Top:* Local point force model. *Bottom:* Global wind force model. We present % win rates of our method against baselines in 2AFC human study results (i.e. values above 50% indicate a preference for Force Prompting) for force adherence, realistic physics, and visual quality. We find that none of the other methods provide consistent adherence to the input force.

3.3 Architecture and Training

We build the force prompting models on top of CogVideoX-5B-I2V (Yang et al., 2024), a video generative model which accepts text and initial frame as conditional inputs. This model generates 49-frame videos at 8-fps. In order to integrate force prompt conditioning, we add a ControlNet (Zhang et al., 2023) which inputs a physics control prompt π, processing it through downscaling, encoding, and temporal compression before combining with hidden states via a zero convolution. The ControlNet clones the first six transformer layers and fine-tunes them while keeping the base model's transformer layers frozen. We base our implementation on (Karachev and Xu, 2025) with modifications to adhere more closely to the original ControlNet design. We train the models on a four 80 GB A100 GPU cluster for 5000 training steps, which takes approximately one day. Training uses an instantaneous batch size per device of 1, with two gradient accumulation steps, for an effective batch size of 8. Full hyperparameter details are listed in Appendix A.1.

4 Quantitative and Qualitative Results

We propose a benchmark dataset for both force prompting models using images that we curate from Pexels. We conduct a 2AFC human study ($N = 10$) using Prolific comparing our force prompting model against three baselines on these benchmark datasets.

Baseline models: The first baseline is *text-only, zero-shot*, which uses the original CogVideoX model and describes the intended force with a string and appending it to the end of the original text prompt. Two example prompt string suffixes are "the apple is moved very forcefully, upwards and to the left", and "the wind is medium strength, blowing right". The second baseline is *text-only, fine-tuned*, which has the same ControlNet architecture as our force prompting model, but with zero-tensor control signals, as well as force suffixes added to the end of the text prompts during training. Our third baseline is *Motion Prompting* (Geng et al., 2024), built on Lumiere (Bar-Tal et al., 2024) (run by the authors). It is the only track-conditioned model that accepts temporally sparse tracks as conditioning signal. We simulate force prompt's impulse by tracing push paths from target objects for the first 3 frames. While the model is meant to accept temporally sparse trajectories, 3 frames of trajectory is out of domain for the intended use case of Motion Prompting.

Human study for local force benchmark: We create a benchmark by curating 63 images from Pexels demonstrating three categories of physical interactions: 1) *linear* movement patterns (toy car, toy train on straight track, hot air balloon); 2) *oscillatory* movement patterns (windmill, pendulums, ornament, and swing); and 3) *complex* movement patterns (toy train on circular track, various plants including ivy, apple tree, and flowers). Table 1 presents human evaluation results for point forces, showing that despite training only on ball rolling (linear) and plant poking (complex) scenarios, our force prompting model demonstrates strong generalization across all motion categories. We note that this model successfully handles multiple forces "zero-shot" during inference, despite only being trained to handle a single force, as seen in Figure 3 and detailed in Appendix B.1.

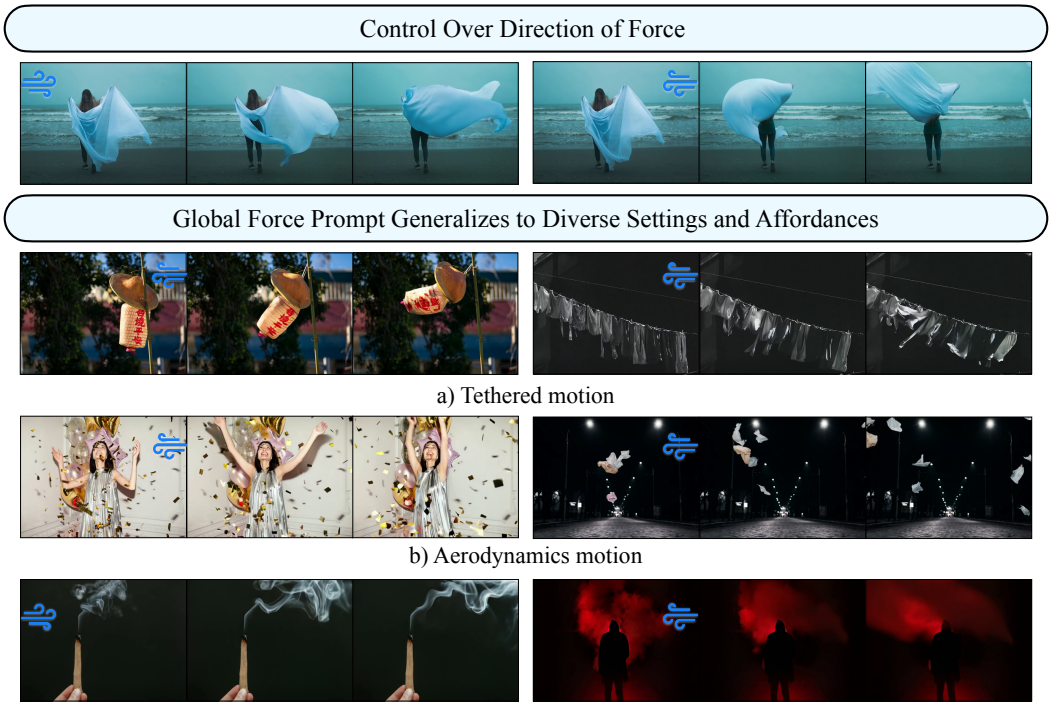


Figure 4: **Qualitative results for the Global Force (Wind) model**. *Top:* from the same starting image, different directions for the force result in different videos. *Bottom:* while the model was only trained on flags, it can generalize to many different settings producing different types of motion.

Human study for global force benchmark: We create a benchmark by curating 41 images from Pexels which demonstrate three different types of physical properties. The first is *tethered motion* (hair, cloth, clothing on person, paper lantern attached to hook). The second is *aerodynamic motion* (bubbles, falling leaves, inflatable tube in pool, floating litter, confetti). And the third is *fluid dynamics* (fog, smoke, snow, steam). In Table 1 we present human evaluation results for the global wind force model. Note that the base CogVideoX model is good at generating videos for all three motion categories (tethered, aerodynamic, fluid dynamic). However, our training data only has tethered motion (flags waving on a flagpole). We observe that the global wind control model trained only on labeled videos with tethered motion results in a model with generalized control over aerodynamic motion and fluid motion as well. We visualize some of these generalization patterns in Figure 4.

Human study comparing to PhysDreamer: The point force model, trained on data from a single carnation, demonstrates remarkable generalization to other plants, as we illustrate in Figures 1 and 3. To evaluate this generalization quantitatively, we compare our approach against PhysDreamer (Zhang et al., 2024a), which employs 3D assets and an integrated physics simulator. Using their benchmark dataset of six plant species, our results in Table 2 show that the point force model successfully generalizes to various roses, tulips, and alocasia without specific training on these plants. While we do not claim to replace physics-based simulation approaches, our purely neural method offers exceptional generalizability and produces responses that align with "intuitive physics," effectively conveying plausible physical interactions to human evaluators. We also also conduct an extended qualitative comparison with 8 other physics simulation models; see Appendix C.2 for more details.

5 Ablation Studies

5.1 Ablation Study #1: Composition of Synthetic Dataset

How do synthetic dataset design choices affect model generalization? In this section we analyze the impact of dataset diversity on force modeling tasks. Sample results are illustrated in Figure 5, more in depth results are in Figure 7 (Appendix), and additional videos are on the project webpage.

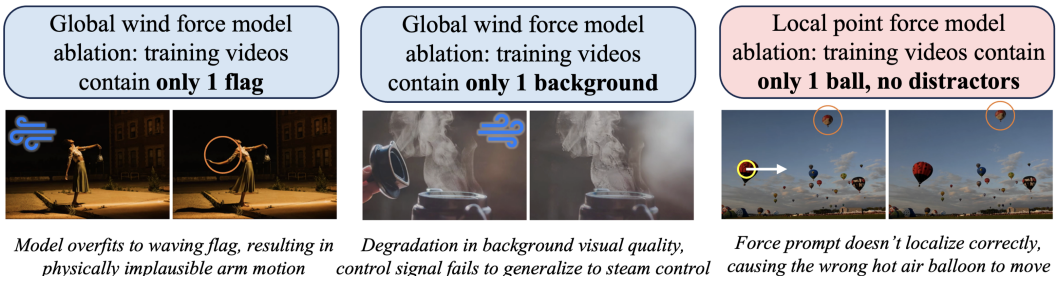


Figure 5: **Results from our ablation studies on synthetic dataset design choices.** *Left:* when the global wind force model is trained on a dataset with only one flag, it overfits, causing the woman's arm to wave unnaturally like fabric. *Middle*: when trained with a single background, the global force model has significantly degraded overall visual quality. *Right:* when trained without distractor objects, the point force model cannot properly localize motion, applying forces indiscriminately rather than to the intended target.

	Alocacia	Carnation	Rose (Orange)	Rose (Red)	Rose (White)	Tulip	**Mean**
Motion Realism	40%	50%	50%	60%	40%	50%	48.33%
Visual Quality	20%	40%	50%	40%	20%	50%	36.67%
Force Adherence	60%	70%	50%	50%	50%	70%	58.33%

Table 2: **Comparison to PhysDreamer.** Values represent the percentage of evaluators preferring the Force Prompting model over PhysDreamer, an approach that uses physics simulation during generation. Values above 50% indicate preference for our force prompting model. The results show that Force Prompting outperforms PhysDreamer on force adherence and achieves comparable performance on motion realism, while PhysDreamer maintains an advantage in visual quality.

Point force training dataset ablation: For the localized point force task, we conduct an ablation study by removing "distractor balls" from scenes, leaving only a single ball affected by the point force. Our results show that the presence distractor balls significantly improves force localization. Without them, the model exhibits undesirable behaviors: when poking one hot air balloon, all balloons move slightly; when poking a rose in a glass vase, both the rose and vase move together, failing to isolate the force application. Visuals are in Figure 7 as well as the project webpage.

Global force training dataset ablation: For the global wind force task, we evaluate two diversity factors: flag quantity and background variety. We find that training with a single background leads to models that follow force physics but frequently fail to differentiate between foreground and background, reducing visual quality. Similarly, when restricting scenes to contain only one flag instead of a variable number ($\text{Unif}\{1, \ldots, 64\}$), the model successfully models cloth mechanics but fails to generalize to other materials. In these cases, smoke from campfires remains unaffected by wind, and confetti either doesn't respond or stays unnaturally suspended. We also observe that bubbles don't respond to wind, while human limbs incorrectly billow like cloth. These failures indicate that insufficient scene diversity causes the model to overfit to stationary backgrounds and limited material interactions. These findings are illustrated in Figure 7 as well as the project webpage. Additionally, we trained a unified model to learn both point force prompts and wind force prompts. We found that this results in more dynamic backgrounds, but has slightly less robust point force control. Additional details are in Appendix C.3.

5.2 Ablation Study #2: Text Prompt Specificity

How does specificity of the text prompt affect model outputs? In this ablation study, we investigate how material descriptions in text prompts affect model generalization through a 2×2 grid search ablation study. We train and test our wind model with and without wind-related keywords (wind/breeze/blow). Our results in Figure 8 and the project webpage show that omitting these keywords during training significantly increases failure cases in our benchmark dataset—fog remains static, lanterns collapse unexpectedly, and steam appears without cause. In contrast, models trained with wind-specific terminology demonstrate superior generalization to diverse wind scenarios. Interestingly, the presence of

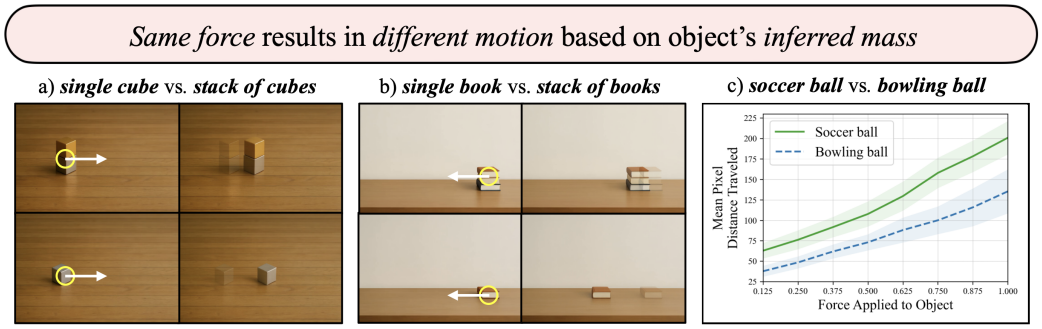


Figure 6: **Mass understanding:** We find that the model has some degree of understanding of mass, in that the same force applied to two objects with different masses will result in different amounts of motion. We demonstrate this qualitatively in (a) and (b) and quantitatively in (c), showing that this result is consistent across a range of force magnitudes. See additional examples in the project webpage.

these keywords during inference has less impact than during training, though using wind terminology generally produces more robust results.

6 Mass Understanding

This section examines the model's capacity for *mass understanding*, which we define to be the ability to recognize that objects with different apparent masses should respond distinctively to the same applied force. A model with robust mass understanding would demonstrate physically intuitive behaviors: a book sliding further than a stack of books when pushed with equal force, or a wooden ornament swinging more freely than its identical metal counterpart under the same impulse. We focus our quantitative analysis on the ball-rolling scenario, as it allows for objective measurement using automatic object detection. Then, we focus our qualitative analysis on other scenarios which present greater challenges for obtaining reliable metrics at scale, such as the swinging ornament.

Force-Mass Relationship Quantitative Study: To quantitatively assess the model's mass understanding, we design an experiment to measure whether soccer balls roll farther than bowling balls when subjected to identical forces. We generate initial condition images across four ground surfaces (dirt, grass, stone, and wood), with three color variations each for both bowling balls and soccer balls. Additional experiment details are in Appendix A.2. Results presented in Figure 6 confirm two key physical principles: the distance traveled increases linearly with applied force for both ball types, and soccer balls consistently travel farther than bowling balls across all force magnitudes, demonstrating the model's intuitive understanding of mass-dependent physics in this scenario.

Force-Mass Relationship Qualitative Study: We evaluate mass understanding across four benchmark tasks featuring geometrically identical objects with different implied masses. Our test scenarios includes ornaments (wooden versus cast iron), laundry baskets (empty versus filled with clothes), book stacks (one, two, or three books), and cube stacks (single versus double cube). To ensure experimental control, we utilize the GPT-Image-1 API to generate initial frames with variations where only the implied mass differs between conditions. Figure 6 presents some of these results, with demonstrating that lighter objects consistently travel farther when subjected to identical forces. This pattern remains robust across four random seeds. This behavior suggests an emergent understanding of mass-dependent physics in our force-prompted model. Other results are in the project webpage. Additionally, we find that the mass understanding behavior persists in the zero-shot multiple objects setting. We include these experimental details in Appendix B.2.

7 Conclusion

We introduce Force Prompting, enabling users to interact with generative video models through physically meaningful controls including localized point forces and global wind effects. Our approach demonstrates that video generation models can successfully learn to respond to force-based conditioning from limited synthetic training data, generalizing remarkably well to diverse objects, materials, and scenarios without requiring physics simulators at inference time. These results suggest a promising direction for developing intuitive world models that respond to natural physical interactions, with potential applications in both creative content generation and embodied AI planning.

Acknowledgments

We would like to thank Bill Freeman, Miki Rubinstein, Junyi Zhang, Junhwa Hur, Noah Fischer, Saining Xie, Calvin Luo, Shijie Wang, Koven Yu, Tian Yun, and Zilai Zeng for useful discussions. We would also like to thank the anonymous NeurIPS reviewers. This project was partially supported by Samsung. Our research was conducted using computational resources at the Center for Computation and Visualization at Brown University.

References

Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. 2025. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*.

Luca Savant Aira, Antonio Montanaro, Emanuele Aiello, Diego Valsesia, and Enrico Magli. 2024. Motioncraft: Physics-based zero-shot video generation. *arXiv preprint arXiv:2405.13557*.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. 2024. Navigation world models. *arXiv preprint arXiv:2412.03572*.

Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, et al. 2024. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11.

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

Tim Brooks, Bill Peebles, Connor Homes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. 2024. Video generation models as world simulators.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. 2024. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*.

Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. 2024. Gamegen-x: Interactive open-world game video generation. *arXiv preprint arXiv:2411.00769*.

Boyuan Chen, Hanxiao Jiang, Shaowei Liu, Saurabh Gupta, Yunzhu Li, Hao Zhao, and Shenlong Wang. 2025. Physgen3d: Crafting a miniature interactive world from a single image. *CVPR*.

Hsiao-yu Chen, Edith Tretschk, Tuur Stuyck, Petr Kadlecek, Ladislav Kavan, Etienne Vouga, and Christoph Lassner. 2022. Virtual elastic objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Tsai-Shien Chen, Chieh Hubert Lin, Hung-Yu Tseng, Tsung-Yi Lin, and Ming-Hsuan Yang. 2023. Motion-conditioned diffusion model for controllable video synthesis. *arXiv preprint arXiv:2304.14404*.

Abe Davis, Katherine L Bouman, Justin G Chen, Michael Rubinstein, Fredo Durand, and William T Freeman. 2015a. Visual vibrometry: Estimating material properties from small motion in video. In *Proceedings of the ieee conference on computer vision and pattern recognition*.

Abe Davis, Justin G Chen, and Frédo Durand. 2015b. Image-space modal bases for plausible manipulation of objects in video. *ACM Transactions on Graphics (TOG)*.

Daniel Geng, Charles Herrmann, Junhwa Hur, Forrester Cole, Serena Zhang, Tobias Pfaff, Tatiana Lopez-Guevara, Carl Doersch, Yusuf Aytar, Michael Rubinstein, et al. 2024. Motion prompting: Controlling video generation with motion trajectories. *arXiv preprint arXiv:2412.02700*.

Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. 2023. Emu video: Factorizing text-to-video generation by explicit image conditioning. *arXiv preprint arXiv:2311.10709*.

David Ha and Jürgen Schmidhuber. 2018. World models. *arXiv preprint arXiv:1803.10122*.

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. 2024. Cameractrl: Enabling camera control for text-to-video generation.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. 2022. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*.

Tianyu Huang, Haoze Zhang, Yihan Zeng, Zhilu Zhang, Hui Li, Wangmeng Zuo, and Rynson WH Lau. 2024. Dreamphysics: Learning physical properties of dynamic 3d gaussians with video diffusion priors. *arXiv preprint arXiv:2406.01476*.

Denis Karachev and Yuancheng Xu. 2025. Cogvideox controlnet extention. `https://github.com/TheDenk/cogvideox-controlnet`.

Simon Le Cleac'h, Hong-Xing Yu, Michelle Guo, Taylor Howell, Ruohan Gao, Jiajun Wu, Zachary Manchester, and Mac Schwager. 2023. Differentiable physics simulation of dynamics-augmented neural objects. *IEEE Robotics and Automation Letters*.

Chenyu Li, Oscar Michel, Xichen Pan, Sainan Liu, Mike Roberts, and Saining Xie. 2025a. Pisa experiments: Exploring physics post-training for video diffusion models by watching stuff drop. *arXiv preprint arXiv:2503.09595*.

Yaowei Li, Xintao Wang, Zhaoyang Zhang, Zhouxia Wang, Ziyang Yuan, Liangbin Xie, Yuexian Zou, and Ying Shan. 2024a. Image conductor: Precision control for interactive video synthesis. *arXiv preprint arXiv:2406.15339*.

Zhengqi Li, Richard Tucker, Noah Snavely, and Aleksander Holynski. 2024b. Generative image dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zizhang Li, Hong-Xing Yu, Wei Liu, Yin Yang, Charles Herrmann, Gordon Wetzstein, and Jiajun Wu. 2025b. Wonderplay: Dynamic 3d scene generation from a single image and actions. *arXiv preprint arXiv:2505.18151*.

Jiajing Lin, Zhenzhong Wang, Shu Jiang, Yongjie Hou, and Min Jiang. 2024. Phys4dgen: A physics-driven framework for controllable and efficient 4d content generation from a single image. *arXiv preprint arXiv:2411.16800*.

Fangfu Liu, Hanyang Wang, Shunyu Yao, Shengjun Zhang, Jie Zhou, and Yueqi Duan. 2024a. Physics3d: Learning physical properties of 3d gaussians via video diffusion. *arXiv preprint arXiv:2406.04338*.

Shaowei Liu, Zhongzheng Ren, Saurabh Gupta, and Shenlong Wang. 2024b. Physgen: Rigid-body physics-grounded image-to-video generation. In *ECCV*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Koichi Namekata, Sherwin Bahmani, Ziyi Wu, Yash Kant, Igor Gilitschenski, and David B Lindell. 2024. Sg-i2v: Self-guided trajectory control in image-to-video generation. *arXiv preprint arXiv:2411.04989*.

Muyao Niu, Xiaodong Cun, Xintao Wang, Yong Zhang, Ying Shan, and Yinqiang Zheng. 2024. Mofa-video: Controllable image animation via generative motion field adaptions in frozen image-to-video diffusion model. *arXiv preprint arXiv:2405.20222*.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Xiaoyu Shi, Zhaoyang Huang, Fu-Yun Wang, Weikang Bian, Dasong Li, Yi Zhang, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, et al. 2024. Motion-i2v: Consistent and controllable image-to-video generation with explicit motion modeling. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. 2022. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*.

Wenqiang Sun, Shuo Chen, Fangfu Liu, Zilong Chen, Yueqi Duan, Jun Zhang, and Yikai Wang. 2024. Dimensionx: Create any 3d and 4d scenes from a single image with controllable video diffusion. *arXiv preprint arXiv:2411.04928*.

Xiyang Tan, Ying Jiang, Xuan Li, Zeshun Zong, Tianyi Xie, Yin Yang, and Chenfanfu Jiang. 2024. Physmotion: Physics-grounded dynamics from a single image. *arXiv preprint arXiv:2411.17189*.

Tomer D Ullman, Elizabeth Spelke, Peter Battaglia, and Joshua B Tenenbaum. 2017. Mind games: Game engines as an architecture for intuitive physics. *Trends in cognitive sciences*, 21(9):649–665.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. 2024. Diffusion models are real-time game engines. *arXiv preprint arXiv:2408.14837*.

Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. 2025. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*.

Chen Wang*, Chuhao Chen*, Yiming Huang, Zhiyang Dou, Yuan Liu, Jiatao Gu, and Lingjie Liu. 2025. Physctrl: Generative physics for controllable and physics-grounded video generation. In *NeurIPS*.

Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. 2024. Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*.

Friedrich Wilkening and Trix Cacchione. 2010. Children's intuitive physics. *The Wiley-Blackwell handbook of childhood cognitive development*, pages 473–496.

Weijia Wu, Zhuang Li, Yuchao Gu, Rui Zhao, Yefei He, David Junhao Zhang, Mike Zheng Shou, Yan Li, Tingting Gao, and Di Zhang. 2024. Draganything: Motion control for anything using entity representation. In *European Conference on Computer Vision*.

Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. 2024. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *CVPR*.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. 2024. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*.

Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. 2023. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. *arXiv preprint arXiv:2308.08089*.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models.

Tianyuan Zhang, Hong-Xing Yu, Rundi Wu, Brandon Y Feng, Changxi Zheng, Noah Snavely, Jiajun Wu, and William T Freeman. 2024a. Physdreamer: Physics-based interaction with 3d objects via video generation. In *ECCV*.

Zhenghao Zhang, Junchao Liao, Menghao Li, Zuozhuo Dai, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. 2024b. Tora: Trajectory-oriented diffusion transformer for video generation. *arXiv preprint arXiv:2407.21705*.

Guangcong Zheng, Teng Li, Rui Jiang, Yehao Lu, Tao Wu, and Xi Li. 2024. Cami2v: Camera-controlled image-to-video diffusion model. *arXiv preprint arXiv:2410.15957*.

Licheng Zhong, Hong-Xing Yu, Jiajun Wu, and Yunzhu Li. 2024. Reconstruction and simulation of elastic objects with spring-mass 3d gaussians. In *European Conference on Computer Vision*.

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: We claim that video generation models trained with force prompting can generalize to out-of-domain scenarios, where out-of-domain refers to conditioning images beyond our synthetic dataset. We believe that our validation benchmark, which is a best effort proxy to test physics-informed generation, paired with our human evaluation study is extensive and diverse enough to support this claim.

   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: We enumerate specific failure modes of our approach in a dedicated section on failures and limitations. We also highlight specific ablations in our synthetic data construction and text-prompt construction that highlight the limitations and failures that we remedied on the path to our final reported results.

   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: We do not present theoretical results in this work.

   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 release our dataset generation code, evaluation benchmark dataset, model training / evaluation code, and detailed instructions for how to reproduce our results therein. We also provide model checkpoints, which may be used to produce exactly the videos used in our human evaluation.

   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 provide all artifacts required to reproduce our work, including code, datasets, survey forms, and precise instruction.

   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: We release the entire data generation pipeline, the data itself, and training code. For our work, we do not optimize hyperparameters - that is, the settings within the code release are those that produced the checkpoints we distribute.

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: We provide error bars for our quantitative mass understanding study, and clearly describe that the variability comes from distinct surfaces, color, and random seeds.

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: We describe the exact GPU models and quantity for all our experiments in the abstract. We also describe roughly how long it takes to produce our results on said hardware.

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: We aim to uphold fairness to our collaborators, research participants, and provide ample attribution to their respective authors. Our synthetic dataset does not depict humans, and any humans depicted in our evaluation benchmark or paper figures are from open-access and royalty free images where persons have given explicit consent to appear.

   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: In the appendix, we describe a scope that we consider appropriate for our work and emphasize the risks in using our method for precise simulation. We believe the potential for positive impact is high, provided that it is clear that our work is no substitute for precise physics simulators.

    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: Our data are not obtained by internet scraping; they are designed by hand or selected manually. While our work builds upon video generation models, which do have high potential for misuse, we inherit the safety precautions present in the models we fine-tune.

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: We have audited and reviewed respective licenses and user agreements of all our work's dependencies. We seek to give proper and thorough attribution to all authors of code, publications, and software to the best of our knowledge. We deliberately chose Blender as our physics simulator because of its permissive license.

    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: Our evaluation benchmark and training data are distributed along with our code as files. For each sample, we note the image, parameters used in its force information, a text-prompt, and any scenery-related specifics used in its generation.

    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: The appendix contains example questions and screenshots from our human studies.

    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: [No]

    Justification: We do not have human subjects in our work, though we collect responses from survey participants. Our survey asks respondents to answer opinion-based questions about the videos we present; our videos free from flashing colors or sensitive topics that may be harmful so there are no risks to the participant.

    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: we use LLMs for prompt upscaling, but this is a standard technique used by CogVideoX and other generative models to automatically obtain detailed image descriptions.

    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.

A Implementation details

A.1 Training hyperparameters

We train for 5000 steps with an effective batch size of 8 (using gradient accumulation over two steps) and saved model checkpoints every five hundred steps. We used `bf16` mixed precision with `tf32` support and initialized from the `THUDM/CogVideoX-5b-I2V` pretrained weights. ControlNet (Zhang et al., 2023) was initialized from the first six transformer layers, for all three input channels, using a downscaling factor of eight, and a unit weight coefficient. We use the AdamW optimizer (Loshchilov and Hutter, 2019) (initial learning rate 1×10^{-5}, $\beta_1 = 0.9$, $\beta_2 = 0.95$, maximum gradient norm 1.0) under a cosine-with-restarts learning-rate schedule (one cycle, 250 warm-up steps). We set our random seed to 42. Training videos contained up to 49 frames at 8 fps.

There are a total of 42 transformer blocks that we may use to initialize our ControlNet. We chose the first six due to memory constraints. With more compute resources, one could potentially achieve higher quality control. We expose the number of transformer blocks to use in the ControlNet as a command line argument for convenience in our training code release.

A.2 Additional details: mass understanding quantitative study

Each ball is subjected to forces of magnitude $0.125 \cdot n$, where $n \in \{1, \ldots, 8\}$, with 10 videos generated per force value using different random seeds. To ensure experimental control, we utilize the GPT-Image-1 API to generate first frames and their variations, maintaining consistent initial ball positions and shapes across conditions. We automatically detect ball position using a Faster R-CNN model with a ResNet-50-FPN backbone (Ren et al., 2015), tracking the sports ball class (ID 37). We compute the distance traveled as the Euclidean distance in pixel space between detected bounding box centers in the first and last frames.

A.3 Additional details: encoding strategy, point force

For the first frame, we set the pixel values equal to 0 everywhere except for a Gaussian blob of radius 20 centered at (x, y), which gets a value of 1; and at the final frame of the control signal, the blob will have moved a distance of $(\frac{1}{8} + \frac{3}{8}F) \cdot w$ pixels. This ensures that when the force is minimized at $F = 0$, the total displacement is $w/8$, and when the force is maximized at $F = 1$, the total displacement is $w/2$.

A.4 Additional details: comparison with PhysDreamer

We took six flower demo videos from the PhysDreamer teaser videos (Alocacia, Carnation, Orange Rose, Red Rose, White Rose, Tulip) and took a screenshot of a still frame from the video that did not have any force annotation on it. We passed these six frames, as well as the six equivalent force prompts (i.e. the same force vector from the original demo) into the Force Prompting model. We presented these videos side by side in the human study, which we served using Qualtrics. Both videos for a given scene were shown to the human annotator simultaneously, with left/right randomization.

B Additional details: zero-shot multiple forces

B.1 Multiple forces for multiple objects, benchmark

Our experiments confirm that the model successfully handles multiple simultaneous forces without requiring retraining. This capability emerges zero-shot by simply adding multiple Gaussian blobs to the control signal videos—one for each applied force. We include these videos in the project webpage. The 6 scenes we tested are:

- An image with 2 Christmas tree ornaments
- An image with 2 toy cars
- An image with 2 vases, each with a flower in it
- An image with 2 roses leaning diagonally, growing out of the ground

- An image with 2 dandelions growing in a field
- An image with 2 apples on separate branches of a tree

We generated these images using the GPT-Image-1 API. To test each image, we identified two directions where it would be reasonable to poke each object; for example, the dandelions can each be poked to the left and to the right. Then for each image, we used 4 different multiple-force force prompts, representing the 2×2 different ways of combining the two different directions; for example, we can poke both dandelions to the left, poke both to the right, poke one to the right and the other to the left, then poke one to the left and the other to the right. We computed one video for each, using the same seed for all of them, with no cherry-picking. We found that $5/6$ of the videos in this multi-poke benchmark had perfect force adherence, and one of them had nearly perfect force adherence. The only failure case was in the two apples scene; when the left apple is poked to the right and the right apple is poked to the left, they move as if they were both poked to the right.

B.2 Multiple forces for multiple objects, mass understanding

We used the GPT-Image-1 API to construct images where two objects of different apparent masses are present. The first image contains an empty laundry basket on the left side of the frame, and a full laundry basket on the right side of the frame. The second image contains a book on the left side of the frame, and a stack of books on the right side of the frame. We passed both sets of images into the multi-poke Force Prompting inference pipeline and instructed the model to poke both sets of objects towards the middle of the frame with the same force magnitude. Across 8 different force magnitudes $\{0.125 * i, i \in \{1, \ldots, 8\}\}$ we found that the lighter object moved much further, with the heavier item barely moving at all. We include videos on our project page.

C Additional qualitative results

C.1 Failures and Limitations

Figure 9 illustrates and categorizes failure cases of Force Prompting. We observe model correlation issues—for example, in hair-blowing scenarios, faces sometimes reorient based on wind direction, likely reflecting patterns in training data where hair typically blows backward. Our method is fundamentally constrained by the underlying video prior's physical understanding; we focus on controlling existing physical capabilities rather than improving the model's physics comprehension. We defer to other works that specifically aim to enhance physical accuracy in generative models, while noting that our approach benefits from efficiently leveraging the scaling properties of the base model.

C.2 Extended comparison with physics simulation models

To demonstrate the point force model's versatility, we curate a benchmark using first-frame images from prominent physics-in-the-loop papers: PhysDreamer (Zhang et al., 2024a), DreamPhysics Huang et al. (2024), MotionCraft (Aira et al., 2024), PhysGaussian (Xie et al., 2024), PhysGen (Liu et al., 2024b), PhysGen3D (Chen et al., 2025), Physics3D (Liu et al., 2024a), and PhysMotion (Tan et al., 2024). We apply our force prompting approach to these diverse scenarios, including poking plants (alocasia, ficus, bouquet of flowers), moving vehicles (boat in water, toy cars), and household objects (rocking horse). Our video results (see project webpage) illustrate that our purely neural method can handle the same visual scenarios almost as effectively as approaches requiring explicit physics simulation at inference time. These qualitative results are in line with our findings in Table 2.

C.3 Training a unified model

We also train a unified model to learn both point force prompts and wind force prompts. We run inference on this joint model using our point force benchmark, as well as our wind force benchmark. Our findings:

- *More dynamic backgrounds:* in many of the videos, the background moves more dynamically in a natural way. For example, for the apple tree, the surrounding leaves move more naturally

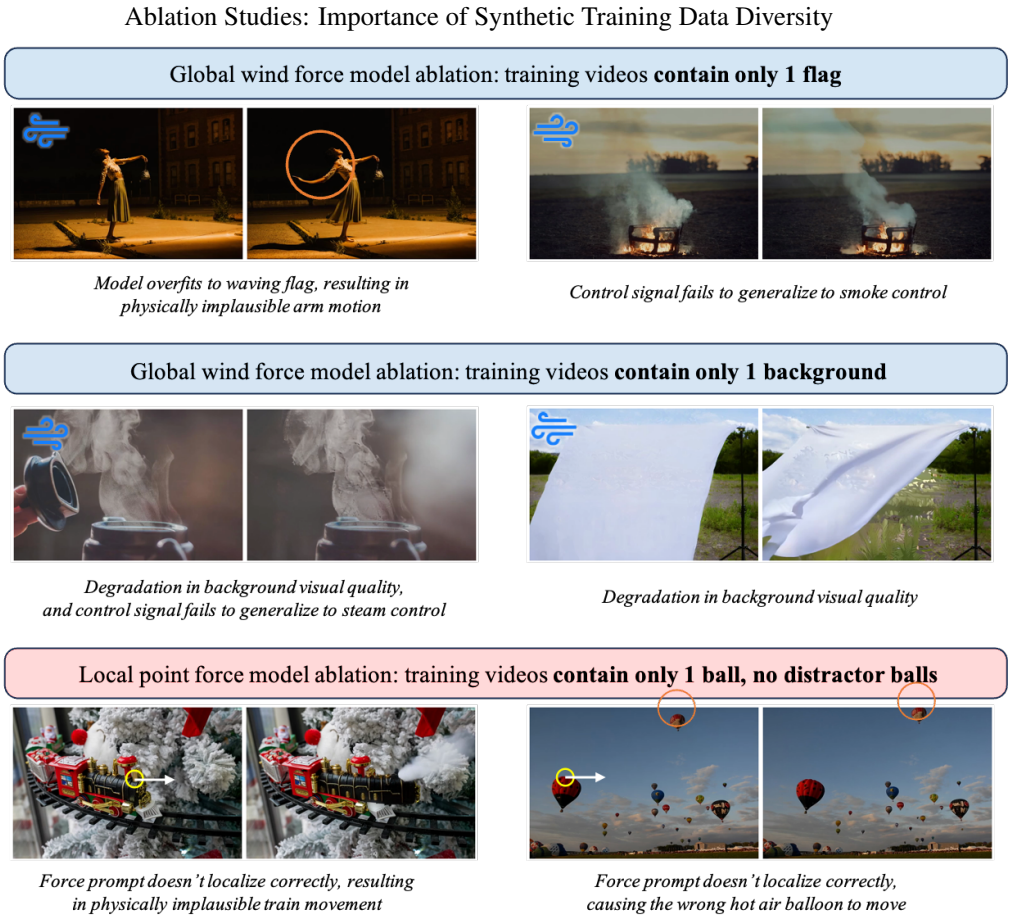


Figure 7: **Results from our ablation studies on synthetic dataset design choices.** *Top:* when the global wind force model is trained on a dataset with only one flag, it overfits, causing the woman's arm to wave unnaturally like fabric and failing to generalize to fluid dynamics scenarios such as smoke. *Middle:* when trained with a single background, the global force model fails to differentiate between foreground and background elements, significantly degrading overall visual quality. *Bottom:* when trained without distractor objects, the point force model cannot properly localize motion, applying forces indiscriminately rather than to the intended target.

after the apple is poked; in the video where a kid is sitting in the middle of a toy train track, the kid is moving more naturally while the train is moving around the track; and in the video with falling leaves in the forest with a woman sitting on a chair in the background, the woman in the chair moves more while the leaves are being blown.

- *Slightly less robust point force control:* on some of the point force videos (e.g. the blueberry bush), the control signal is not respected.

We designed this experiment by sourcing 50% of the training data and control signals in each batch from the synthetic point force dataset, and the other 50% from the synthetic wind force dataset. We trained using the same architecture and number of training steps as the original model.

C.4 Scaling the dataset size

We trained from scratch a wind force model which uses half as many synthetic flag waving videos. The only variable that we changed was the dataset size; everything else (including the number of training steps and learning rate scheduler) remained the same. For this model, we found some additional failure cases. One failure case: for the image where a woman holds a sheet on the beach,

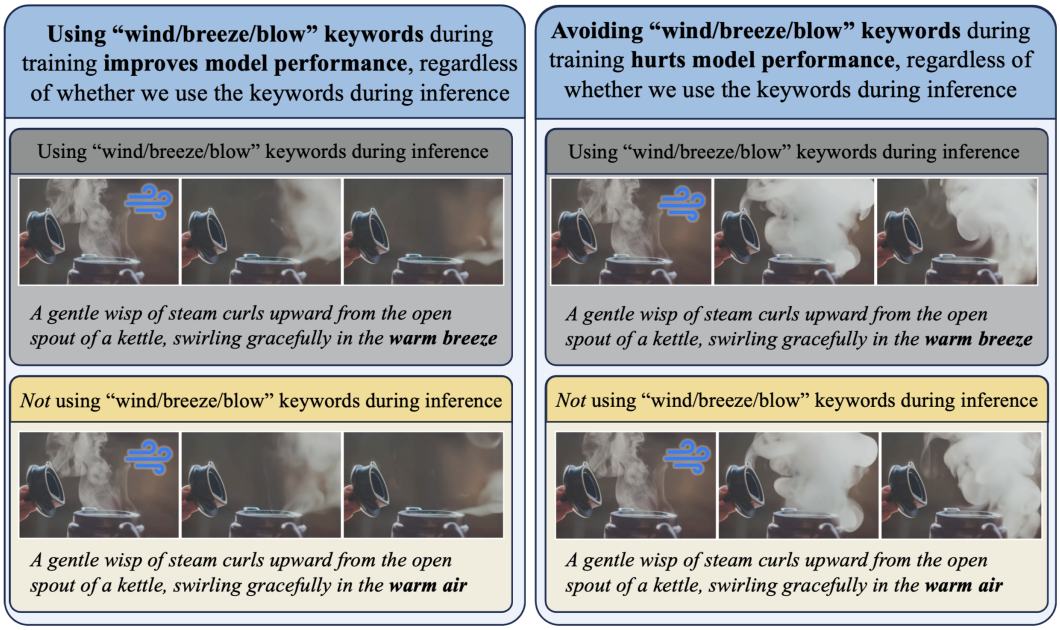


Figure 8: **Results from our ablation studies on text prompt specificity.** In this ablation study, we investigate how material descriptions in text prompts affect model generalization through a 2×2 grid search ablation study. We train and test our wind model with and without wind-related keywords (wind/breeze/blow). Our results demonstrate that omitting these keywords during training significantly increases failure cases in our benchmark dataset. For example, steam is conjured out of thin air instead of being blown correctly. In contrast, models trained with wind-specific terminology demonstrated superior generalization to diverse wind scenarios.

the model hallucinates a bedsheet in the background, indicating that the model has likely memorized the "waving flag" pattern from the training dataset and is injecting it into the output video. A second failure case: the confetti's response to the wind isn't as convincing. For example, some of the confetti will blow in the wind's direction but some will stay stationary. This indicates that the model hasn't generalized properly, perhaps because of less training data.

D Additional emergent phenomena

D.1 Case study #1: Does the model enforce physical affordances?

The Force Prompting models demonstrate a surprising capability to respect object-specific movement constraints. For example, when a train on a circular track is poked forward, it follows the curved trajectory of the track rather than continuing in a straight line—a behavior similarly observed with windmills respecting their rotational axis. We also note interesting emergent behaviors with multi-part objects: poking the lead car of a toy train forward sometimes pulls the entire train along, while other times only the first car moves; conversely, backward forces consistently push the entire train as a unit.

D.2 Case study #2: Does the model understand atomicity of objects?

We evaluate the local force model's sensitivity to the specific pixel chosen as the application point for localized forces. Results demonstrate consistent object movement regardless of which part of the object receives the force. For example, whether poking a train's engine, middle car, or caboose, the entire object responds appropriately to the applied force. This suggests the model has developed a holistic understanding of object wholeness rather than simply responding to pixel-level manipulations.

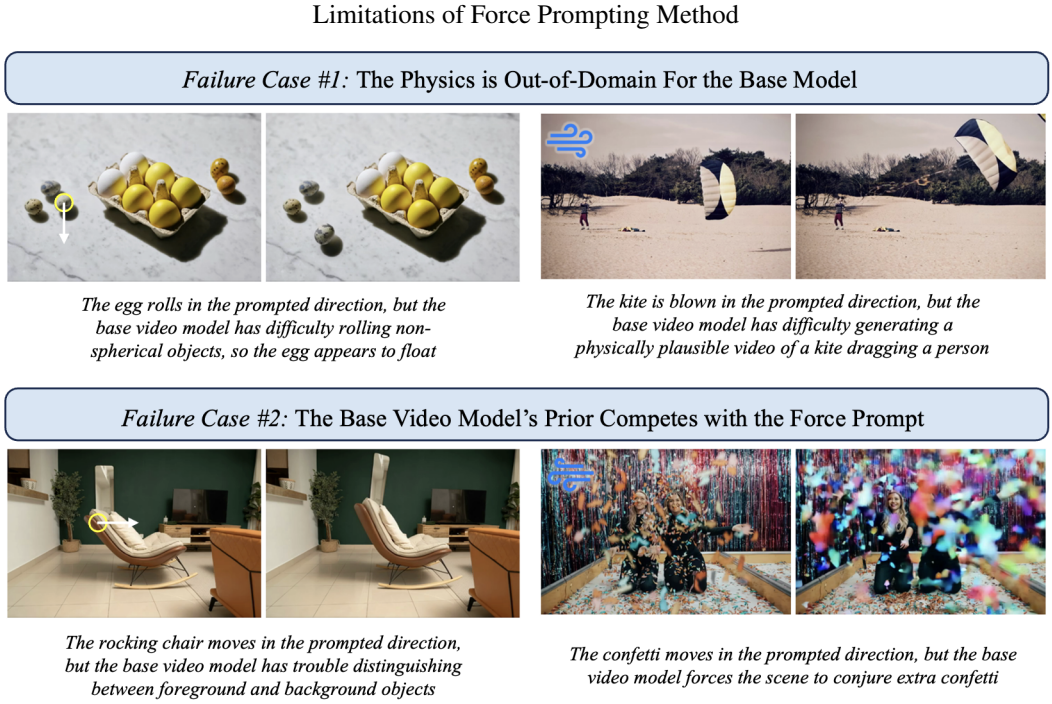


Figure 9: **Analysis of failure cases.** We illustrate and categorize failure cases of Force Prompting. The *top row* shows scenarios where the generated physical motion is out-of-domain for the base CogVideoX model, leading to partial adherence to the force prompt. The *bottom row* depicts failures in visual fidelity or physical realism when the video prior conflicts with the force prompt's intent. More examples are available on our project webpage.

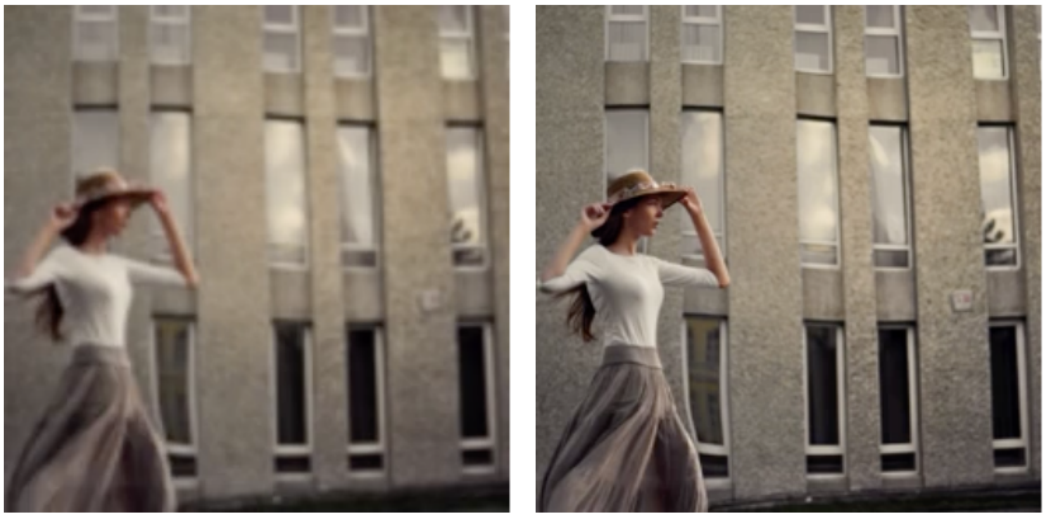
D.3 Case study #3: Does the model preserve cinematic effects of the original image?

The model demonstrates an ability to maintain the original image's stylistic and cinematic properties throughout generated sequences. For example, when animating a toy car from an image with a depth-of-field effect, the model preserves the background blur as the car moves, ensuring visual continuity with the source image's aesthetic. This suggests the model not only understands physical motion but also respects the artistic intent and visual language of the input image. See the project page for videos.

E Impact Statement

This work may be used to enhance the physical plausibility and controllability of video generation models. Applications include video content creation with fine-grained output control for physics-based forces and second-order solutions that leverage enhanced intuitive physics such as motion planning and world simulation. We urge the community to think critically about the potential risks of our work, specifically in the modeling of physical phenomena. Our work is not a substitute for precise physical simulation, rather we focus on what we have described as "intuitive physics", i.e. motion that is visually plausible to humans. Indeed, there are many *unintuitive* physical phenomena in the world where precise and specialist-level simulation is required. We emphasize that our work is unsuitable for use cases requiring high fidelity and precise simulation, including but not limited to materials science, architecture, mechanical engineering, and civil engineering.

Example 1. Please watch both videos. Which video more accurately shows the effect of wind **blowing to the right**?



Video 2 more accurately shows wind blowing to the right since the dress is blown to the right, while in Video 1 the dress is not blown at all.

Figure 10: **A demonstration question from one of our surveys.** Participants are shown an example question with a response along with the reasoning for that response.

F Survey Details and Instructions

We sourced participants from Prolific, compensating responders $12/hr. Our surveys specify the number of questions and an expected time limit. For example, we present the following to participants at the start of the survey:

> *Thank you for taking this survey!* ***It should take less than 25 minutes.***
>
> *There are 208 questions total. You should aim to spend around 7-8 seconds per question. You will be shown two videos, and you must choose which video more accurately* ***shows the effect of the wind blowing in the direction indicated by the question. Please read each question carefully.***
>
> ***There are hidden vigilance questions, so please make sure you answer to the best of your ability.*** *We will be rejecting extremely poor quality responses.*
>
> *At the end of the survey, there is a place to put your Prolific ID so we can confirm you've taken the survey.* ***Please respond only once to this survey*** *(you may have done a similar survey in the past day, that is fine) and thank you for your time!*
>
> *Please do not spend more than 25 minutes on this survey! We don't want to waste your time :)*

We then present participants with example questions and what we consider an appropriate response along with our reasoning, a screen shot of an example can be found in Figure 10. We then present participants with questions following the example for them to answer. They may select their preference from two videos by selecting the radio button underneath their selection, which is depicted in Figure 11.

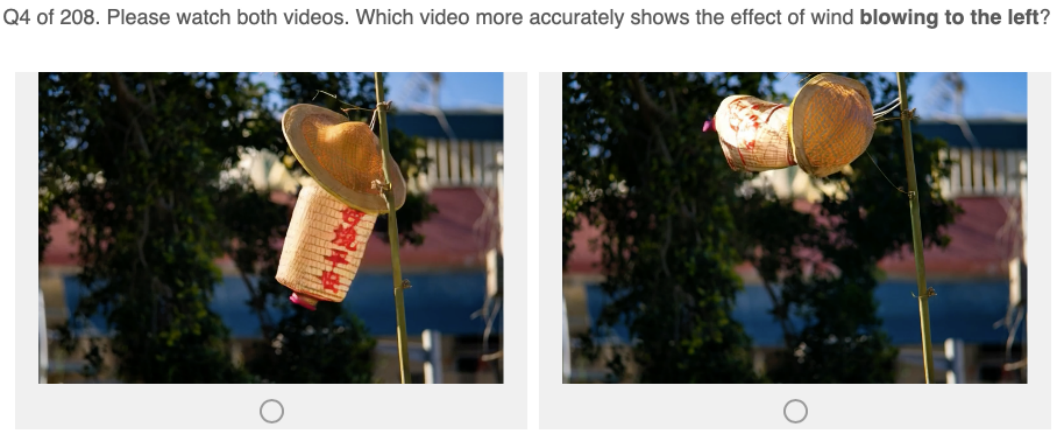


Figure 11: **A question from one of our surveys.** Participants are shown two videos side to side, with radio buttons beneath that they may use to make a selection of which better adheres to the question. The videos play automatically and simultaneously.

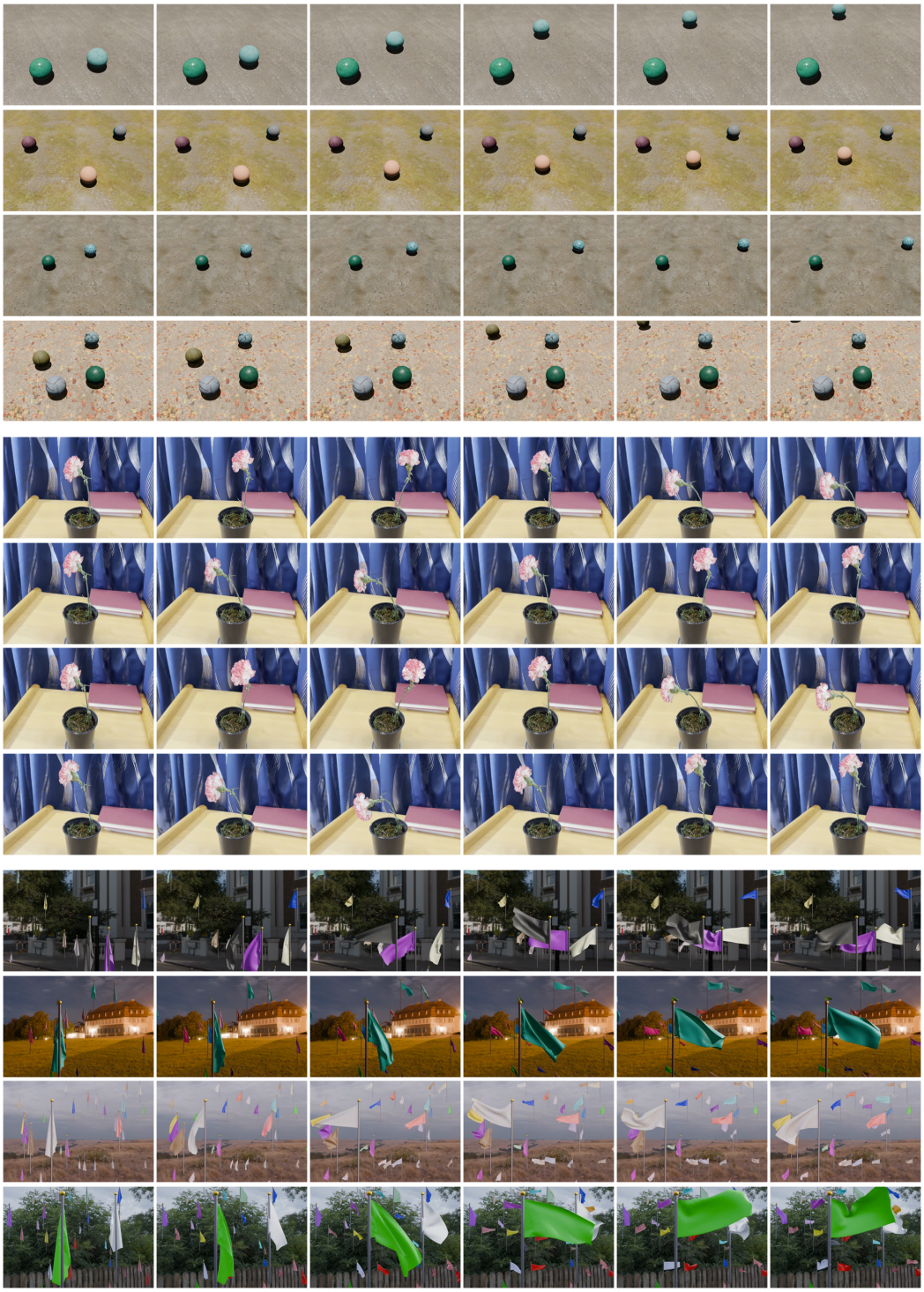


Figure 12: **Samples from our synthetic training datasets.** Top (ball) and middle (flower) are timelapses from our point force training dataset; bottom (flag) are timelapses from the global force training dataset. Our key finding is that video generation models can generalize well when adapted to follow physical force conditioning from videos synthesized by Blender, even with limited demonstrations on few ojects.