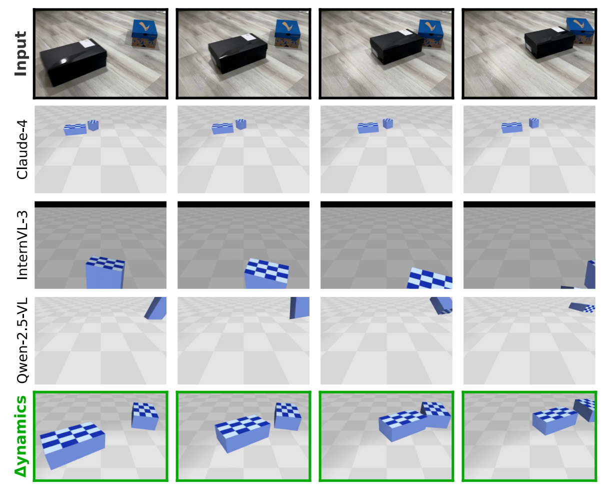
ΔYNAMICS: Language-Based Representation for Inferring Rigid-Body Dynamics From Videos

Chia-Hsiang Kao[1*] Cong Phuoc Huynh[2] Chien-Yi Wang[2] Noranart Vesdapunt[2]
Stefan Stojanov[2] Bharath Hariharan[1] Oleksandr Obiednikov[2] Ning Zhou[2]

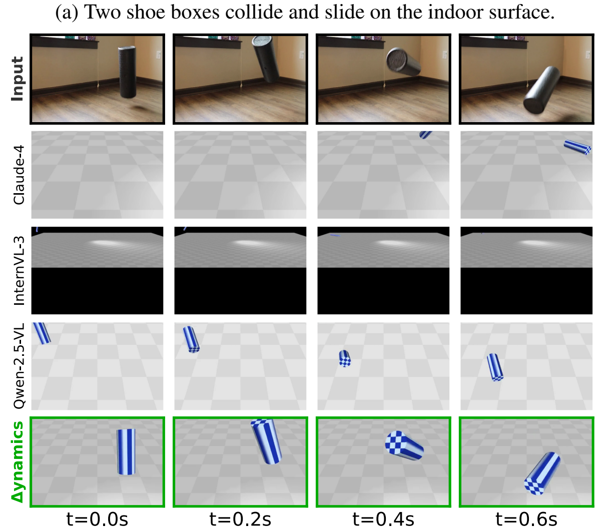
[1]Cornell University [2]Amazon

Abstract

Inferring rigid-body physical states and properties from monocular videos is a fundamental step toward physics-based perception and simulation. Existing approaches assume specific underlying physical systems, object types, and camera poses, which are unable to generalize to complex real-world settings. We introduce ΔYNAMICS, *a vision-language framework that uses language as a unified representation of rigid-body dynamics. Instead of directly predicting parameters,* ΔYNAMICS *generates scene configurations in a structured text format for physics simulation. We enhance the model's generalization by integrating natural language motion reasoning and leveraging optical flow as a semantic-agnostic input. On the CLEVRER dataset [59],* ΔYNAMICS *achieves a segmentation IoU of* 0.30, *a* 7× *improvement over leading VLMs (InternVL3-8B, Qwen2.5-VL-7B and Claude-4-Sonnet). Further, test-time sampling and evolutionary search further boost performance by* 27% *and* 120% *in segmentation IoU, respectively. Finally, we demonstrate strong transfer to a new dataset of* 235 *real-world rigid-body videos, highlighting the potential of language-driven physics inference for bridging perception and simulation. Additional results and videos are available at the project page:* https://iandrover.github.io/2026_dynamics/



(a) Two shoe boxes collide and slide on the indoor surface.



(b) A cylindrical massage roller spins in mid-air.

Figure 1. **Motion transfer from real videos to simulation environments.** ΔYNAMICS accurately reproduces the object shapes, initial position and orientation, material properties, and camera pose with respect to the input videos, while competing VLMs (Claude-4-Sonnet, InternVL-3-8B, Qwen-2.5-VL-7B) fail.

1. Introduction

Understanding physical dynamics from visual observations is a foundational capability for intelligent systems operating in the real world [8, 20, 30, 51]. When perceiving events such as a ball sliding or bouncing, the system should not only identify and track objects but also infer their intrinsic physical attributes, including friction, elasticity, and other parameters that govern motion. These inferred properties enable reasoning about cause and effect, anticipating outcomes under varying conditions, and consequently planning

*Corresponding author: ck696@cornell.edu

and control in embodied settings.

In this work, we focus on rigid-body motion dynamics. Given a video, our goal is to infer the underlying physical model that allows the reproduction of the video trajectories within a simulator. While prior works [4, 14, 19, 23, 26, 53, 54] have made progress in estimating physics parameters from videos in constrained settings, such as sliding boxes [19, 53], billiards [54], or projectiles [4, 14, 23], they are not yet applicable to complex real-world motion that involves multiple object interactions and motion types. First, they assume a model-specific, fixed-length vector of physics parameters for particular object types (e.g., spheres or boxes) and motion types (e.g., sliding or projectile). This representation is not scalable and does not accommodate the full variety of object interactions. Second, prior works typically assume a known or fixed camera pose, thus failing to generalize across varying object distances and camera viewpoints. As a result, prior methods solve only a narrow subset of this video-to-simulation problem and fail to generalize to complex, real-world scenes involving multiple moving objects and unconstrained viewpoints.

A fundamental challenge lies in how the scene itself is parameterized. In this work, we introduce ***a unified, language-based representation of rigid body motion*** as a bridge between perception and simulation. Instead of regressing a fixed-length numeric vector, we reformulate the problem as the generation of symbolic scene configurations specifying object geometry, initial states, material properties, and camera parameters. This language representation is inherently *interpretable* and *scalable* to diverse motion types and object interactions.

This language-centric formulation naturally motivates the use of Vision-Language Models (VLMs) [16, 18, 34, 35, 52], widely employed for visual reasoning [2, 36] and physics understanding [5, 7, 17, 39, 41]. Taking this direction, we develop ΔYNAMICS, a VLM trained on 400K synthetic videos rendered with MuJoCo [50], whose output is a YAML format of the scene configuration. To enhance generalization, we make two key design choices. First, we take optical flow as the input as it is agnostic to visual semantics and background, which provides explicit motion cues and improves full-sequence segmentation IoU by 26% (from 0.19 to 0.24) on CLEVRER [59]. Second, we augment supervision with *natural-language motion descriptions* that capture trajectories, object visibility, and collision events as an auxiliary textual target. Together, these components make ΔYNAMICS robust to domain shifts.

To evaluate cross-domain generalization, we adapt CLEVRER for controlled testing and curate a new dataset of 235 real-world rigid-body motion videos. The reasoning-enhanced model consistently outperforms the vanilla version on real-world transfer, indicating stronger generalization capabilities. We also investigate several test-time enhancement strategies that do not require labeled ground truth in the target domain. We find that best-of-k sampling consistently yields a 10% improvement, and that an additional evolutionary search provides over 50% further gains. For real-world application, we also show the potential of physically plausible video editing using our framework; the corresponding results are deferred to Appendix E.

Our main contributions are summarized below:

- **Language-based representation for motion dynamics:** We reformulate rigid object motion estimation from videos as a *language modeling* problem, where the model generates structural textual scene configurations that are directly consumable by a physics engine.
- **VLM for rigid-body physics inference:** We present ΔYNAMICS, a VLM that directly infers the underlying physics parameters of rigid object motion, which enables the reconstruction of physically-plausible motion trajectories from monocular videos.
- **Cross-domain generalization:** We boost the generalization of models for different physics engines and real-world videos by introducing two key innovations: using optical flow as semantics-agnostic input and training the model to predict natural-language motion descriptions.
- **Comprehensive evaluation benchmarks:** We adapt the CLEVRER dataset for controlled benchmarking and curate a new dataset of 235 real-world rigid-body motion videos with corresponding annotations for segmentation masks and optical flows.

2. Related Work

Rigid-Body Motion Parameter Estimation. Estimating physics parameters of rigid moving objects and camera geometry from videos is a key step towards physics-based perception and simulation. Early efforts tackled the problem in narrow scenarios, e.g., sliding boxes [19, 53], billiard games [54], projectile motion [4, 14], articulated rigid body [26], or free fall [23]), to keep the parameter estimation problem tractable. Furthermore, they assume fixed camera parameters, preventing their applications from general real-world settings [4, 14, 19, 23, 26, 53, 54]. Our contribution is a general solution to the physics parameter estimation problem, which is applicable to a wide spectrum of physical motions in unconstrained real-world settings.

Structured Representations for Visual Content. Representing image and video content using structured graphics programs has been widely adopted for graphics simulation engines such as Blender and MuJoCo format [50]. Recent works in image simulation and generation make use of programmatic formats such as SVG [15, 42, 46, 55–57] and TikZ [9, 10, 48] to formulate the problem as conditional generation of structured text based on textual and image prompts using diffusion models. Further, the use of struc-

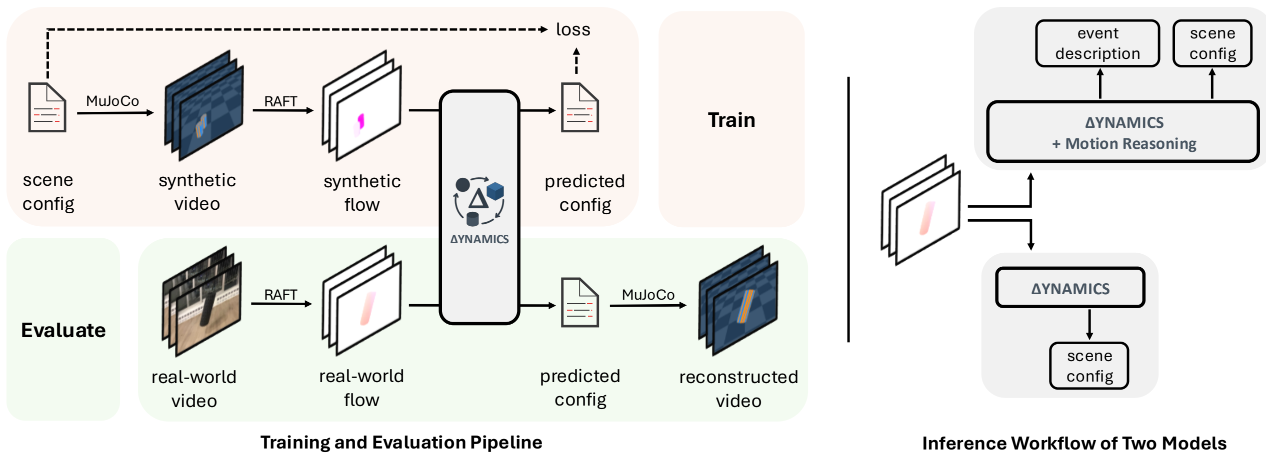


Figure 2. **Training, evaluation and inference workflow for ΔYNAMICS . Training (top left):** We sample scene configurations and render corresponding synthetic videos using the MuJoCo physics engine. Next, we compute optical flows using RAFT [49] and train ΔYNAMICS to generate scene configurations in a structured text format given optical flows. **Evaluation (bottom left):** ΔYNAMICS takes input optical flows derived from real-world videos to infer scene configurations. **Inference (right):** The base variant (bottom) directly generates the scene configuration, whereas the motion reasoning variant (top) first generates a motion event description (details illustrated in Figure 3), then predicts the scene configuration.

tured graphics programs has also been extended to the domain of inverse rendering and the editing and generation of 3D scenes. Specifically, the workflow in [11, 31, 32, 60] involves training VLMs to translate images into a structured format (e.g., JSON) and employing graphic engines for rendering. Other works train VLMs to infer the programmatic representation of existing scenes for graphics editing [24] and 3D asset creation [61]. Our proposed approach is inspired by this school of thought because it offers the capability to interpret both the underlying physics and controllable editing of visual content via scene attributes. However, our research problem is distinguished from prior works by the focus on the modeling of motion dynamics in videos.

Physics Simulation. Another line of research focuses on end-to-end training with differentiable simulation and rendering pipelines [21, 27, 28], enabling physical scene understanding via gradient-based optimization. These works [29, 37, 38, 47, 58] jointly optimize object geometry and physical parameters to directly match the scene dynamics and image formation. Our approach differs in three ways. First, we do not require a predefined physics model; instead, our model learns to infer physical properties and dynamics directly from video observations. Second, we do not assume access to differentiable simulators, as most physics engines are not differentiable in practice. Third, instead of per-scene optimization, we employ a direct feedforward model that predicts a complete scene configuration in a single pass.

3. Method

We present the training, evaluation and inference workflows of ΔYNAMICS in Figure 2. We now formalize the problem and describe each component in detail.

3.1. Problem Statement

We address the problem of recovering physical scene parameters and dynamics from a monocular video. Given an input video $\mathbf{X}$, a model $\mathcal{F}_\theta$ predicts a parameter set $\mathbf{c} = \mathcal{F}_\theta(\mathbf{X})$, which is then provided to a physics engine $\mathcal{S}$ to generate a reconstructed sequence $\hat{\mathbf{X}} = \mathcal{S}(\mathbf{c})$. The objective is to learn $\mathcal{F}_\theta$ such that the simulated dynamics in $\hat{\mathbf{X}}$ faithfully reproduce those observed in the input video.

3.2. Unified Scene Representation

A fundamental challenge lies in how the scene itself is parameterized. Prior work typically regresses a fixed-length parameter vector specific to a particular object set, simulation model, or physics system, which limits generalization across real-world scenarios.

Language Representation. To address these limitations, we shift the core paradigm from numerical regression to symbolic generation. The key idea is a unified, language-based representation that acts as a bridge between perception and simulation. Specifically, we recast physics estimation as a text-generation problem: the model outputs a YAML-formatted sequence that encodes the entire scene configuration, including object geometry, initial states, material properties, and camera parameters. This provides three key advantages:

- **Extensibility and Interpretability.** A textual format scales naturally to scenes with arbitrary numbers of objects. It is human-readable, easy to edit, and conducive to counterfactual analysis. Extended results on physically

Table 1. **Parameter categories.** The complete parameter space spans object properties, initial states, and global parameters.

Category	Parameters
Object Property	*Geometry / Inertial:* radius, height, width, depth, mass. *Material:* friction (rolling, sliding) and damping.
Initial State	*Kinematics:* position, linear and angular velocity. *Orientation:* quaternion.
Global Parameter	*Camera:* pose (height, angle, FOV). *Environment:* gravity.

plausible video editing are provided in Appendix E.

- **Natural Integration with VLM.** Casting simulation as text generation enables end-to-end training of a unified VLM without engineering multi-stage components. We simply format the target as <answer> configuration </answer>, allowing the model to directly output the scene description.
- **Joint Reasoning and Configuration Generation.** Language models can interleave descriptive reasoning with configuration prediction, enabling richer intermediate representations within the same autoregressive process.

Parameterization Details. To operationalize this language-based approach, we define a structured schema for the scene configuration. This configuration represents the full set of geometry, physics, and camera parameters required for simulation, as summarized in Table 1. We compose our scenes using three primitive shapes (spheres, cylinders, and boxes) that can cover common household objects such as tennis balls, soda cans, mugs, books, and crates. These primitives are sufficient to simulate essential rigid-body dynamics, including bouncing, rolling, sliding, and collisions. For the camera, we place it at $(0, -2, h)$, where h denotes its height, and vary the pitch angle while setting roll and yaw to zero. We include gravity as a parameter to account for variations in frame rate or time scale. This scene configuration format supports both single-object and arbitrary multi-object scenes. For example, a scene containing four box-shaped objects includes 20×4 box-specific parameters, along with 3 camera parameters and 1 gravity term, totaling 84 parameters to be estimated.

3.3. Motion Reasoning

By reasoning about motion and object interactions before predicting scene parameters, the model learn richer representations of the underlying dynamics, which in turn improve the accuracy of the subsequent scene parameter estimates. To enable motion reasoning, we train a variant of the model that first generates a natural language description of the observed dynamics and then produces the scene configuration. As shown in Figure 3, these descriptions are derived from simulation traces and artifacts (i.e., ob-

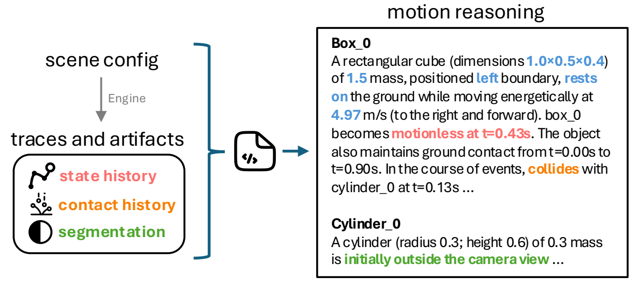


Figure 3. **Synthetic training data generation.** During the data generation process, we create natural language descriptions of motion events. An event-mining script processes the simulation traces and artifacts (left), including state history, contact history, and segmentation maps, to find key dynamic events. The resulting textual descriptions (right) serve as ground-truth targets for the motion reasoning model during training.

ject state histories, contact logs, segmentation masks) together with ground-truth configurations. We specifically consider events such as visibility (e.g., when the object enters or leaves the camera view), motion change (e.g., when it stops rolling or sliding), and collisions (e.g., when it touches the ground or another object), and we design rule-based functions to parse them. These detected events are then inserted into predefined templates to generate a structured, natural language description of the motion. Finally, we prepend the motion description to the scene configuration as <think> description </think> <answer> configuration </answer>. Table 8 provides an illustrative example.

3.4. Motion-Aware Input Representation

Raw RGB videos contain visual semantics unrelated to motion, which can introduce confounding factors for our model. As an alternative, we use optical flow fields to represent motion as it is agnostic to visual semantics and appearance. Specifically, we compute optical flows using RAFT [49] and convert them into a 2D array per color channel, which can further be fed into VLM without architectural changes. We evaluate models trained with both RGB video inputs and the RGB-transformed optical flow maps.

4. Training and Evaluation Method

4.1. Training Approach

Synthetic Data Curation. We generate a dataset of 400K unique physical scenes using the MuJoCo simulator [50]. Each training example is created by sampling a full YAML scene configuration, which is then converted into MuJoCo's XML format to initialize the simulator and assign dynamic states (e.g., initial velocities), as detailed in Appendix A.1. Each data point includes a rendered RGB video of a scene with up to four objects and the corresponding YAML file. To specifically test for compositional generalization, four distinct object-type combinations in four-object scenes (e.g., two boxes and two cylinders) are also held out.

To ensure physically plausible and visually meaningful interactions, we filter out (i) scenes with overlapping objects at initialization, identified using MuJoCo's built-in collision detection; (ii) scenes where more than one object remains outside the camera's field of view throughout simulation; and (iii) scenes where any object is too small (with a total area less than 8000 pixels). Simulations are run for 1 second at 30 FPS with eight physics substeps per frame, rendering images at 480×320 resolution.

Learning Objective. Let $\mathcal{D} = \{(\mathbf{X}_i, \mathbf{c}_i)\}_{i=1}^{N}$ denote the training set, where $\mathbf{X}_i$ is a video observation and $\mathbf{c}_i$ is the corresponding scene configuration, represented as a tokenized text sequence. We aim to learn a model $\mathcal{F}_\theta$ that maximizes the conditional likelihood $p_\theta(\mathbf{c} \mid \mathbf{X})$. This likelihood is modeled autoregressively over the tokens of $\mathbf{c}$ as

$$p_\theta(\mathbf{c} \mid \mathbf{X}) = \prod_{t=1}^{|\mathbf{c}|} p_\theta(c_t \mid \mathbf{X}, c_{<t}), \tag{1}$$

where c_t denotes the t-th token of $\mathbf{c}$. We train $\mathcal{F}_\theta$ by minimizing the negative log-likelihood (NLL) over the dataset:

$$\mathcal{L}_{\text{VLM}} = -\sum_{(\mathbf{X},\mathbf{c})\in\mathcal{D}} \log p_\theta(\mathbf{c} \mid \mathbf{X}). \tag{2}$$

Model and Training Implementation. Our architecture is based on Qwen2.5-VL-3B [6]. The inputs consist of 10 frames uniformly sampled from 1-second, 30 FPS videos. We train two variants, one predicting only scene configuration, and the other generating motion reasoning as an additional output. The format of the target text sequences for these variants has been described in Sections 3.2 and 3.3. We fine-tune the full model for 10 epochs in bfloat16 mixed precision on eight 40 GB A100 GPUs. We employ the AdamW optimizer with a learning rate of 2×10^{-5}, weight decay of 0.01, and a global batch size of 128.

4.2. Test-Time Strategy

To improve instance-level accuracy at inference time, we explore three complementary test-time optimization strategies: best-of-K sampling, preference-based refinement, and evolutionary search.

Best-of-K Sampling. The greedy decoding strategy in VLMs is not guaranteed to find the parameter set with the best quality, as the optimal parameter set may lie in the long tail of the model's output distribution. Hence, we adopt a best-of-N evaluation scheme to explore this distribution: for each case, we generate $N = 32$ diverse predictions with a temperature of 0.1 and top-p of 0.9 and report the *Best@32* performance. This reflects the model's ability to recover accurate physical dynamics with multiple attempts.

Preference Optimization. While ground-truth configurations are typically unavailable for novel environments, the similarities between the forward rendering and the input videos, such as object mask IoU, can serve as *implicit reward signals* for configuration selection without explicit supervision. We conduct experiments with preference rank optimization [45], where details and relevant results are provided in Appendix A.4.

Evolutionary Search. Since preference optimization generally requires training data, it is generally impractical in real-world scenarios. Thus, we explore Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [25], which is an evolutionary algorithm suited for non-convex black-box optimization. When using CMA-ES, we initialize the search with the *Best@32* sample, and we optimize scene configurations, including object sizes, initial states, physics parameters, and camera poses, while keeping object types fixed. We employ a heuristic fitness function that maximizes the segmentation Intersection-over-Union (IoU) while minimizing the optical flow end-point error (EPE), formulated as (IoU − EPE). We use a population size of 128 and optimize for 100 iterations.

4.3. Evaluation Method

Evaluation via Simulation. We evaluate the quality of the predicted configuration $\hat{\mathbf{c}}$ through re-simulation. The generated text is passed to the physics engine $\mathcal{S}$ to produce a simulated RGB video $\hat{\mathbf{X}} = \mathcal{S}(\hat{\mathbf{c}})$, along with auxiliary outputs such as object segmentation masks and optical flow fields. We then compare the simulated outputs against the ground-truth counterparts from $\mathbf{X}$ using segmentation Intersection-over-Union (IoU) and optical flow end-point error (EPE). For synthetic data, ground-truth masks are available from the renderer, while for real-world videos, we use pretrained models [40, 49] for pseudo-annotation.

Metrics. We evaluate across three dimensions:
- Object Composition: Accuracy of object composition.
- Motion Reconstruction Quality: Similarity between the re-simulated and reference videos, measured by segmentation IoU and flow EPE.
- Physics Parameter: L_1 distance between estimated and ground-truth parameters, computed only when the object combination is correct.

Baselines. Since our work is the first to estimate a complete scene configuration for diverse physical systems from a single monocular video, it is not directly comparable to prior methods on physics parameter estimation. For evaluation, we establish baselines using both proprietary and open-source vision–language models (VLMs), including InternVL3-8B [63], Qwen2.5-VL-7B [6], and Claude-

4-Sonnet [1]. Each model is evaluated using *three-shot in-context learning (ICL)*. We adopt ICL over zero-shot prompting because (1) zero-shot generation of a MuJoCo XML file is insufficient for running a simulation, since dynamic states can only be set after engine initialization; and (2) Generating both the XML file and the dynamic-state initialization code is difficult. Details about a few-shot examples are deferred to Appendix A.3.

We also establish non-VLM baselines that directly predict scene configuration parameters from videos. We concatenate these parameters into a fixed-length vector, represent the object type with one-hot encoding, and perform zero-padding for missing objects. Objects are ordered by their x- and then y-positions. We adopt the pretrained ViViT [3] model, designed for video classification, and fully fine-tune it using an ℓ_2 loss on the regression targets.

5. Results on Synthetic Dataset

We divide the synthetic data into two subsets and evaluate the model in the following settings

1. Comparative evaluation: Scenes containing 1–3 objects, with 100 samples for each object count.
2. Complex scene dynamics: 400 four-object scenes constructed from the four specific object-type combinations that were excluded from the training set. 400 scenes with five objects and 400 scenes with six objects, which exceed the object counts seen during training.

5.1. Comparison with Baselines

As shown in Table 2, Claude-4 is the best model among the baselines. While sufficient to roughly identify object composition in a scene, they perform poorly in reconstructing motion trajectories and estimating the physics parameters, with low segmentation IoU (≤ 0.09), and high optical flow errors (> 11) observed.

Meanwhile, the RGB-based ΔYNAMICS model outperforms all baselines in object composition accuracy, segmentation IoU, and parameter estimation, but underperforms in optical flow end-point error (EPE). The higher EPE is primarily due to occasional interpenetration in the predicted initial states, which causes MuJoCo to apply large corrective contact forces to separate overlapping objects, resulting in abrupt motions that increase flow error.

When explicitly conditioned on optical flow, ΔYNAMICS achieves 97% object composition accuracy, improves segmentation IoU, and substantially reduces EPE. One exception is the damping estimation, where the raw-RGB model performs slightly better, likely because the checkerboard ground plane provides additional visual cues helpful to estimating damping parameters. Finally, adding motion reasoning, as shown in the last row, further improves overall performance.

5.2. Evaluation on Complex Scenes

Next, we evaluate the model's generalization ability on unseen scene configurations. In particular, we focus on the behavior of ΔYNAMICS variant with motion reasoning (which is the best one based on the previous section). In Table 3, the model shows only a marginal degradation of segmentation map IoU (compared to the last row of Table 2) for scenes with 4 or 5 objects. Even for scenes with 6 objects, the degradation is gradual and slow. These results show that incorporating motion reasoning adds robustness to more complex, unseen multi-object dynamics.

6. Cross-Engine and Real-World Results

6.1. Cross-Engine Generalization

We assess our model's ability to generalize across a fundamentally different simulation and rendering engine. In particular, we perform this evaluation on the CLEVRER dataset [59]. CLEVRER is a video question-answering benchmark rendered by Blender, which covers sliding motion dynamics for three object types: cubes, spheres, and cylinders. We sampled 100 test videos from the CLEVRER for evaluation. To evaluate against the baseline VLMs, we adopt a similar few-shot, in-context prompting approach (details in Appendix A.3).

Comparison with Baselines In Table 4, ΔYNAMICS consistently demonstrates superior performance compared to baseline models. Furthermore, the previous observations on our synthetic dataset hold true: (i) the model using optical flow inputs outperforms that using RGB videos, and (ii) incorporating reasoning further increases motion reconstruction accuracy, e.g., full-sequence segmentation map IoU increases from 0.24 to 0.29, a 21% relative improvement. Complementary to numerical results, Figure 4 shows that ΔYNAMICS accurately captures multi-object motion trajectories, even in an unseen domain such as CLEVRER.

Test-Time Enhancement. We evaluate the model's performance with different testing time strategies. As shown in Table 5, the vanilla ΔYNAMICS model gains a marginal improvement by sampling more scene configurations. For example, the full-sequence IoU increases from 0.24 (with the greedy sampling approach) to 0.28 (the best out of 32 sampled configurations), a 14% increase.

Consistent with earlier findings, the motion-reasoning variant significantly outperforms the base model, and using best-of-32 sampling further boosts performance: the first-frame IoU increases from 0.30 to 0.38 (+27%), and the full-sequence optical flow EPE decreases by 13%. These relative gains are larger than those achieved by the vanilla model under the same sampling strategy. We hypothesize that the intermediate motion-reasoning step provides a more structured and physically meaningful representation, which

Table 2. **Evaluation metrics for the in-distribution setting on the synthetic evaluation data.** When ΔYNAMICS takes optical flows as the input, it consistently outperforms baseline methods across most evaluation dimensions. Note that parameter estimation metrics are unavailable for non-VLM baselines since they do not correctly predict the object composition. **Best** and runner-up results are highlighted.

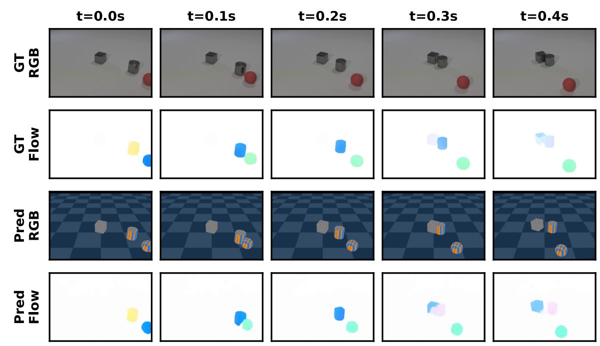
	Input	Obj. Comp. Acc. (↑)	Segmentation Map IoU (↑)		Optical Flow EPE (↓)		Physics Parameter MAE (↓)		
			First-Frame	Full Sequence	First-Frame	Full Sequence	Damping	Roll Friction	Slide Friction
Non-VLM Models									
ViViT [3]	RGB	0.00	0.08	0.07	18.52	9.38	–	–	–
ViViT [3]	Opt. Flow	0.00	0.07	0.06	8.54	8.90	–	–	–
VLM Models									
InternVL3-8B [63]	RGB	0.02	0.05	0.05	25.13	15.77	2.94	0.35	0.81
Qwen2.5-VL-7B [6]	RGB	0.27	0.03	0.03	39.98	16.33	1.97	0.32	0.66
Claude-4-Sonnet [1]	RGB	0.45	0.09	0.07	13.79	11.07	1.71	0.24	0.43
Ours									
ΔYNAMICS	RGB	0.60	0.52	0.32	27.58	19.66	**1.52**	0.16	0.16
ΔYNAMICS	Opt. Flow	0.97	0.88	0.49	5.75	9.24	1.72	**0.15**	0.16
+ Motion Reasoning	Opt. Flow	**0.99**	**0.91**	**0.54**	**4.88**	**8.52**	1.60	0.16	**0.15**

Table 3. **Robustness to complex scene dynamics.** ΔYNAMICS models, trained on up to four objects, generalize effectively to more complex scenes with up to six interacting objects. Structured motion reasoning enhances robustness and consistency under increasing scene complexity. Note that four four-object configurations are held out during training and evaluated here to assess true out-of-distribution generalization.

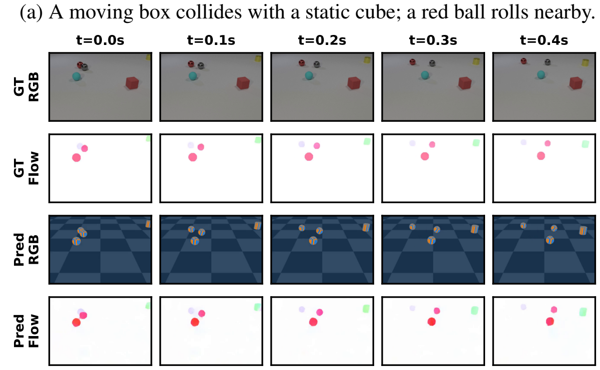
# Objects	Model	Segmentation Map IoU (↑) First Frame	Full Sequence
4	ΔYNAMICS	0.88	0.53
	+ Motion Reasoning	0.89	0.54
5	ΔYNAMICS	0.87	0.51
	+ Motion Reasoning	0.88	0.54
6	ΔYNAMICS	0.85	0.50
	+ Motion Reasoning	0.81	0.52

Table 4. **Cross-engine generalization.** Evaluating transfer from MuJoCo (training) to Blender (CLEVRER [59]) demonstrates that ΔYNAMICS maintains its performance in a zero-shot setting despite domain shifts. Incorporating structured motion description consistently improves segmentation map IoU.

	Modality	Segmentation Map IoU (↑) First Frame	Full Sequence
VLM Models			
InternVL3-8B	RGB	0.01	0.02
Qwen2.5-VL-7B	RGB	0.01	0.01
Claude-4-Sonnet	RGB	0.03	0.04
Ours			
ΔYNAMICS	RGB	0.43	0.19
ΔYNAMICS	Opt. Flow	0.63	0.24
+ Motion Reasoning	Opt. Flow	**0.67**	**0.30**



(a) A moving box collides with a static cube; a red ball rolls nearby.



(b) A scene with three rolling balls and one sliding cylinder. The static red box is not reconstructed, as it is not dynamic.

Figure 4. **Zero-shot generalization between engines, from MuJoCo to Blender.** We train ΔYNAMICS on MuJoCo data and evaluate it on CLEVRER [59]. For each example, we show (from top to bottom) (1) the original RGB video, (2) the ground truth optical flow, (3) our model's reconstructed video, and (4) the optical flow of our reconstruction.

enables sampling to explore a broader and effective set of plausible solutions rather than drifting into implausible regions of the parameter space. This broader yet more guided search helps resolve long-tailed errors and yields higher-quality reconstructions.

Lastly, we perform evolutionary search with an initialization from the best-of-32 sample. This method yields the highest accuracy for the full sequence, partly thanks to the quality initialization. This result shows that CMA-ES is the method of choice for optimal accuracy during test-time.

6.2. Real-World Applications

In this section, we evaluate ΔYNAMICSin the real world. Additional results on physically plausible video editing are

Table 5. **Evaluation of test-time optimization strategies on CLEVRER.** We compare the base and motion reasoning variants of ΔYNAMICS with greedy decoding, best-of-32 sampling under temperature of 0.1, and evolutionary search (CMA-ES). *Best@1* denotes the average of the 32 samples, while *Best@32* reports the best.

	Segmentation Map IoU (↑)						**Optical Flow EPE (↓)**					
	First Frame			Full Sequence			First Frame			Full Sequence		
	Greedy	Best@1	Best@32	Greedy	Best@1	Best@32	Greedy	Best@1	Best@32	Greedy	Best@1	Best@32
ΔYNAMICS	0.63	0.63	0.67	0.24	0.24	0.28	3.66	3.65	2.92	6.91	6.86	6.21
+ Motion Reasoning	0.67	0.68	**0.76**	0.30	0.30	0.38	2.92	2.93	2.22	5.94	5.95	5.17
+ + CMA-ES	0.62	-	-	**0.66**	-	-	**0.13**	-	-	**0.11**	-	-

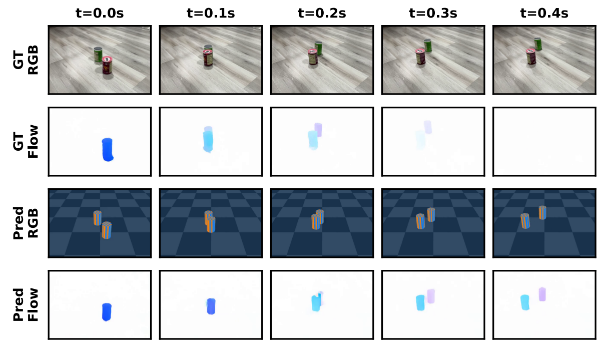
Table 6. **Performance of real-world rigid-body motion reconstruction.** We evaluate ΔYNAMICS on real-world video dataset. ΔYNAMICS successfully generalizes from synthetic training to real scenes. Incorporating motion reasoning improves segmentation and flow alignment, while Best-of-32 sampling further refines accuracy. CMA-ES optimization provides the best full sequence alignment results.

	Segmentation Map IoU (↑)		**Optical Flow EPE (↓)**	
	First Frame	Full Seq.	First Frame	Full Seq.
ΔYNAMICS	0.57	0.26	1.62	0.67
+ Motion Reasoning	0.54	0.29	1.39	0.58
+ + Best@32	**0.72**	0.41	**1.06**	0.46
+ + CMA-ES	0.57	**0.65**	1.26	**0.36**

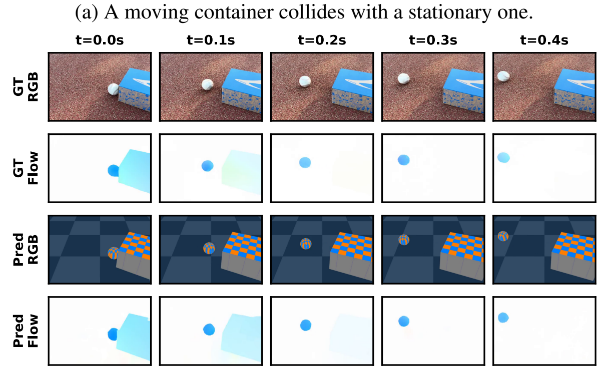
provided in Section E.

Dataset. We collected real-world videos using an iPhone 13 and Canon cameras in landscape orientation. To assess robustness to diverse surface conditions, we collected data for multiple environments, including indoor floors, outdoor running tracks, and basketball courts. Test objects include everyday items such as shoe boxes, balls, massage roll, cookie containers, and some irregular-shaped objects such as apples. The dataset details are provided in Appendix D.1.

Results. As shown in Table 6, the motion reasoning variant improves segmentation IoU by 12% and flow EPE by 13%, while best-of-32 sampling further enhances motion accuracy. Evolutionary search provides the largest gains in the metrics. Qualitatively, Figure 5 shows that our model captures the trajectories and locations of two-object motion precisely, implying a high level of accuracy in the estimated initial states and physics parameters. We provide more real-world examples and failure analysis in Appendix D.2.



(a) A moving container collides with a stationary one.



(b) A sliding shoebox collides with a baseball on a running track.

Figure 5. **Motion capture for real-world videos.** ΔYNAMICS is able to reproduce motion trajectory and object location on real-world surfaces and complex lighting. It can also capture multi-body collision dynamics despite the domain gap between synthetic and real data.

7. Conclusion

We have presented a novel viewpoint on the problem of predicting physics configurations for rigid-body motion from monocular videos. Our main contribution is a general structured textual representation of the physics states and parameters for a wide range of motion dynamics and object interaction. In addition, we trained ΔYNAMICS, a vision–language model to generate physics configurations and camera geometry in a structured textual format. We also incorporate motion reasoning and test-time optimization techniques to enhance our model's accuracy. Being trained on $400K$ synthetically generated scenes in MuJoCo, our model shows robust generalization across rendering engines and to real-world data, and consistently outperforms off-the-shelf vision–language models. Our results demonstrate a promising line of research on using language modeling to provide a common physics representation for physics perception and physics simulation.

References

[1] Anthropic. The claude 3 model family: Opus, sonnet, haiku. https://assets.anthropic.com/m/61e7d27f8c8f5919/original/Claude-3-Model-Card.pdf, 2024. 6, 7

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015. 2

[3] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *ICCV*, pages 6836–6846, 2021. 6, 7

[4] Martin Asenov, Michael Burke, Daniel Angelov, Todor Davchev, Kartic Subr, and Subramanian Ramamoorthy. Vid2param: Modeling of dynamics parameters from video. *IEEE Robotics and Automation Letters*, 5(2):414–421, 2019. 2

[5] Tayfun Ates, M Samil Atesoglu, Cagatay Yigit, Ilker Kesen, Mert Kobas, Erkut Erdem, Aykut Erdem, Tilbe Goksun, and Deniz Yuret. Craft: A benchmark for causal reasoning about forces and interactions. *arXiv preprint arXiv:2012.04293*, 2020. 2

[6] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 5, 7

[7] Fabien Baradel, Natalia Neverova, Julien Mille, Greg Mori, and Christian Wolf. Cophy: Counterfactual learning of physical dynamics. *arXiv preprint arXiv:1909.12000*, 2019. 2

[8] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. *Advances in neural information processing systems*, 29, 2016. 1

[9] Jonas Belouadi, Simone Ponzetto, and Steffen Eger. Detikzify: Synthesizing graphics programs for scientific figures and sketches with tikz. *Advances in Neural Information Processing Systems*, 37:85074–85108, 2024. 2

[10] Jonas Belouadi, Eddy Ilg, Margret Keuper, Hideki Tanaka, Masao Utiyama, Raj Dabre, Steffen Eger, and Simone Paolo Ponzetto. Tikzero: Zero-shot text-guided graphics program synthesis. *arXiv preprint arXiv:2503.11509*, 2025. 2

[11] Siyuan Bian, Chenghao Xu, Yuliang Xiu, Artur Grigorev, Zhen Liu, Cewu Lu, Michael J Black, and Yao Feng. Chatgarment: Garment estimation, generation and editing via large language models. In *CVPR*, pages 2924–2934, 2025. 3

[12] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. 1

[13] Ryan Burgert, Yuancheng Xu, Wenqi Xian, Oliver Pilarski, Pascal Clausen, Mingming He, Li Ma, Yitong Deng, Lingxiao Li, Mohsen Mousavi, et al. Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13–23, 2025. 7, 9

[14] Pradyumna Chari, Chinmay Talegaonkar, Yunhao Ba, and Achuta Kadambi. Visual physics: Discovering physical laws from videos. *arXiv preprint arXiv:1911.11893*, 2019. 2

[15] Yamei Chen, Haoquan Zhang, Yangyi Huang, Zeju Qiu, Kaipeng Zhang, Yandong Wen, and Weiyang Liu. Symbolic graphics programming with large language models. *arXiv preprint arXiv:2509.05208*, 2025. 2

[16] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198, 2024. 2

[17] Wei Chow, Jiageng Mao, Boyi Li, Daniel Seita, Vitor Guizilini, and Yue Wang. Physbench: Benchmarking and enhancing vision-language models for physical world understanding. *arXiv preprint arXiv:2501.16411*, 2025. 2

[18] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv e-prints*, pages arXiv–2409, 2024. 2

[19] Mingyu Ding, Zhenfang Chen, Tao Du, Ping Luo, Josh Tenenbaum, and Chuang Gan. Dynamic visual reasoning by learning differentiable physics models from video and language. *Advances in Neural Information Processing Systems*, 34:887–899, 2021. 2

[20] Sebastien Ehrhardt, Aron Monszpart, Niloy Mitra, and Andrea Vedaldi. Unsupervised intuitive physics from visual observations. In *Asian Conference on Computer Vision*, pages 700–716. Springer, 2018. 1

[21] C Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. Brax–a differentiable physics engine for large scale rigid body simulation. *arXiv preprint arXiv:2106.13281*, 2021. 3

[22] Hiroki Furuta, Kuang-Huei Lee, Shixiang Shane Gu, Yutaka Matsuo, Aleksandra Faust, Heiga Zen, and Izzeddin Gur. Geometric-averaged preference optimization for soft preference labels. *Advances in Neural Information Processing Systems*, 37:57076–57114, 2024. 2

[23] Alejandro Castañeda Garcia, Jan Warchocki, Jan van Gemert, Daan Brinks, and Nergis Tomen. Learning physics from video: Unsupervised physical parameter estimation for continuous dynamical systems. In *CVPR*, pages 27924–27933, 2025. 2

[24] Yunqi Gu, Ian Huang, Jihyeon Je, Guandao Yang, and Leonidas Guibas. Blendergym: Benchmarking foundational model systems for graphics editing. In *CVPR*, pages 18574–18583, 2025. 3

[25] Nikolaus Hansen and Andreas Ostermeier. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In *Proceedings of IEEE international conference on evolutionary computation*, pages 312–317. IEEE, 1996. 5

[26] Eric Heiden, Ziang Liu, Vibhav Vineet, Erwin Coumans, and Gaurav Sukhatme. Learning articulated rigid body dynamics simulations from video. In *ICLR Workshop on the Elements of Reasoning: Objects, Structure and Causality*, 2022. 2

[27] Yuanming Hu, Luke Anderson, Tzu-Mao Li, Qi Sun, Nathan Carr, Jonathan Ragan-Kelley, and Frédo Durand. Difftaichi: Differentiable programming for physical simulation. *arXiv preprint arXiv:1910.00935*, 2019. 3

[28] Zhiao Huang, Yuanming Hu, Tao Du, Siyuan Zhou, Hao Su, Joshua B Tenenbaum, and Chuang Gan. Plasticinelab: A soft-body manipulation benchmark with differentiable physics. *arXiv preprint arXiv:2104.03311*, 2021. 3

[29] Krishna Murthy Jatavallabhula, Miles Macklin, Florian Golemo, Vikram Voleti, Linda Petrini, Martin Weiss, Breandan Considine, Jérôme Parent-Lévesque, Kevin Xie, Kenny Erleben, et al. gradsim: Differentiable simulation for system identification and visuomotor control. *arXiv preprint arXiv:2104.02646*, 2021. 3

[30] James R Kubricht, Keith J Holyoak, and Hongjing Lu. Intuitive physics: Current research and controversies. *Trends in cognitive sciences*, 21(10):749–759, 2017. 1

[31] Peter Kulits, Haiwen Feng, Weiyang Liu, Victoria Abrevaya, and Michael J Black. Re-thinking inverse graphics with large language models. *arXiv preprint arXiv:2404.15228*, 2024. 3

[32] Peter Kulits, Michael J Black, and Silvia Zuffi. Reconstructing animals and the wild. In *CVPR*, pages 16565–16577, 2025. 3

[33] Long Le, Jason Xie, William Liang, Hung-Ju Wang, Yue Yang, Yecheng Jason Ma, Kyle Vedder, Arjun Krishna, Dinesh Jayaraman, and Eric Eaton. Articulate-anything: Automatic modeling of articulated objects via a vision-language foundation model. *arXiv preprint arXiv:2410.13882*, 2024. 5

[34] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 2

[35] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306, 2024. 2

[36] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021. 2

[37] Himangi Mittal, Peiye Zhuang, Hsin-Ying Lee, and Shubham Tulsiani. Uniphy: Learning a unified constitutive model for inverse physics simulation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16208–16218, 2025. 3

[38] Yi-Ling Qiao, Alexander Gao, and Ming Lin. Neuphysics: Editable neural geometry and physics from monocular videos. *Advances in Neural Information Processing Systems*, 35:12841–12854, 2022. 3

[39] Nazneen Fatema Rajani, Rui Zhang, Yi Chern Tan, Stephan Zheng, Jeremy Weiss, Aadit Vyas, Abhijit Gupta, Caiming Xiong, Richard Socher, and Dragomir Radev. Esprit: Explaining solutions to physical reasoning tasks. *arXiv preprint arXiv:2005.00730*, 2020. 2

[40] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 5

[41] Ronan Riochet, Mario Ynocente Castro, Mathieu Bernard, Adam Lerer, Rob Fergus, Véronique Izard, and Emmanuel Dupoux. Intphys 2019: A benchmark for visual intuitive physics understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5016–5025, 2021. 2

[42] Juan A Rodriguez, Abhay Puri, Shubham Agarwal, Issam H Laradji, Pau Rodriguez, Sai Rajeswar, David Vazquez, Christopher Pal, and Marco Pedersoli. Starvector: Generating scalable vector graphics code from images and text. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16175–16186, 2025. 2

[43] Aadarsh Sahoo, Vansh Tibrewal, and Georgia Gkioxari. Aligning text, images, and 3d structure token-by-token. *arXiv preprint arXiv:2506.08002*, 2025. 5

[44] Arsalan Sharifnassab, Saber Salehkaleybar, Sina Ghiassian, Surya Kanoria, and Dale Schuurmans. Soft preference optimization: Aligning language models to expert distributions. *arXiv preprint arXiv:2405.00747*, 2024. 2

[45] Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. Preference ranking optimization for human alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 18990–18998, 2024. 5, 1

[46] Yiren Song, Danze Chen, and Mike Zheng Shou. Layertracer: Cognitive-aligned layered svg synthesis via diffusion transformer. *arXiv preprint arXiv:2502.01105*, 2025. 2

[47] Priya Sundaresan, Rika Antonova, and Jeannette Bohgl. Diffcloud: Real-to-sim from point clouds with differentiable simulation and rendering of deformable objects. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10828–10835. IEEE, 2022. 3

[48] Cheng Tan, Qi Chen, Jingxuan Wei, Gaowei Wu, Zhangyang Gao, Siyuan Li, Bihui Yu, Ruifeng Guo, and Stan Z Li. Sketchagent: Generating structured diagrams from hand-drawn sketches. *arXiv preprint arXiv:2508.01237*, 2025. 2

[49] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, pages 402–419. Springer, 2020. 3, 4, 5, 7

[50] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012. 2, 4

[51] Nicholas Watters, Daniel Zoran, Theophane Weber, Peter Battaglia, Razvan Pascanu, and Andrea Tacchetti. Visual interaction networks: Learning a physics simulator from video. *Advances in neural information processing systems*, 30, 2017. 1

[52] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025. 2

[53] Jiajun Wu, Ilker Yildirim, Joseph J Lim, Bill Freeman, and Josh Tenenbaum. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. *NeurIPS*, 28, 2015. 2

[54] Jiajun Wu, Erika Lu, Pushmeet Kohli, Bill Freeman, and Josh Tenenbaum. Learning to see physics via visual de-animation. *NeurIPS*, 30, 2017. 2

[55] Ronghuan Wu, Wanchao Su, and Jing Liao. Chat2svg: Vector graphics generation with large language models and image diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 23690–23700, 2025. 2

[56] Ximing Xing, Haitao Zhou, Chuang Wang, Jing Zhang, Dong Xu, and Qian Yu. Svgdreamer: Text guided svg generation with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4546–4555, 2024.

[57] Ximing Xing, Qian Yu, Chuang Wang, Haitao Zhou, Jing Zhang, and Dong Xu. Svgdreamer++: Advancing editability and diversity in text-guided svg generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 2

[58] Gengshan Yang, Shuo Yang, John Z Zhang, Zachary Manchester, and Deva Ramanan. Ppr: Physically plausible reconstruction from monocular videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3914–3924, 2023. 3

[59] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*, 2019. 1, 2, 6, 7, 5

[60] Yunzhi Zhang, Zizhang Li, Matt Zhou, Shangzhe Wu, and Jiajun Wu. The scene language: Representing scenes with programs, words, and embeddings. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24625–24634, 2025. 3

[61] Wang Zhao, Yan-Pei Cao, Jiale Xu, Yuejiang Dong, and Ying Shan. Di-pcg: Diffusion-based efficient inverse procedural content generation for high-quality 3d asset creation. In *CVPR*, pages 11061–11072, 2025. 3

[62] Xian Zhou, Yiling Qiao, Zhenjia Xu, TH Wang, Z Chen, J Zheng, Z Xiong, Y Wang, M Zhang, P Ma, et al. Genesis: A generative and universal physics engine for robotics and beyond. *arXiv preprint arXiv:2401.01454*, 2024. 5

[63] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 5, 7