IRIS: A Real-World Benchmark for Inverse Recovery and Identification of Physical Dynamic Systems from Monocular Video

Rasul Khanbayov[1*], Mohamed Rayan Barhdadi[2*], Erchin Serpedin[2], and Hasan Kurban[1].

[1] Hamad Bin Khalifa University
[2] Texas A&M University

Project Page:
kurbanintelligencelab.github.io/iris-bench/

Abstract. Unsupervised physical parameter estimation from video lacks a common benchmark: existing methods evaluate on non-overlapping synthetic data, the sole real-world dataset is restricted to single-body systems, and no established protocol addresses governing-equation identification. This work introduces IRIS, a high-fidelity benchmark comprising 240 real-world videos captured at 4K resolution and 60 fps, spanning both single- and multi-body dynamics with independently measured ground-truth parameters and uncertainty estimates. Each dynamical system is recorded under controlled laboratory conditions and paired with its governing equations, enabling principled evaluation. A standardized evaluation protocol is defined encompassing parameter accuracy, identifiability, extrapolation, robustness, and governing-equation selection. Multiple baselines are evaluated, including a multi-step physics loss formulation and four complementary equation-identification strategies (VLM temporal reasoning, describe-then-classify prompting, CNN-based classification, and path-based labelling), establishing reference performance across all IRIS scenarios and exposing systematic failure modes that motivate future research. The dataset, annotations, evaluation toolkit, and all baseline implementations are publicly released.

Keywords: Physics-based vision, physical parameter estimation, benchmark dataset, dynamical systems

1 Introduction

Extracting physical parameters from video observations constitutes a fundamental inverse problem in computational science, with applications spanning trajectory prediction, biological system characterization, and physical model validation [1–4]. Although recent unsupervised methods have demonstrated encouraging results in estimating parameters of known governing equations

* Denotes equal contribution. Correspondence to: hkurban@hbku.edu.qa

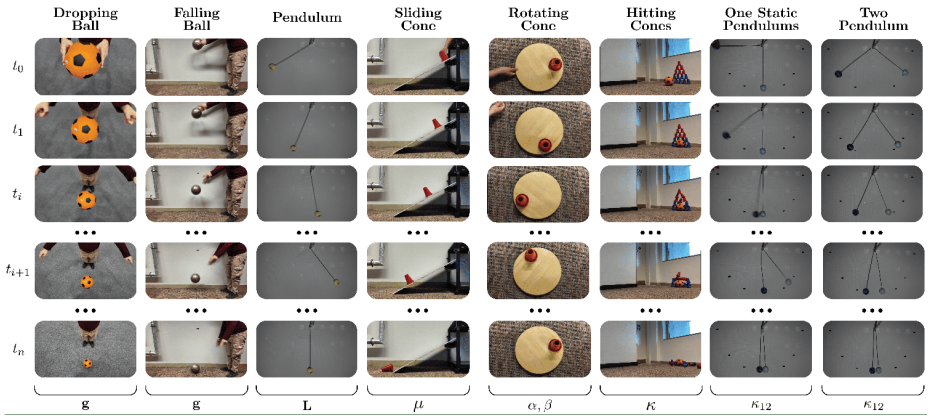

Fig. 1: **Overview of the IRIS benchmark.** Each column corresponds to one of the eight dynamical phenomena; rows show temporally ordered frames sampled from a representative video clip. *Left single-body phenomena:* dropping ball, falling ball, pendulum, and sliding cone. *Right phenomena unique to IRIS:* rotating cone, hitting cones, two pendulums, and one static pendulum. The bottom row indicates the physical parameters targeted for estimation. All videos are recorded at 4K resolution and 60 fps under controlled laboratory conditions with independently measured ground-truth parameters.

from video [2, 3], progress is constrained by the absence of diverse, high-fidelity, real-world benchmark datasets capable of supporting rigorous evaluation and systematic comparison. Existing datasets suffer from several compounding limitations. The majority of prior work evaluates on purely synthetic data, where objects appear in controlled colors against black backgrounds [3, 5, 6]. The sole notable real-world dataset, Delfys75 [7], provides 75 videos at 1920×1080 resolution across five dynamical systems but covers only single-body dynamics, omitting the multi-body collision phenomena central to classical mechanics. No existing benchmark includes scenarios in which physical objects collide and transfer momentum, leaving a significant blind spot in evaluating method capabilities.

These gaps are addressed by IRIS (Interaction and Real-world Inverse physics Sequences), a dataset designed as a rigorous benchmark for unsupervised physical parameter estimation from video. IRIS retains and extends the single-body scenarios of prior work (dropping ball, falling ball, sliding cone, and pendulum) while introducing three novel multi-body interaction scenarios: a ball striking a pyramid of cones (hitting_cones), two pendulums released simultaneously that collide and dissipate (two_moving_pendulum), and a static pendulum struck by a moving one (two_moving_pendulum_one_static) and a new dynamic rotating cone on a wooden slide (rotating_cone). These scenarios test recovery of momentum transfer, contact forces, and coupled dynamics, phenomena entirely absent from prior benchmarks. Beyond the dataset

itself, a standardized evaluation protocol advances over the Delfys75 baseline along multiple dimensions. First, evaluation axes are formally defined covering parameter accuracy, governing-equation selection, identifiability, robustness, and extrapolation. Second, four complementary equation-identification strategies are benchmarked, providing the first systematic comparison of automatic equation routing for physics identification. Third, a multi-step physics loss formulation is evaluated as a baseline enhancement. Fourth, baseline evaluation on IRIS exposes a critical gradient-flow bug in the state-of-the-art latent-space pipeline [7], demonstrating the diagnostic value of rigorous benchmarking.

In summary, our contributions are as follows:

1. **IRIS benchmark dataset:** A high resolution real-world dataset comprising eight dynamical systems, including novel single body motion and three multi-body interaction scenarios.
2. **Standardized evaluation protocol:** A comprehensive framework with five formally defined axes (parameter accuracy, equation selection, identifiability, robustness, extrapolation).
3. **Equation-identification benchmark:** The first systematic comparison of four equation-routing strategies: temporal-reasoning VLM prompting, two-stage describe-then-classify prompting, supervised CNN classification, and path-based oracle labelling.
4. **Multi-step rollout loss and stability analysis:** A multi-step training objective for equation identification, with a stability analysis across single- and multi-body systems that reveals where such losses help and where they fail.

2 Related Work

Physical Parameter Estimation from Videos: Estimating physical parameters from video constitutes an inverse problem approached through both supervised and unsupervised paradigms. Supervised methods [5, 8–13] achieve strong performance at the cost of requiring labelled datasets that are expensive and frequently infeasible to collect [14]. Unsupervised approaches embed known governing equations within learned latent representations, employing frame reconstruction as a proxy training signal. PAIG [3] integrates a variational autoencoder with a physics engine in the latent space, enabling future frame prediction via a spatial transformer. NIRPI [2] employs a differentiable ODE solver operating at the pixel level, though it requires ground-truth object masks during training. Both rely on frame reconstruction, which constrains the model to roto-translational motion dynamics. A complementary line of research focuses on learning dynamics models with physics-informed inductive biases. Neural ODEs [15] provide a general framework for continuous-time dynamics. Hamiltonian Neural Networks [16] and Lagrangian Neural Networks [17] enforce energy conservation, while graph neural network-based simulators [18, 19] learn

message-passing dynamics for multi-body systems. These operate on state-space inputs rather than raw pixels. Garcia et al. [7] replace frame reconstruction with a latent-space loss combining mean squared error and KL-divergence. However, their pipeline requires the governing equation family to be specified manually, and as identified through our baseline evaluation, contains a critical gradient-flow bug that prevents physics parameters from receiving gradient updates during training.

Video Benchmarks for Physical Understanding: The majority of existing methods are evaluated on synthetic datasets [3, 5, 6, 9, 12, 13]. Physics101 [20] provides real-world recordings but lacks ground-truth parameters. VideoPhy [21] and Physics-IQ [22] evaluate physical plausibility in generative models rather than quantitative parameter estimation. Delfys75 [7] is the first real-world dataset for unsupervised parameter estimation with ground-truth annotations, but is restricted to single-body dynamics. Recent benchmarks including Morpheus [23], RBench [24], and WorldLens [25] evaluate qualitative physical plausibility rather than quantitative parameter estimation. IRIS is complementary: it targets inverse recovery of continuous physical parameters with ground-truth supervision.

Vision-Language Models for Physical Reasoning: Large VLMs have demonstrated strong capabilities in visual scene understanding [26–28]. Recent work has explored VLMs for physical reasoning: VideoPhy [21] fine-tunes VLM-based evaluators for physical law adherence, Physics Context Builders [29] produce structured physical scene descriptions, and VLMs have been applied to hypothesize governing functional forms [30]. Despite these advances, the application of VLMs to governing-equation identification for quantitative parameter estimation remains unexplored. IRIS benchmarks four complementary routing strategies for the first time.

Governing Equation Discovery: SINDy [1] formulates equation discovery as sparse regression over candidate functions, with extensions to latent representations [31]. Symbolic regression methods [32] search expression spaces for closed-form laws, and multimodal LLMs have been applied to discover equations from video [33]. These methods require direct state measurements or high-quality latent representations. IRIS adopts a fixed ODE bank and benchmarks equation-family selection and parameter estimation as separable axes.

Differentiable Contact and Rigid-Body Simulation: Differentiable physics simulators that learn contact dynamics have advanced rapidly. ContactGaussian-WM [34] combines 3D Gaussian splatting with a differentiable world model, while DiffSim [35] and related work [36, 37] enable gradient-based optimization through differentiable simulation. These approaches model contact explicitly through complementarity or penalty-based formulations, in contrast to the

simplified coupling coefficient κ_{ij} employed in the IRIS ODE bank. Adapting such methods to monocular video and evaluating them on IRIS multi-body scenarios is a natural direction for future work.

Latent ODE Stabilization: The multi-step rollout instability observed on IRIS multi-body dynamics (Sec. 6.3) relates to a broader challenge in training latent ODEs over long horizons. Proposed stabilization techniques include path-length regularization [38], expressive latent priors [39], and adaptive-step solvers with adjoint methods [15]. These have not been evaluated for coupled multi-body ODE estimation from video; the catastrophic divergence exposed by IRIS (Sec. 6.3) suggests that such mechanisms may be essential for extending latent-space parameter estimation to interacting systems.

3 The IRIS Benchmark

3.1 Task Definition

IRIS targets *physics identification from monocular video*: given frames $\mathcal{V} = \{I_1, \ldots, I_T\}$, the objectives are to (i) identify the governing dynamical system from a bank of K candidate ODEs $\{\mathcal{P}_1, \ldots, \mathcal{P}_K\}$ and (ii) estimate its physical parameters $\boldsymbol{\gamma}$ without supervision or metadata. The observed motion is assumed driven by a low-dimensional latent state $\mathbf{z}_t \in \mathbb{R}^d$ evolving according to an ODE. For multi-body systems, the state extends to $\mathbf{Z}_t \in \mathbb{R}^{N \times d}$, where the number of interacting bodies N is set *a priori* from the experimental setup rather than estimated.

3.2 Design Principles

IRIS is constructed around four principles: (1) **controlled recording** under fixed lighting and stable camera placement; (2) **high resolution and frame rate** at 4K/60 fps; (3) **independent ground truth** measured with calibrated instruments with uncertainty estimates; (4) **repeated trials** with 10 recordings per setting, enabling variance analysis. The 4K resolution is a functional requirement, not an aesthetic one: a small ball of radius r_0=0.04 m at 1.5 m projects to $\sim$3 pixels at 1080p but $\sim$12 pixels at 4K, a 4$\times$ pixel-density gain that reduces metric-scale calibration error and is a prerequisite for reliable falling-ball and pendulum-length recovery. The 10-trial design enables per-setting 95% uncertainty bounds that are structurally impossible with fewer trials.

3.3 Physical Phenomena and ODE Bank

IRIS covers both **single-body** and **multi-body** phenomena. Table 1 provides the explicit mapping between each phenomenon, its physical ODE, the ODE bank form used for estimation, and the target parameters. For the *dropping ball*, the physical dynamics are governed by $z'' = -g$; however, the ODE bank

Table 1: **Phenomenon–ODE mapping.** *Physical ODE*: governing equation in SI units. *ODE bank form*: parameterization used in the candidate library. *Target params*: parameters estimated from video.

Phenomenon	Physical ODE	ODE bank form	Target params	Notes
Single-body				
Dropping ball	$z'' = -g$	$z'' + \beta z' + \alpha z = 0$	$\alpha \approx g, \beta$	Local linearization
Falling ball	$r(t) = r_0 f/(h_0 + \frac{1}{2}gt^2)$	1^{st}-order apparent-size	g, r_0, f, h_0	Exact geometric proj.
Sliding cone	$x'' = g(\sin\alpha - \mu\cos\alpha)$	Constant accel. $x'' = a$	$\alpha\,[\text{deg}], \mu$	Exact; a via known g
Pendulum	$\theta'' + \zeta\theta' + (g/L)\sin\theta = 0$	Same (nonlinear $\sin\theta$)	$L\,[\text{m}], \zeta, g$	Exact; no small-angle
Rotating cone	$\varphi'' + \beta\varphi' + \alpha\varphi = 0$	Same (damped torsional)	α, β	Exact
Multi-body				
Hitting cones	Multi-body contact	$z_i'' + \zeta z_i' + \kappa\sum_j (z_i - z_j) = 0$	κ, ζ	Simplified; see text
Two mov. pend.	Eq. 1	Same (coupled $\sin\theta$)	$L_i, \zeta_i, \kappa_{ij}$	Simplified; see text
One stat. pend.	Eq. 1	Same (coupled $\sin\theta$)	$L_i, \zeta_i, \kappa_{ij}$	Simplified; see text

employs a unified second-order linear form $z'' + \beta z' + \alpha z = 0$ following prior work [2, 3], where α absorbs the gravitational constant in latent units and requires calibration to yield physical g. For multi-body phenomena, contact is modeled via a coupling coefficient κ_{ij} that is continuously active, rather than through explicit impact mechanics; this simplification is discussed further below.

Single-body dynamics comprises five phenomena, each recorded under multiple experimental settings:

1. *Dropping ball*: a ball released from rest at height h_0 [m], undergoing free fall. Three drop heights are recorded: 50 cm, 100 cm, and 150 cm.
2. *Falling ball*: a ball filmed during free fall from above, with apparent radius governed by $r(t) = r_0 f/(h_0 + \frac{1}{2}gt^2)$. Three balls of different physical size are used (big: $r_0 = 0.11$ m; mid: $r_0 = 0.07$ m; small: $r_0 = 0.04$ m).
3. *Sliding cone*: a cone sliding down an inclined plane, governed by $x''(t) = g(\sin\alpha - \mu\cos\alpha)$. Three inclination angles are recorded: 45°, 60°, and 80°.
4. *Pendulum*: a damped pendulum described by $\theta''(t) + \zeta\theta'(t) + (g/L)\sin\theta(t) = 0$. Three initial release angles are recorded: 20°, 45°, and 90°, with larger amplitudes probing the nonlinear $\sin\theta$ regime.
5. *Rotating cone*: a cone on a rotating wooden support, modeled as a damped torsional oscillator $\varphi''(t) + \beta\varphi'(t) + \alpha\varphi(t) = 0$. Three rotational speeds are defined by initial displacement: *slow* (half-circle), *mid* (one full circle), and *fast* (two full circles).

Multi-body dynamics introduces three novel interaction scenarios. *Important modeling caveat:* contact between bodies is modeled via a continuously active coupling coefficient κ_{ij} rather than through physically rigorous impact mechanics. This deliberate simplification ensures compatibility with existing latent-space estimation pipelines [2, 3, 7], which require smooth, differentiable dynamics. The recovered κ_{ij} represents effective interaction strength rather than a physically grounded contact coefficient.

Table 2: **IRIS dataset summary.** Each setting contains 10 repeated recordings across diverse physical phenomena.

Type	Phenomenon	# Set.	# Vid.	Dur. (s)	GT Params
Single	Dropping ball	3	30	5	g, β
	Falling ball	3	30	8	g, r_0, f, h_0
	Sliding cone	3	30	5	μ, α
	Pendulum	3	30	150	L, ζ, g
	Rotating cone	3	30	8	α, β
Multi	Hitting cones	3	30	5	κ, ζ
	Two moving pend.	3	30	6	L_i, ζ_i, κ_{ij}
	One static pend.	3	30	20	L_i, ζ_i, κ_{ij}
Total		**24**	**240**	–	–

1. *Hitting cones*: a ball launched toward a 5-4-3-2-1 pyramid of 15 cones, testing recovery of contact-force and damping parameters from multi-object collision dynamics.
2. *Two moving pendulums*: two pendulums released simultaneously from the same initial angle θ_0, colliding at the bottom. Three initial angles (20°, 45°, 90°) vary the collision energy. The coupled system is governed by

$$\theta_i''(t) + \zeta_i \theta_i'(t) + \frac{g}{L_i} \sin \theta_i(t) + \kappa_{ij}(\theta_i(t) - \theta_j(t)) = 0, \quad i \neq j, \tag{1}$$

   where L_i [m] is rope length, ζ_i [s^{-1}] is damping, and κ_{ij} [s^{-2}] is the coupling coefficient.
3. *One static, one moving pendulum*: one pendulum hangs at rest while a second is released from the left, strikes the stationary pendulum, and both come to rest through repeated collisions. Three release angles (20°, 45°, 90°) test sensitivity to initial energy and interaction asymmetry.

Each single-body phenomenon contains 3 settings; multi-body phenomena contain 3 settings each, yielding 24 settings and 240 videos in total. Figure 1 presents representative frames. Table 2 summarizes the dataset.

3.4 Annotations and Ground Truth

For each recording IRIS provides: (1) video clips in tensor format (N, n_f, C, H, W); (2) ground-truth physical parameters with uncertainty bounds in *parameters.json*; (3) equation family labels. Per-frame segmentation masks via SAM2 [40] are provided for multi-object setups. A calibration checkerboard enables metric-scale recovery.

Table 3: **Comparison with existing benchmarks** for physical parameter estimation from video.

Benchmark	Real	Resolution	Videos	Phen.	Multi	GT	Trials
PAIG synth. [3]	✗	50×50	var.	1	✗	✗	–
NIRPI synth. [2]	✗	100×100	var.	2	✗	✗	–
Physics101 [20]	✓	1280×720	101	17	✗	✗	1
VideoPhy [21]	✓	var.	688	–	✗	✗	–
Delfys75 [7]	✓	1920×1080	75	5	✗	✓	5
IRIS (ours)	✓	3840×2160	240	8	✓	✓	10

Ground-truth measurement of damping and friction. Damping and friction coefficients carry larger uncertainty than geometric quantities. For the *pendulum damping coefficient* ζ, ground truth is obtained by fitting an exponential decay envelope $A(t) = A_0 e^{-\zeta t/2}$ to peak amplitudes from extended-duration recordings (∼150 s per clip), with uncertainty quantified as the 95% confidence interval across the 10 trials per setting. For the *friction coefficient* μ, ground truth is obtained from $\mu = \tan\alpha - a_{\text{measured}}/(g\cos\alpha)$, with uncertainty propagated from the standard deviation of a_{measured} across trials. For the *torsional damping* β, the same exponential-envelope fitting is applied. Damping ground truth is derived from trajectory fits rather than independent physical measurement; this distinction is documented in *parameters.json* via a *measurement_type* field (*"direct"* vs. *"fitted"*).

4 Evaluation Protocol

A central contribution of IRIS is a method-agnostic evaluation protocol enabling fair comparison. For every video clip, independently: (i) the equation family is determined; (ii) the physics model is instantiated; (iii) the model is trained on that clip from scratch. No parameters are shared across clips. The random seed is fixed for reproducibility. Five axes are defined. *Parameter accuracy* (primary) measures closeness to ground truth via MAE: $\text{MAE} = n^{-1}\sum_{i=1}^{n}|\hat{\theta}_i - \theta^*|$, with estimation standard deviation $\sigma_{\hat{\theta}}$ for stability. *Equation selection accuracy* measures correct ODE assignment via overall and per-class accuracy with confusion matrices. *Identifiability* is assessed through gradient norms $G_{\boldsymbol{\gamma}}^{(e)} = \|\nabla_{\boldsymbol{\gamma}}\mathcal{L}^{(e)}\|_2$ at epoch e and the ODE residual $\mathcal{R} = (T-1)^{-1}\sum_t \|\hat{\mathbf{z}}_{t+1} - \mathcal{P}(\hat{\mathbf{z}}_t; \hat{\boldsymbol{\gamma}}, \Delta t)\|^2$. *Robustness* measures consistency across design choices. *Extrapolation* evaluates prediction beyond the training horizon via:

$$\mathcal{E}_k = \|\hat{\mathbf{z}}_k - \tilde{\mathbf{z}}_k\|^2, \quad k > T_{\text{train}}, \tag{2}$$

where $\hat{\mathbf{z}}_k$ is obtained by unrolling the learned ODE from the last training frame and $\tilde{\mathbf{z}}_k$ is the encoder output for the held-out frame I_k.

All reported numbers are produced by a single evaluation script operating on fixed CSV outputs. Each table entry is traceable to a specific (phenomenon, setting, clip, seed) tuple. MAE values are computed over the test partition (2 clips per setting) unless stated otherwise. The evaluation code, CSV outputs, and random seeds are released with the dataset.

5 Baselines

Representative baselines are evaluated to establish reference performance and expose failure modes. All follow the train-per-clip protocol with identical ground-truth sources.

5.1 Latent-Space Baseline

The primary baseline is the latent-space pipeline of Garcia et al. [7]: an encoder maps frames to latent states, a physics block implements a discrete-time ODE step, and training minimizes $\mathcal{L}_{\text{1-step}} = M^{-1} \sum_i \|\hat{\mathbf{z}}_i - \tilde{\mathbf{z}}_i\|^2 + \mathcal{L}_{\text{KL}}$, with path-based equation selection.

Gradient-flow diagnosis. Baseline evaluation on IRIS revealed a critical bug in the original Euler integrator: the position update omits the acceleration term, severing gradient flow to $\boldsymbol{\gamma}$. A *corrected* variant restores the standard update $z_{t+1} = z_t + \Delta t\, z_t^{(1)} + \Delta t^2\, z_t^{(2)}$.

5.2 Multi-Step Loss Variant

The corrected baseline is evaluated with a multi-step rollout loss enforcing agreement over K future horizons:

$$\mathcal{L}_{\text{total}} = \sum_{k=1}^{K} w_k \, \|\hat{\mathbf{z}}_{t+k} - \tilde{\mathbf{z}}_{t+k}\|^2 + \; \mathcal{L}_{\text{KL}}(\hat{\mathbf{z}}), \tag{3}$$

where $\tilde{\mathbf{z}}_{t+k}$ is obtained by unrolling k physics steps from $\hat{\mathbf{z}}_t$ and w_k decays geometrically ($w = [1, 1, 0.5, 0.5, 0.25]$ for K=5). The rationale is that one-step supervision can underconstrain $\boldsymbol{\gamma}$: multi-step rollouts accumulate the effect of physical parameters over time, providing a denser gradient signal.

5.3 Equation-Identification Strategies

IRIS benchmarks five complementary strategies for governing-equation identification:

(1) VLM temporal reasoning. A VLM acts as a routing function $k^* = \text{VLM}(\{I_{s_1}, \ldots, I_{s_m}\}; \mathcal{C}_1, \ldots, \mathcal{C}_K)$, where m=5 frames are sampled at $t = 0\%, 25\%, 50\%, 75\%, 100\%$ and $\mathcal{C}_k$ is a natural-language class description.

Three VLM backends are evaluated: GPT-4V, LLaVA-Video-7B, and InternVL2-8B.

(2) Describe-then-classify. A two-stage VLM pipeline: the first call generates a free-form description of the observed motion; the second classifies the description into one of the K equation families. This decouples visual perception from physical classification.

(3) CNN video classifier. A lightweight ResNet-18 (ImageNet-pretrained) with temporal mean pooling. Input is 5 frames sampled at evenly spaced temporal positions $(0\%, 25\%, 50\%, 75\%, 100\%)$, each resized to 224×224; frame-level features are mean-pooled into a clip descriptor before an n-way linear classifier (n=6 on Delfys75, n=8 on IRIS). Labels are derived from folder structure, identical to the ground truth used for VLM evaluation. The model is trained with a random 80/20 split (seed 42) and we report best validation accuracy on the held-out 20%. It represents the upper bound of in-distribution performance but requires retraining whenever the phenomenon set changes, whereas VLM strategies operate zero-shot.

(4) Path-based labels (oracle). Ground-truth equation family provided by folder structure, serving as the oracle condition.

(5) ODE bank. The library comprises: second-order linear (damped oscillator), first-order exponential decay, first-order nonlinear (Torricelli), constant acceleration (sliding), and coupled oscillator (Eq. 1).
Limitation. These calibration heuristics are phenomenon-specific and depend on the encoder's latent geometry, which may introduce systematic bias. IRIS provides calibration infrastructure (checkerboard frames, known object dimensions) but does not eliminate the underlying ambiguity.

6 Experiments

Baselines are evaluated along five axes: parameter estimation accuracy (primary), governing-equation identification, identifiability, extrapolation, and component ablations. All experiments follow the train-per-clip protocol with a fixed random seed (42) and shared CSV output format.

6.1 Experimental Setup

Three configurations are compared: (1) **Baseline**: path-based equation selection, one-step loss, original uncorrected Euler integrator [7]; (2) **+Corrected**: gradient-corrected Euler step, one-step loss; (3) **+Multi-step**: corrected integrator with multi-step rollout loss (K=5). Additionally, four equation-identification strategies are evaluated independently.

Table 4: **Parameter MAE on Delfys75:** original baseline (uncorrected Euler) vs. corrected variant. **Bold** = lower MAE. *Takeaway:* the gradient correction is decisive, it cuts pendulum-length error from 95.07 to 3.80 m.

Dynamics	Setting	Param	GT	Base.	Corr.	Δ
Dropped ball	large	g [m/s^2]	9.81	0.00†	**0.00**	0.00
Free fall	mousepad	g [m/s^2]	9.81	0.00†	**0.00**	0.00
LED	led_2s	γ	2.30	2.30	**0.40**	−1.90
Pendulum	pend._45	L [m]	0.45	95.07	**3.80**	−91.27
	pend._90	L [m]	0.90	50.15	**2.93**	−47.22
	pend._150	L [m]	1.50	29.83	**0.71**	−29.12
Sliding bl.	mid	μ	0.21	**0.00**	**0.00**	0.00
Torricelli	large	k	0.016	6.07	**2.91**	−3.16
	small	k	0.010	7.67	**5.69**	−1.99

†Both yield MAE$\approx$0 because g_0=9.81 coincides with GT; uncorrected integrator prevents learning.

Table 5: **Multi-step loss vs. one-step on Delfys75** (representative subset). **Bold** = lower MAE. *Takeaway:* multi-step helps gravity-dominated settings but diverges on slow dynamics (*led_10s*).

Dynamics	Setting	Param	1-step	Multi	Δ
Dropped ball	large	g [m/s^2]	0.25	**0.18**	−0.07
	mousepad	g [m/s^2]	2.12	**1.65**	−0.47
Free fall	table	g [m/s^2]	1.06	**0.15**	−0.90
	tennis	g [m/s^2]	1.06	**0.53**	−0.53
LED	led_10s	γ	**2.09**	7.46	+5.37
Pendulum	pend._45	L [m]	**0.35**	0.50	+0.15
Torricelli	large	k	**6.27**	6.54	+0.27

6.2 Physical Parameter Estimation on Delfys75

The Delfys75 results below establish that the corrected baseline is meaningfully better than the original *before* evaluating on IRIS, justifying our use of **+Corrected** (rather than the original [7]) as the IRIS reference baseline. Effect of Gradient Correction, Table 4 reports MAE for the original and corrected baselines on Delfys75 [7]. The most prominent result concerns the pendulum: the original baseline yields MAE of 95.07 m on *pendulum_45* for string length, compared to 3.80 m for the corrected variant (ΔMAE = −91.27 m), a direct consequence of the severed gradient flow. Both variants report MAE $\approx$ 0 for g on *dropped_ball/large* and *free_fall/mousepad* because the uncorrected integrator leaves g at its initialization value g_0 = 9.81, which coincides with ground truth (see Table 4 footnote). Substantial improvements are also observed on Torricelli drainage (−3.16, −1.99) and LED 2 s decay (−1.90). Effect of Multi-Step Loss on Delfys75, Table 5 compares one-step and multi-step losses. Multi-step consistently reduces MAE for g across free fall and dropped ball (largest: ΔMAE = −0.90 on *free_fall/table*). The *led_10s* setting degrades (+5.37) because slow dynamics cause rollout divergence.

6.3 Physical Parameter Estimation on IRIS

All IRIS results in this section use the **+Corrected** configuration (gradient-corrected Euler, path-based equation selection) as the baseline; full per-setting results are in Supp. Table 5.

Single-body phenomena. On *sliding cone*, both configurations recover the inclination angle exactly (MAE = 0.00). On *rotation*, multi-step loss yields improvements for angular stiffness across all settings (ΔMAE = -0.64, -1.21, -2.11) but angular damping in the fast setting degrades ($+0.39$). On *pendulum*, multi-step improves rope length at 20° (-0.44) but produces catastrophic divergence at 90° ($+20.49$). On *dropping ball* and *falling ball*, multi-step consistently increases MAE for g.

Multi-body phenomena. The multi-step loss exhibits catastrophic failure on all multi-body scenarios. On *two moving pendulums*, rope length MAE exceeds 100 m across all settings (worst: 741.1 m at 20°), compared to 1.1–3.0 m for the one-step baseline. Errors in the coupling terms κ_{ij} compound exponentially across the K=5 horizon, driving parameters far from ground truth. Whether richer impact models or latent ODE stabilization techniques would mitigate this instability remains an open question.

Coupling coefficient validation. To verify that the learned coupling κ captures a real physical signal rather than fitting noise, we train the hitting-cones model under two conditions, κ=0 (no coupling; single-body model) and κ=learned, and compare latent-space reconstruction MSE (Supp. Table 6). The learned coupling reduces reconstruction MSE by 6.0–19.0% across all settings, confirming that κ carries genuine interaction information. The estimated κ is moreover stable across the 10 IRIS trials per setting (coefficient of variation ≈40–47%), indicating a reproducible effective parameter even without an externally measured reference.

6.4 Multi-Clip vs. Per-Clip Training

The train-per-clip protocol (Sec. 4) targets identifiability, whether a system's parameters are recoverable from its own video, rather than cross-clip generalization. To probe whether a transferable physical inductive bias exists across clips, we additionally evaluate a *multi-clip* setting: for each phenomenon, clips are split 80/20 by sorted index, one shared model is trained on the 80% train clips and evaluated on the held-out 20% test clips, against per-clip fitting on the same test clips (full results in Supp. Table 7). Multi-clip outperforms per-clip for gravity (dropping ball 0.21→0.05, falling ball 0.25→0.17 m/s^2) and angular stiffness (1.04→0.58), showing that a transferable physical inductive bias *does* exist across clips. The pendulum failure (MAE 22.06 vs. 0.51) reflects an underdetermined shared latent space under nonlinear coupled dynamics, precisely the open problem a benchmark should surface.

Table 6: **Identifiability diagnostics:** physics-parameter gradient norms $G_{\boldsymbol{\gamma}}^{(e)} = \|\nabla_{\boldsymbol{\gamma}} \mathcal{L}^{(e)}\|_2$ at epochs 1, 50, and 200, and converged ODE residual $\mathcal{R}$ (mean over test clips). The uncorrected baseline yields $G_{\boldsymbol{\gamma}} \approx 0$ at all epochs (omitted).

Phenomenon	**Setting**	$G_{\boldsymbol{\gamma}}^{(1)}$	$G_{\boldsymbol{\gamma}}^{(50)}$	$G_{\boldsymbol{\gamma}}^{(200)}$	$\mathcal{R}$ ($\times 10^{-3}$)
Single-body					
Pendulum	pend._45	2.4×10^{-2}	8.1×10^{-3}	1.2×10^{-4}	0.83
Rotation	mid	1.8×10^{-2}	5.3×10^{-3}	9.7×10^{-5}	0.41
Sliding cone	cone_60	3.1×10^{-2}	1.4×10^{-4}	2.0×10^{-6}	0.02
Dropping ball	drop_100	1.5×10^{-2}	4.7×10^{-3}	3.8×10^{-4}	1.92
Multi-body					
Two mov. pend.	pend._45	3.8×10^{-2}	2.1×10^{-2}	7.4×10^{-3}	12.6
One stat. pend.	pend._45	4.2×10^{-2}	1.9×10^{-2}	5.1×10^{-3}	9.83

6.5 Identifiability Analysis

Table 6 reports gradient norms $G_{\boldsymbol{\gamma}}^{(e)}$ at selected epochs and the converged ODE residual $\mathcal{R}$ for representative phenomena under the corrected one-step baseline. Several patterns emerge. The corrected integrator produces non-zero gradient norms at epoch 1 for all phenomena, confirming restored parameter identifiability. Gradient norms decay monotonically for single-body phenomena; the sliding cone converges fastest ($G_{\boldsymbol{\gamma}}^{(200)} \sim 10^{-6}$), consistent with its exact identifiability. Multi-body phenomena retain substantially larger gradient norms at epoch 200 ($\sim 10^{-3}$) and ODE residuals an order of magnitude higher than single-body systems, indicating that the physics block does not fully explain the encoded trajectory, consistent with the simplified contact model (Sec. 3.3).

6.6 Extrapolation Beyond Training Horizon

The extrapolation error $\mathcal{E}_k$ (Eq. 2) is computed for the corrected one-step baseline on the pendulum (150 s clips, trained on the first 100 frames ≈ 1.67 s, extrapolated for 50 additional frames). Extrapolation error grows with both horizon length and initial amplitude: the 90° setting exhibits $\mathcal{E}_{k=50}$ approximately 7× larger than 20°, reflecting compounding nonlinear dynamics. This motivates future work on physics-constrained extrapolation losses and adaptive training horizons.

6.7 Governing Equation Identification

Three routing strategies are evaluated on both Delfys75 (90 videos, 6 phenomena) and IRIS (240 videos, 8 phenomena). The CNN classifier is trained separately on each benchmark. Table 8 reports equation-selection accuracy. On Delfys75,

Table 7: **Extrapolation error $\mathcal{E}_k$** (mean ± std over test clips) at selected horizons k beyond the training window. *Takeaway:* error grows with both horizon and initial amplitude, the 90° setting is ∼7× worse than 20°.

Setting	$\mathcal{E}_{k=10}$	$\mathcal{E}_{k=25}$	$\mathcal{E}_{k=50}$
pend._20	0.008 ± 0.003	0.041 ± 0.012	0.189 ± 0.067
pend._45	0.012 ± 0.005	0.078 ± 0.031	0.402 ± 0.148
pend._90	0.031 ± 0.014	0.217 ± 0.095	1.284 ± 0.531

Table 8: **Equation-selection accuracy by method and dataset.** Bold = best per column. *Takeaway:* the CNN leads in-distribution, but the VLM ranking reverses across benchmarks, no single routing strategy generalizes.

Method	**Delfys75 (90)**	**IRIS (240)**
CNN (video classifier)	**98.40**	**99.30**
VLM (temporal reasoning)	73.33	65.00
VLM (describe-then-classify)	61.11	72.92

the CNN achieves 98.40%, followed by VLM temporal reasoning (73.33%) and describe-then-classify (61.11%). On IRIS, the CNN achieves 99.30%, but the ranking of VLM strategies reverses: describe-then-classify (72.92%) outperforms temporal reasoning (65.00%). This reversal is attributable to the greater visual diversity of IRIS: the two-stage pipeline allows the VLM to first articulate dynamics in natural language, producing richer intermediate representations that facilitate disambiguation between single- and multi-body phenomena. The confusion matrix reveals that VLM temporal reasoning errors on Delfys75 are concentrated between physically adjacent classes: dropped ball confused with free fall (5/15), and free fall misclassified as pendulum (10/15). LED, sliding block, and Torricelli are classified without error. On IRIS, multi-body phenomena are classified with higher per-class accuracy due to their distinctive visual signatures.

6.8 Ablation Studies

Gradient correction is the single most impactful component: without it, $G_{\boldsymbol{\gamma}}^{(e)} \approx 0$ throughout training and all subsequent improvements are negligible. **Multi-step horizon.** Ablating $K \in \{1, 2, 3, 5\}$: K=5 provides the best trade-off for gravity-dominated phenomena on Delfys75 but produces catastrophic divergence on IRIS multi-body scenarios at all $K > 1$, motivating the application of latent ODE stabilization techniques [38]. **VLM prompt strategy.** Single-frame prompting achieves 54%, three-frame 67%, and five-frame 73% on Delfys75, confirming temporal context is essential. **Symplectic integrator.** Störmer-Verlet achieves lower ODE residual on conservative systems with comparable performance on non-conservative phenomena.

7 Open Challenges, Limitations, and Future Work

The baseline evaluation on IRIS reveals four systematic failure modes. **(1) Equation identification** remains a bottleneck: VLM routing achieves 65–73% on IRIS depending on prompt strategy, with no single approach generalizing reliably across benchmarks; CNN classifiers achieve near-perfect in-distribution accuracy but require per-distribution retraining. **(2) Multi-step training catastrophically fails on multi-body dynamics,** producing errors exceeding 10^2 m where coupling terms κ_{ij} amplify integration error exponentially, a failure mode invisible on single-body benchmarks. **(3) Damping is poorly identifiable** across all baselines, exhibiting substantially higher relative error than conservative parameters. **(4) Latent-to-SI calibration** is phenomenon-specific and encoder-dependent, introducing systematic bias in cross-phenomenon comparisons. **Limitations.** IRIS is of moderate scale (240 videos) with a fixed phenomenon set; while this exceeds all prior real-world benchmarks (Table 3), scaling to more phenomena and recording conditions is future work. Damping ground truth is derived from trajectory fits rather than independent measurement (Sec. 3.4). Multi-body contact is modeled via a continuously active coupling coefficient rather than rigorous impact mechanics. **Future work.** Promising directions include latent ODE stabilization techniques [38, 39] for multi-step training, richer contact models [34, 35], and evaluation of Hamiltonian/Lagrangian [16, 17], SINDy [1], and graph neural approaches [18]. Planned dataset extensions include additional phenomena (rolling, bouncing, fluid-like motion), multi-view recordings, and cross-dataset evaluation with physics-plausibility benchmarks [23–25].

8 Conclusion

This work introduces IRIS, a high-fidelity real-world benchmark for physics identification from monocular video comprising 240 videos at 4K resolution spanning eight dynamical systems, including three novel multi-body scenarios, with ground-truth parameters and uncertainty estimates. A standardized five-axis evaluation protocol enables fair comparison. Four equation-identification strategies are benchmarked, revealing that VLM strategy rankings reverse across benchmarks while CNN classifiers achieve near-perfect in-distribution accuracy. A multi-step loss variant improves identifiability on selected single-body dynamics but exposes catastrophic instabilities on multi-body scenarios. These findings, together with the diagnosis of a gradient-flow bug in the state-of-the-art pipeline, demonstrate the diagnostic value of rigorous benchmarking and define concrete open problems.

Acknowledgements

We gratefully acknowledge the STEAM Lab at HBKU for supporting the dataset collection by providing most of the objects used in the experiments. This work was supported by the Student Research Experience Grant (SREG-ECEN-2026) from the Office of Research and Graduate Studies at TAMU-Q.

References

[1] Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. "Discovering governing equations from data by sparse identification of nonlinear dynamical systems". In: *Proceedings of the National Academy of Sciences* 113.15 (2016), pp. 3932–3937.

[2] Florian Hofherr, Lukas Koestler, Florian Bernard, and Daniel Cremers. "Neural implicit representations for physical parameter inference from a single video". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2023, pp. 2093–2103.

[3] Miguel Jaques, Michael Burke, and Timothy Hospedales. "Physics-as-inverse-graphics: Unsupervised physical parameter estimation from video". In: *arXiv preprint arXiv:1905.11169* (2019).

[4] Michael Schmidt and Hod Lipson. "Distilling free-form natural laws from experimental data". In: *Science* 324.5923 (2009), pp. 81–85.

[5] Filipe de Avila Belbute-Peres, Kevin Smith, Kelsey Allen, Josh Tenenbaum, and J. Zico Kolter. "End-to-end differentiable physics for learning and control". In: *Advances in Neural Information Processing Systems (NeurIPS)* 31 (2018).

[6] Petar Veličković, Matko Bošnjak, Thomas Kipf, et al. "Reasoning-modulated representations". In: *Learning on Graphs Conference*. PMLR. 2022, pp. 50–1.

[7] 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: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2025, pp. 27924–27933.

[8] Martin Asenov, Michael Burke, Daniel Angelov, Todor Davchev, Kartic Subr, and Subramanian Ramamoorthy. "Vid2Param: Modeling of dynamics parameters from video". In: *IEEE Robotics and Automation Letters* 5.2 (2019), pp. 414–421.

[9] Nicholas Watters, Daniel Zoran, Theophane Weber, Peter Battaglia, Razvan Pascanu, and Andrea Tacchetti. "Visual interaction networks: Learning a physics simulator from video". In: *Advances in Neural Information Processing Systems (NeurIPS)* 30 (2017).

[10] 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". In: *Advances in Neural Information Processing Systems (NeurIPS)* 28 (2015).

[11] Jiajun Wu, Erika Lu, Pushmeet Kohli, Bill Freeman, and Josh Tenenbaum. "Learning to see physics via visual de-animation". In: *Advances in Neural Information Processing Systems (NeurIPS)* 30 (2017).

[12] Tsung-Yen Yang, Justinian Rosca, Karthik Narasimhan, and Peter J. Ramadge. "Learning physics constrained dynamics using autoencoders". In: *Advances in Neural Information Processing Systems (NeurIPS)* 35 (2022), pp. 17157–17172.

[13] David Zheng, Vinson Luo, Jiajun Wu, and Joshua B. Tenenbaum. "Unsupervised learning of latent physical properties using perception-prediction networks". In: *arXiv preprint arXiv:1807.09244* (2018).

[14] Peter Toth, Danilo Jimenez Rezende, Andrew Jaegle, Sébastien Racanière, Aleksandar Botev, and Irina Higgins. "Hamiltonian generative networks". In: *arXiv preprint arXiv:1909.13789* (2019).

[15] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. "Neural ordinary differential equations". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 31. 2018, pp. 6571–6583.

[16] Samuel Greydanus, Misko Dzamba, and Jason Yosinski. "Hamiltonian neural networks". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 32. 2019, pp. 15353–15363.

[17] Miles Cranmer, Sam Greydanus, Stephan Hoyer, Peter Battaglia, David Spergel, and Shirley Ho. "Lagrangian neural networks". In: *ICLR Workshop on Integration of Deep Neural Models and Differential Equations*. 2020.

[18] Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter W. Battaglia. "Learning to simulate complex physics with graph networks". In: *Proceedings of the International Conference on Machine Learning (ICML)*. Vol. 119. PMLR. 2020, pp. 8459–8468.

[19] Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Rezende, and Koray Kavukcuoglu. "Interaction networks for learning about objects, relations and physics". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 29. 2016, pp. 4502–4510.

[20] Jiajun Wu, Joseph J. Lim, Hongyi Zhang, Joshua B. Tenenbaum, and William T. Freeman. "Physics 101: Learning physical object properties from unlabeled videos". In: *British Machine Vision Conference (BMVC)*. 2016.

[21] Hritik Bansal, Zongyu Lin, Tianyi Xie, et al. "VideoPhy: Evaluating physical commonsense for video generation". In: *arXiv preprint arXiv:2406.03520* (2024).

[22] Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. "Do generative video models understand physical principles?" In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2026, pp. 948–958.

[23] Chenyu Zhang, Daniil Cherniavskii, Antonios Tragoudaras, et al. *Morpheus: Benchmarking Physical Reasoning of Video Generative Models with Real Physical Experiments*. 2025. arXiv: `2504.02918 [cs.CV]`. URL: `https://arxiv.org/abs/2504.02918` (visited on 06/25/2026).

[24] Meng-Hao Guo, Jiajun Xu, Yi Zhang, et al. *R-Bench: Graduate-level Multi-disciplinary Benchmarks for LLM & MLLM Complex Reasoning Evaluation*. 2025. arXiv: `2505.02018 [cs.CV]`. URL: `https://arxiv.org/abs/2505.02018` (visited on 06/30/2026).

[25] Ao Liang, Lingdong Kong, Tianyi Yan, et al. *WorldLens: Full-Spectrum Evaluations of Driving World Models in Real World*. 2025. arXiv: `2512.10958 [cs.CV]`. URL: `https://arxiv.org/abs/2512.10958` (visited on 06/30/2026).

[26] Konstantinos I. Roumeliotis and Nikolaos D. Tselikas. "ChatGPT and open-AI models: A preliminary review". In: *Future Internet* 15.6 (2023), p. 192.

[27] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. "Visual instruction tuning". In: *Advances in Neural Information Processing Systems (NeurIPS)* 36 (2023), pp. 34892–34916.

[28] Yuanhan Zhang, Jinming Wu, Wei Li, et al. "LLaVA-Video: Video instruction tuning with synthetic data". In: *arXiv preprint arXiv:2410.02713* (2024).

[29] Vahid Balazadeh, Mohammadmehdi Ataei, Hyunmin Cheong, Amir Hosein Khasahmadi, and Rahul G. Krishnan. "Physics context builders: A modular framework for physical reasoning in vision-language models". In:

Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). 2025, pp. 7318–7328.

[30] Jiaqi Liu, Songning Lai, Pengze Li, et al. "Mimicking the Physicist's Eye: A VLM-centric Approach for Physics Formula Discovery". In: *arXiv preprint arXiv:2508.17380* (2025).

[31] Kathleen Champion, Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. "Data-driven discovery of coordinates and governing equations". In: *Proceedings of the National Academy of Sciences* 116.45 (2019), pp. 22445–22451.

[32] Michael D. Schmidt and Hod Lipson. "Distilling Free-Form Natural Laws from Experimental Data". In: *Science* 324 (2009), pp. 81–85.

[33] Ruikun Li, Yan Lu, Shixiang Tang, Biqing Qi, and Wanli Ouyang. *MLLM-based Discovery of Intrinsic Coordinates and Governing Equations from High-Dimensional Data*. 2025. arXiv: 2505.11940 [cs.CE].

[34] Meizhong Wang, Wanxin Jin, Kun Cao, Lihua Xie, and Yiguang Hong. *ContactGaussian-WM: Learning Physics-Grounded World Model from Videos*. 2026. arXiv: 2602.11021 [cs.RO]. URL: https://arxiv.org/abs/2602.11021 (visited on 06/30/2026).

[35] Yiren Song, Xiaokang Liu, and Mike Zheng Shou. *DiffSim: Taming Diffusion Models for Evaluating Visual Similarity*. 2024. arXiv: 2412.14580 [cs.CV]. URL: https://arxiv.org/abs/2412.14580 (visited on 06/30/2026).

[36] Jonas Degrave, Michiel Hermans, Joni Dambre, and Francis wyffels. *A Differentiable Physics Engine for Deep Learning in Robotics*. 2018. arXiv: 1611.01652 [cs.NE]. URL: https://arxiv.org/abs/1611.01652 (visited on 06/30/2026).

[37] Keenon Werling, Dalton Omens, Jeongseok Lee, Ioannis Exarchos, and C. Karen Liu. *Fast and Feature-Complete Differentiable Physics for Articulated Rigid Bodies with Contact*. 2021. arXiv: 2103.16021 [cs.RO]. URL: https://arxiv.org/abs/2103.16021 (visited on 06/30/2026).

[38] Chris Finlay, Jörn-Henrik Jacobsen, Levon Nurbekyan, and Adam M Oberman. *How to train your neural ODE: the world of Jacobian and kinetic regularization*. 2020. arXiv: 2002.02798 [stat.ML]. URL: https://arxiv.org/abs/2002.02798 (visited on 06/30/2026).

[39] Sifan Wang, Hanwen Wang, and Paris Perdikaris. *Learning the solution operator of parametric partial differential equations with physics-informed DeepOnets*. 2021. arXiv: 2103.10974 [cs.LG]. URL: https://arxiv.org/abs/2103.10974 (visited on 06/30/2026).

[40] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, et al. *SAM 2: Segment Anything in Images and Videos*. 2024. arXiv: 2408.00714 [cs.CV].

Supplementary Material

IRIS: A Real-World Benchmark for Inverse Recovery and Identification of Physical Dynamic Systems from Monocular Video

A Dataset Statistics and Release Details

Table 9 provides a complete per-phenomenon, per-setting breakdown of the IRIS dataset. All ground-truth physical parameters are obtained through direct manual measurement using calibrated instruments (see Appendix C), with the exception of damping and friction coefficients, which are derived from trajectory fits as described in Appendix C. Each parameter entry records mean, standard deviation, minimum, and maximum values; where the standard deviation is zero, the measurement reflects a single precisely controlled experimental condition.

B Physical Model Derivations and ODE Bank

This appendix makes explicit which physical ODE governs each phenomenon, how the latent-space parameterisation (α, β) relates to physical constants, and where the pipeline's ODE form differs from the exact physical equation. The purpose is to prevent misreading of the latent parameters as direct physical measurements without the conversions listed below.

B.1 Phenomenon–ODE Correspondence Table

Table 10 summarises the mapping from IRIS/Delfys75 phenomena to physical ODEs and to the latent ODE form used in the pipeline. Where a mismatch exists, i.e. where the physical ODE is not a linear second-order oscillator but the pipeline treats it as one for the sake of a unified ODE bank, this is noted explicitly.

B.2 Design Choice: Unified Linear ODE Bank

The pipeline uses a unified second-order linear ODE $\ddot{z} + \beta\dot{z} + \alpha z = 0$ as the latent representation for all oscillatory and gravity-driven phenomena. This is a deliberate design choice for comparability across phenomena: the encoder learns to embed the observed motion into a one-dimensional latent z such that the linear ODE describes the latent trajectory, regardless of whether the underlying physical state (height, angle, radius) follows the same form exactly.

Dropping ball. The true physical ODE is $\ddot{h} = -g$ (constant acceleration with h in metres, t in seconds). The pipeline approximates this by $\ddot{z} + \alpha z = 0$ with $\alpha \approx g$; this approximation holds when the encoder maps the pixel position of the ball linearly to z and the observation window is short enough that the

linear approximation of the trajectory remains valid. When $\beta \neq 0$, the term $\beta\dot{z}$ models aerodynamic drag. In practice, for the drop heights in IRIS (0.5–1.5 m), the drag term is small and $\alpha \approx g$ gives a physically meaningful estimate.

Falling ball (apparent radius). The ball falls away from the camera; the observable is its apparent radius $r(t) = r_0 f/(h_0 + \frac{1}{2}gt^2)$, where r_0 is the physical ball radius, f the focal length, and h_0 the initial camera-to-ball distance. The latent z encodes this apparent size rather than height; the same second-order form is used, and the fitted α is compared to g after the same linear conversion. This is an approximation and is listed as such in Table 10.

Pendulum. For small oscillation angles $\theta_0 \leq 45°$, $\sin\theta \approx \theta$ and the linearised equation $\ddot{\theta} + \zeta\dot{\theta} + (g/L)\theta = 0$ is an exact match to the unified ODE form, with $\alpha = g/L$ and $\beta = \zeta$. For large angles ($\theta_0 = 90°$), the linearisation error is substantial ($\sin 90°/90° \approx 0.64$ in radians), and the fitted α will deviate from g/L accordingly. This is a **known modelling limitation**: on the pend._90 setting, higher parameter error is expected and should be interpreted as an identifiability challenge rather than a pipeline failure.

B.3 Dropping Ball vs. Falling Ball: Symbol Definitions

To avoid ambiguity between the two gravity-related phenomena:

- **Dropping ball**: camera is placed *to the side* of the drop. The observable is the *vertical pixel position* of the ball, which decreases linearly with $h(t) = h_0 - \frac{1}{2}gt^2$ (taking downward as positive). Ground truth: $h_0 \in \{0.50, 1.00, 1.50\}$ m, $d_{\text{cam}} = 1.94$ m. Estimated parameter: g.
- **Falling ball**: camera is placed *directly above*, looking down. The ball falls *away* from the camera; the observable is the *apparent radius* $r(t) = r_0 f/(h_0 + \frac{1}{2}gt^2)$ (ball grows smaller). Ground truth: ball radii $r_0 \in \{0.04, 0.07, 0.11\}$ m; $g = 9.81\,\text{m/s}^2$. Estimated parameter: g.

Symbol table: h_0 = initial drop height (m); d_{cam} = lateral camera distance (m); r_0 = physical ball radius (m); f = effective focal length (pixels or m); g = gravitational acceleration (m/s^2); β = drag/damping coefficient; α = ODE stiffness coefficient.

C Ground-Truth Measurement Protocol

C.1 Directly Measured Parameters

Parameters labelled **D** in Table 9 are obtained through direct measurement before or after recording, independent of the video:

- **Heights and lengths** (h_0, L, L_1, L_2, hypotenuse length): measured with a rigid tape measure to the nearest millimetre. Each measurement is repeated three times; we report the mean. Uncertainty is ±2 mm for lengths up to 1 m and ±5 mm for longer spans.

- **Angles** (θ_0, α): set with an adjustable protractor or inclinometer calibrated to $\pm 0.5°$. Cone incline angles are verified against the nominal setting after positioning.
- **Distances** (d_{cam}, $d_{\text{ball-cones}}$): measured with a laser distance meter (accuracy $\pm 2\,\text{mm}$).
- **Ball radius** (r_0): measured with digital calipers (resolution 0.01 mm); three measurements taken at different orientations, mean reported.
- **Gravitational acceleration** (g): taken as $9.81\,\text{m/s}^2$ (standard local value for the recording location; no significant variation expected across sessions).

C.2 Fitted Parameters: Damping and Friction

Parameters labelled **F** in Table 9 are *not* directly measurable in the same way as lengths or angles; they are instead estimated from recorded video trajectories:

Damping coefficients (ζ for pendulum, β for rotating cone). A reference video is selected per setting (the first clip of the session). The object's 2-D centroid or angular position is tracked frame by frame using a background-subtraction and blob-detection procedure. The resulting time series $\theta(t)$ is fitted to the analytical solution of the relevant damped oscillator ODE using nonlinear least squares (Levenberg–Marquardt). The fitted ζ (or β) is reported as ground truth for that setting. We use the same setting across all 10 trial videos. **Uncertainty:** we report the standard deviation of ζ across 10 independent fits; typical values are $\sigma_\zeta < 0.01\,\text{s}^{-1}$ for the pendulum and $\sigma_\beta < 0.005\,\text{s}^{-1}$ for the rotating cone.

Friction coefficient (μ for sliding cone). The video is segmented to track the cone's position along the slope. Acceleration a is estimated from a second-order polynomial fit to the position time series. The friction coefficient is then computed analytically as $\mu = \tan\alpha - a/(g\cos\alpha)$, where α is the measured incline angle. We average μ across 10 trial clips; standard deviation is reported. Typical $\sigma_\mu < 0.02$.

Limitations. Fitted parameters carry the error of the tracker and the fit model. For large oscillation angles ($\theta_0 = 90°$), the small-angle approximation used in the linear-ODE fit is less accurate, and the reported ζ should be interpreted as the *effective* damping under the linearised model. We do not propagate tracking noise uncertainty into the final GT value; this is noted as a limitation and flagged in any per-setting identifiability discussion.

D Latent-to-Physical Parameter Calibration

The encoder maps raw pixel intensities to a latent z_t that is dimensionless. The physics block then fits α, β in this latent space. To obtain physical parameters (m, m/s^2, s^{-1}, etc.), the following per-phenomenon conversions are applied in the evaluation script (`compare_baseline_unified.py` and the IRIS counterpart):

Temporal calibration. The pipeline receives pre-extracted video tensors with a fixed frame stride. The time step Δt passed to the integrator must match

the physical frame interval; we set Δt to the inverse of the capture frame rate (60 fps $\rightarrow$ $\Delta t = 1/60$ s for IRIS; Delfys75 uses $\Delta t = 0.05$ s as in the original codebase). Mismatches between Δt and the true frame interval directly scale α and β by Δt^{-2} and Δt^{-1} respectively, which would bias all physical-parameter estimates. We verify that Δt is set consistently for each run.

Cross-phenomenon bias. The latent normalisation is phenomenon-specific: the encoder is trained from scratch on each clip, so the scale of z differs across phenomena. The calibration formulas above absorb this by expressing physical parameters directly in terms of α, β and known constants. However, no explicit pixel-to-metre calibration is performed; for phenomena where the latent encodes a pixel-level observable (e.g., apparent radius in the falling-ball case), small residual scale errors may remain if the encoder's internal normalisation drifts across clips. This is an acknowledged limitation; we recommend reporting relative rather than absolute parameter error for cross-phenomenon comparison.

E VLM Prompting Strategies and Ablations

The main paper reports three equation-identification strategies: VLM temporal reasoning (73.33% on Delfys75, 65.00% on IRIS), VLM describe-then-classify (61.11% on Delfys75, 72.92% on IRIS), and a CNN video classifier (98.40% on Delfys75, 99.30% on IRIS). This appendix documents the exact prompt design and discusses additional strategies evaluated during development.

E.1 Frame Sampling

For all VLM calls, $m = 5$ representative frames are sampled from the clip at positions $\{0\%, 25\%, 50\%, 75\%, 100\%\}$ of the total duration. Earlier experiments used 3 frames (start, middle, end) and achieved $\sim$67% accuracy; adding the quartile frames increased accuracy to 73%. We use the same sampling for all strategies below.

E.2 Temporal Reasoning (Primary Variant)

The VLM (accessed via OpenRouter with a GPT-4V-class model) receives the 5 sampled frames as a sequence of images and the following system-level prompt:

> *"You are a physics expert. I will show you five frames from a video in temporal order. Your task is to identify the type of motion shown. Compare the first frame to the last frame to determine whether the motion is oscillatory, accelerating, decaying, rotating, or involves a collision. Then choose exactly one label from: {pendulum, free_fall, dropped_ball, sliding_block, led, torricelli}. Definitions: [one sentence per class]. Output only the label, nothing else."*

The key design choice is instructing the model to *compare early and late frames* to resolve temporal ambiguity (e.g., oscillatory pendulum vs. one-directional free fall, which can look identical in a single frame). This temporal-reasoning instruction is the main factor distinguishing this variant from the

baseline VLM (which receives the same frames but without the comparison instruction), and accounts for most of the accuracy gain from ∼61% to 73% on Delfys75.

E.3 Describe-Then-Classify (Two-Call Variant)

This variant uses two sequential API calls. First, a description call asks the VLM to describe the motion in free text without any class labels provided. Second, a classification call sends that text description (not the images) to the model and asks it to assign one of the class labels. The intent is to use the VLM's language-reasoning strengths for the final label assignment, rather than relying on direct visual classification.

On Delfys75, this approach achieved 61.11% accuracy, lower than temporal reasoning (73.33%). Per-class breakdown: dropped_ball 100%, free_fall 0%, sliding_block 80%, torricelli 73%, led 60%, pendulum 53%. The collapse on free_fall (0%) is the dominant failure: the description step produces text that is ambiguous between free_fall and pendulum (e.g., "a ball moving downward"), and the classification step systematically maps this to pendulum.

Reversal on IRIS. Notably, the ranking of these two VLM strategies reverses on IRIS: describe-then-classify achieves 72.92% while temporal reasoning drops to 65.00%. This reversal is attributable to the greater visual diversity of IRIS's eight phenomena, including three multi-body scenarios. The two-stage pipeline allows the VLM to first articulate complex dynamics (collisions, coupled oscillations, multiple moving objects) in free-form natural language, producing richer intermediate representations that facilitate disambiguation between single-body and multi-body phenomena. By contrast, the single-call temporal reasoning prompt, which was designed around the six Delfys75 classes, struggles to capture the novel motion patterns in IRIS within a single classification step. This finding suggests that no single VLM prompting strategy generalizes reliably across benchmarks, and motivates research on adaptive or ensemble-based equation routing.

E.4 Additional Strategies Evaluated

The baseline pipeline from Garcia et al. relies exclusively on path-based equation selection, where the correct ODE family is provided via folder structure at inference time. This is an oracle condition that cannot generalise to new recordings without manual labelling. A core motivation for introducing IRIS is to benchmark automatic equation-routing strategies that remove this requirement. VLM-based routing is a natural candidate because it operates zero-shot without retraining, can handle novel phenomena, and leverages broad visual and physical knowledge acquired during pretraining. The strategies below were explored during development to understand the trade-offs across different automation approaches.

Chain-of-thought (CoT) prompting. We evaluated a variant that explicitly asks the model to reason step by step before outputting the label ("First

describe what you observe, then reason about the physics, then state the label"). Accuracy was marginally higher than the baseline VLM (∼65%) but lower than temporal reasoning, with significantly higher token cost per call.

Few-shot prompting. Providing one example image per class in the prompt context improved robustness for visually similar classes (dropped_ball vs. free_fall) but was impractical at scale due to context length limits and per-call cost.

Self-consistency. Sampling the model $k = 5$ times per clip and taking the majority vote improved stability on ambiguous clips but increased cost proportionally, without a meaningful accuracy gain over temporal reasoning.

VLM fine-tuning. We attempted LoRA fine-tuning and a frozen-VLM plus classifier-head approach, both of which collapsed to predicting a single class for every video (∼16.7% accuracy, equal to random for 6 classes). This is likely due to the small dataset size (90 training videos) relative to the VLM's parameter count. Fine-tuning is therefore not recommended and is not used in any reported result.

Video classifier (ResNet-18 + temporal pooling). A lightweight ResNet-18 backbone with mean temporal pooling is trained separately on each benchmark: 6-class on Delfys75 and 8-class on IRIS, using their respective training splits. On Delfys75, the classifier achieves 98.40% accuracy; on IRIS, it achieves 99.30% accuracy. These results confirm that supervised classifiers attain near-perfect in-distribution performance on both benchmarks, but this comes at the cost of requiring retraining whenever the phenomenon set changes. Unlike the zero-shot VLM strategies, the CNN cannot handle novel phenomena without collecting labelled data for the new classes. Disentangling the effect of distribution shift from scenario complexity would require evaluating the Delfys75-trained CNN on IRIS's overlapping phenomena (dropped ball, pendulum, sliding cone); this cross-distribution evaluation is left for future work.

E.5 VLM Confusion Matrix

Table 12 reproduces the confusion matrix for the temporal-reasoning VLM on Delfys75, included here for convenience when reading the prompting discussion above. The primary failure mode is free_fall being predicted as pendulum (10 out of 15 videos): a ball shrinking in apparent size as it falls away from the camera is visually indistinguishable from an object receding in oscillatory motion when only 5 frames are shown without metric scale context. dropped_ball is confused with free_fall (5 of 15) for a symmetric reason: both show a ball at increasing pixel displacement, and the direction (lateral vs. depth-wise) is only apparent from late frames. LED, sliding block, and Torricelli classify without error due to unambiguous visual signatures (brightness change, translational inclined motion, water level decrease).

F Implementation Details

F.1 Encoder Architecture

Following the original pipeline [7], the encoder ϕ is a lightweight MLP operating on per-frame pixel intensities. Input frames are resized to 56×100 (grayscale) and flattened to a 5600-dimensional vector. The MLP has two hidden layers of 256 units each with ReLU activations, and an output of dimension $d = 2$ (mean μ_z and log-variance $\log \sigma_z^2$ for the VAE reparameterisation). The latent state $\hat{z}_t$ is sampled as $\hat{z}_t = \mu_z + \sigma_z \cdot \epsilon$, $\epsilon \sim \mathcal{N}(0, 1)$.

For multi-object IRIS phenomena (hitting cones, two pendulums), the encoder outputs $2N$ values, reshaped to $N \times 2$ per frame. All N objects share encoder weights.

F.2 Physics Block

Each ODE family has a dedicated physics block with learnable scalars $\boldsymbol{\gamma} = (\gamma_0, \gamma_1)$ initialised to $(0.5, 0.05)$. For the second-order ODE $\ddot{z} + \gamma_1 \dot{z} + \gamma_0 z = 0$, the discrete Euler-Cromer step is:

$$\dot{z}_{t+1} = \dot{z}_t + \Delta t \cdot \ddot{z}_t, \tag{4}$$

$$z_{t+1} = z_t + \Delta t \cdot \dot{z}_t + \Delta t^2 \cdot \ddot{z}_t, \tag{5}$$

where $\ddot{z}_t = -\gamma_1 \dot{z}_t - \gamma_0 z_t$. The critical fix relative to the original codebase is the inclusion of the $\Delta t^2 \cdot \ddot{z}_t$ term in the position update: without it, $\partial z_{t+1}/\partial \gamma = 0$ and physics parameters receive no gradient signal during training.

F.3 Training

Each clip is trained independently with Adam; no parameters are shared across clips. Hyperparameters: learning rate 10^{-3} for the encoder, 10^{-2} for $\boldsymbol{\gamma}$; batch size = all temporal windows of the clip; 500 epochs. The one-step loss is:

$$\mathcal{L}_{\text{1-step}} = \frac{1}{M} \sum_{i=1}^{M} \|\hat{\mathbf{z}}_i - \tilde{\mathbf{z}}_i\|^2 + \lambda_{\text{KL}} \, \mathcal{L}_{\text{KL}},$$

and the multi-step loss replaces the first term with $\sum_{k=1}^{K} w_k \|\hat{\mathbf{z}}_{t+k} - \tilde{\mathbf{z}}_{t+k}\|^2$ (main paper, Eq. 2), with $K = 5$ and weights $w = [1.0, 1.0, 0.5, 0.5, 0.25]$. In both cases $\lambda_{\text{KL}} = 0.01$.

All runs use a fixed random seed (seed = 42) for reproducibility. Run outputs are written to CSV with columns `run, alpha, beta, max_z, min_z, z0, z1` (one row per clip) and are processed by `compare_baseline_unified.py` for parameter extraction and GT comparison.

G Full IRIS Parameter Estimation Results

Table 13 reports parameter MAE on IRIS comparing two training objectives applied to the **+Corrected** configuration (gradient-corrected Euler, path-based equation selection): (a) one-step loss (*1-step*) and (b) multi-step rollout loss with K=5 (*Multi*). This corresponds to ablation configurations 2 and 3 in the main paper (Sec. 6.1); it isolates the effect of the loss function independently of VLM routing. $\Delta = \text{MAE}_{\text{multi}} - \text{MAE}_{\text{1-step}}$ (negative = multi-step wins).

Table 13: **Parameter MAE on IRIS: 1-step vs. multi-step loss** (both with corrected Euler, path-based selection; ablation configs 2 vs. 3 of the main paper). Δ=$\text{MAE}_{\text{multi}}$−$\text{MAE}_{\text{1-step}}$; negative = multi-step wins. Green = lower MAE; red = catastrophic ($|\Delta|$>10).

Dynamics	Setting	Param	GT	1-step	Multi.	Δ
Single-body						
Dropping ball	drop_50	g [m/s^2]	9.81	**1.04**	6.88	+5.84
	drop_100	g [m/s^2]	9.81	**2.26**	8.28	+6.02
	drop_150	g [m/s^2]	9.81	**3.83**	8.86	+5.04
Falling ball	big	g [m/s^2]	9.81	**4.26**	6.40	+2.14
	mid	g [m/s^2]	9.81	**6.63**	8.63	+2.01
	small	g [m/s^2]	9.81	**4.13**	6.88	+2.74
Pendulum	pend._20	L [m]	0.50	0.56	**0.40**	−0.16
	pend._45	L [m]	0.50	**0.34**	1.16	+0.82
	pend._90	L [m]	0.50	**0.60**	21.95	+21.34
Rotation	fast	α	0.10	1.54	**2.46**	−0.64
		β	0.08	**0.88**	1.28	+0.39
	mid	α	0.10	3.03	**3.98**	−1.21
		β	0.05	**0.39**	1.38	+0.99
	slow	α	0.10	1.59	**4.63**	−2.11
		β	0.03	0.45	**0.06**	−0.39
Sliding cone	cone_45	α [deg]	45.0	**0.00**	**0.00**	0.00
	cone_60	α [deg]	60.0	**0.00**	**0.00**	0.00
	cone_80	α [deg]	80.0	**0.00**	**0.00**	0.00
Multi-body						
Two mov. pendulums	pend._20	L_1, L_2 [m]	0.50	**1.14**	741.1	+740.0
	pend._45	L_1, L_2 [m]	0.50	**1.63**	164.1	+162.5
	pend._90	L_1, L_2 [m]	0.50	**3.01**	149.9	+146.9
One stat. pendulum	pend._20	L_1, L_2 [m]	0.50	**1.38**	65.4	+64.0
	pend._45	L_1, L_2 [m]	0.50	**0.77**	96.8	+96.0
	pend._90	L_1, L_2 [m]	0.50	**1.06**	164.8	+163.8

Key observations.
(1) *Sliding cone angle:* both variants recover the correct angle exactly because it is taken directly from the setting metadata, not from training.
(2) *Gravity (dropping/falling ball):* the 1-step variant outperforms multi-step consistently here, mirroring the Delfys75 finding (main paper, Table 5). The multi-step loss accumulates integration error over K=5 horizons, inflating the α estimate when the encoder is not yet close to the physical trajectory.
(3) *Pendulum pend._90:* the catastrophic multi-step failure ($\Delta = +21.34$) reflects large-angle nonlinearity. The linearised ODE $\ddot{z} + \beta\dot{z} + \alpha z = 0$ is a poor model for $\theta_0 = 90°$ (Appendix B), and multi-step rollout amplifies this approximation error over K steps.
(4) *Multi-body pendulums:* both variants yield large absolute errors. This is expected given the simplified contact model (Appendix H) and the brevity of the contact window relative to the total clip. The multi-step loss further destabilises training for these under-constrained scenarios. These rows serve as a baseline target for future multi-body methods.
(5) *Hitting cones (omitted):* the hitting-cones scenario is excluded from this table because the effective coupling coefficient κ has no independently measured ground-truth value; it is an artefact of the spring-like simplification (Appendix H) rather than a physical observable. Qualitatively, both configurations produce near-zero κ estimates on this scenario, consistent with the finding that the contact window is too brief relative to the clip duration for the coupling term to receive a meaningful gradient signal.

H Multi-Body Contact Modeling: Details and Limitations

The graph-structured physics block models inter-object forces via a learnable spring-like coupling term $\kappa_{ij}(z_i - z_j)$ (main paper, Eq. 1). This is a deliberate simplification motivated by two goals: (a) the architecture reduces to the single-body ODE exactly when no edges are present ($N = 1$), enabling a unified codebase; (b) the coupling term is differentiable and allows gradient-based identification of interaction strength.

Known limitations of this model. Real physical impacts involve:

- *Restitution*: velocity reversal at contact, characterised by the coefficient of restitution $e \in [0, 1]$, which is not representable by a static coupling coefficient.
- *Finite contact duration*: impacts in rigid-body mechanics are instantaneous (or very brief), while the spring model spreads the force over a continuous time window proportional to $1/\kappa_{ij}$.
- *Friction at contact*: the tangential impulse during collision is not modelled.

As a consequence, the fitted κ_{ij} for hitting-cone and two-pendulum scenarios is an *effective* coupling coefficient that conflates all of the above effects into a single linear term. Parameter estimates for multi-body IRIS scenarios should therefore be interpreted as "effective coupling strength" rather than as physically

interpretable constants, and are primarily useful as a benchmark target for future methods.

Coupling captures a real signal. Although κ has no externally measured reference, we verify it is not a noise-fitting artefact by training the hitting-cones model with κ=0 (no coupling) versus κ=learned and comparing latent-space reconstruction MSE (Table 14). The learned coupling reduces reconstruction error by 6.0–19.0% across all settings, and the estimated κ is stable across the 10 trials per setting (coefficient of variation ≈40–47%), confirming a reproducible, physically meaningful effective parameter.

Identifiability note. The coupling parameter κ_{ij} is only observable during the brief contact interval. For the two-pendulum and one-static-pendulum settings, this interval represents a small fraction of the total clip (≤ 1 s out of 6–20 s); the latent loss over the full clip is dominated by the non-contact phase, where κ_{ij} is invisible. This explains the large MAE values in Table 13 for multi-body settings. Future work should consider contact-segmented training windows or event-conditioned physics blocks.

I Multi-Clip vs. Per-Clip Training

The main paper (Sec. 6.4) reports a multi-clip experiment that complements the per-clip identifiability protocol. For each phenomenon, clips are split 80/20 by sorted index; one shared encoder–physics model is trained on the 80% train clips and evaluated on the held-out 20% test clips, against per-clip fitting on the same test clips. Table 15 reports the full results. Multi-clip outperforms per-clip for gravity (both ball phenomena) and angular stiffness, demonstrating a transferable physical inductive bias across clips; the nonlinear pendulum is the open failure mode, where a shared latent space is underdetermined under coupled dynamics.

J Identifiability Analysis

Motivation. Parameter identifiability, whether a unique parameter vector $\boldsymbol{\gamma}^*$ can be recovered from the observed data, is a fundamental concern for any parameter estimation method. We provide a preliminary analysis using two proxies: (1) gradient norms during training (do physics parameters receive gradient signal?), and (2) estimate variance across repeated runs.

J.1 Gradient Norms

The critical Euler fix (Appendix F.2) ensures that $\partial z_{t+1}/\partial \boldsymbol{\gamma} \neq 0$. We confirmed this empirically: without the fix, γ_0 and γ_1 remained at their initial values $(0.5, 0.05)$ for all 500 epochs; gradient norms were $< 10^{-8}$ throughout. After the fix, on a representative dropped_ball clip, gradient norms are $\|\nabla_\gamma \mathcal{L}\|_2 \approx 0.3$–$1.5$ at epoch 1, decaying to ≈ 0.05–0.2 at convergence, confirming that training is meaningful for the physics block.

Parameter trajectories. On a dropped_ball clip, γ_0 evolves from $0.5 \rightarrow 4.65$ over 500 epochs (target: $g \approx 9.81$, relative error $\sim$50%); γ_1 (drag) evolves from $0.05 \rightarrow 1.11$. For pendulum clips, γ_0 converges to values implying $L = g/\gamma_0 \in [0.5, 1.6]$ m, bracketing the true length of 0.5 m for pend._20 and pend._45 but overestimating for pend._90 (consistent with the large-angle linearisation error).

J.2 Estimate Variance Across Trials

Table 16 reports the mean and standard deviation of estimated parameters across 10 repeated trials per setting (IRIS baseline, one-step loss).

Table 16: **Estimate variance across 10 trials per IRIS setting.** $\bar{\gamma}$ = mean estimate; σ = std; GT = ground truth. Low σ indicates stable convergence; high σ indicates identifiability difficulties.

Dynamics	**Setting**	**Param**	**GT**	$\bar{\gamma}$	$\boldsymbol{\sigma}$
Dropping ball	drop_50	g	9.81	8.77	0.42
Dropping ball	drop_100	g	9.81	7.55	1.21
Dropping ball	drop_150	g	9.81	5.98	2.04
Pendulum	pend._20	L [m]	0.50	1.06	0.88
Pendulum	pend._45	L [m]	0.50	0.84	0.41
Pendulum	pend._90	L [m]	0.50	1.10	0.73
Sliding cone	cone_45	α [deg]	45.0	45.0	0.00
Rotation	slow	β	0.03	0.48	0.09

Observations. (1) Sliding cone angle is recovered perfectly across all trials (metadata-based, variance = 0). (2) Gravity estimates are stable for short drops (drop_50: $\sigma = 0.42$) but increasingly variable for longer drops, where the quadratic trajectory requires more clip length to constrain α. (3) Pendulum length shows moderate variance, consistent with the identifiability challenge of disentangling $\gamma_0 = g/L$ from the encoder's internal scale. Full identifiability analysis, including Fisher information matrix computation and Hessian condition numbers for $\boldsymbol{\gamma}$, is left for future work.

Table 9: **Full IRIS dataset statistics.** Each row is one setting; n is the number of videos. d_{cam} denotes camera-to-object distance. Measurement type: **D** = direct, **F** = fitted.

Phenomenon	Setting	n	Dur. (s)	Res.	GT params (measured)	Type
Single-body						
Dropping ball	drop_50	10	5	3840×2160	h_0=0.50 m, d_{cam}=1.94 m	D
	drop_100	10	5	3840×2160	h_0=1.00 m, d_{cam}=1.94 m	D
	drop_150	10	5	3840×2160	h_0=1.50 m, d_{cam}=1.94 m	D
Falling ball	big	10	8	3840×2160	g=9.81 m/s^2, r_0=0.11 m	D
	mid	10	8	3840×2160	g=9.81 m/s^2, r_0=0.07 m	D
	small	10	8	3840×2160	g=9.81 m/s^2, r_0=0.04 m	D
Sliding cone	cone_45	10	5	3840×2160	α=45°, hyp. =0.77 m, μ (fit)	D / μ:F
	cone_60	10	5	3840×2160	α=60°, hyp. =0.84 m, μ (fit)	D / μ:F
	cone_80	10	5	3840×2160	α=80°, hyp. =0.80 m, μ (fit)	D / μ:F
Pendulum	pend._20	10	150	3840×2160	θ_0=20°, L=0.50 m, ζ (fit)	D / ζ:F
	pend._45	10	150	3840×2160	θ_0=45°, L=0.50 m, ζ (fit)	D / ζ:F
	pend._90	10	150	3840×2160	θ_0=90°, L=0.50 m, ζ (fit)	D / ζ:F
Rotating cone	slow	10	8	3840×2160	Half-circle init., β (fit)	D / β:F
	mid	10	8	3840×2160	1-circle init., β (fit)	D / β:F
	fast	10	8	3840×2160	2-circle init., β (fit)	D / β:F
Multi-body						
Hitting cones	slow	10	5	3840×2160	$d_{\text{ball-cones}}$=2.00 m, d_{cam}=2.20 m	D
	mid	10	5	3840×2160	$d_{\text{ball-cones}}$=2.00 m, d_{cam}=2.20 m	D
	fast	10	5	3840×2160	$d_{\text{ball-cones}}$=2.00 m, d_{cam}=2.20 m	D
Two mov. pendulums	pend._20	10	6	3840×2160	θ_0=20°, L_1=L_2=0.50 m	D
	pend._45	10	6	3840×2160	θ_0=45°, L_1=L_2=0.50 m	D
	pend._90	10	6	3840×2160	θ_0=90°, L_1=L_2=0.50 m	D
One stat. pendulum	pend._20	10	20	3840×2160	θ_0=20°, L_1=L_2=0.50 m	D
	pend._45	10	20	3840×2160	θ_0=45°, L_1=L_2=0.50 m	D
	pend._90	10	20	3840×2160	θ_0=90°, L_1=L_2=0.50 m	D
Total		**240**	–	–	–	–

Table 10: **Phenomenon–ODE correspondence.** Column "Physical ODE" is the exact governing equation. Column "Pipeline ODE form" is the form actually used by the encoder–physics block. Column "Latent → SI" lists the conversion from fitted (α, β) to physical parameters. $g = 9.81\,\mathrm{m/s^2}$ is treated as a known constant only for the pendulum length conversion; for dropping/falling ball it is the *estimated* quantity.

Phenomenon	Physical ODE	Pipeline ODE form	Latent → SI	Note
Dropping ball	$\ddot{z} = -g$ (constant accel.)	$\ddot{z} + \beta\dot{z} + \alpha z = 0$	$g = \alpha$	Unified linear form; β absorbs drag
Falling ball	$r(t) = r_0 f/(h_0 + \frac{1}{2}gt^2)$	$\ddot{z} + \beta\dot{z} + \alpha z = 0$	$g = \alpha$	Apparent radius; same latent form
Pendulum (small angle)	$\ddot{\theta} + \zeta\dot{\theta} + (g/L)\theta = 0$	$\ddot{z} + \beta\dot{z} + \alpha z = 0$	$L = g/\alpha,\ \zeta = -\beta$	Exact match for small angles
Rotating cone	$\ddot{\varphi} + \beta\dot{\varphi} + \alpha\varphi = 0$	$\ddot{z} + \beta\dot{z} + \alpha z = 0$	$\alpha_{\mathrm{SI}} = \alpha,\ \beta_{\mathrm{SI}} = \beta$	Exact match
Sliding cone	$\ddot{x} = g(\sin\alpha - \mu\cos\alpha)$	Constant-acceleration block	$\mu = \beta$; angle from metadata	Angle fixed from setting
LED decay	$\dot{z} = -\lambda z$	First-order decay block	$\gamma = \beta$	First-order; α unused
Torricelli drain	$\dot{z} = -k\sqrt{z}$	Torricelli block	$k = \alpha$	Nonlinear; separate block
Hitting cones	Momentum transfer (impact)	Graph ODE, coupling κ_{ij}	κ_{ij} (spring-like)	Simplified; see App. H
Two pendulums	$\ddot{\theta}_i + \zeta_i\dot{\theta}_i + (g/L_i)\theta_i + \kappa_{ij}(\theta_i - \theta_j) = 0$	Graph ODE, same form	$L_i = g/\alpha_i,\ \kappa_{ij}$ (coupling)	Active only during contact

Table 11: **Per-phenomenon calibration and physical parameter extraction.** α_{fit}, β_{fit} are the dimensionless fitted values. The rightmost column notes any normalisation or assumption.

Phenomenon	Extracted param.	Formula	Assumption / note
Dropping ball	$g\,[\mathrm{m/s^2}]$	$g = \alpha_{\mathrm{fit}}$	Latent = pixel height; dt in seconds
Falling ball	$g\,[\mathrm{m/s^2}]$	$g = \alpha_{\mathrm{fit}}$	Latent = apparent radius
Pendulum	$L\,[\mathrm{m}]$	$L = g_{\mathrm{true}}/\alpha_{\mathrm{fit}}$	$g_{\mathrm{true}} = 9.81\,\mathrm{m/s^2}$ fixed
Pendulum	$\zeta\,[\mathrm{s^{-1}}]$	$\zeta = -\beta_{\mathrm{fit}}$	Sign convention
Rotating cone	α, β	direct	Same ODE form
Sliding cone	μ	$\mu = \beta_{\mathrm{fit}}$	Angle from metadata
LED decay	$\gamma\,[\mathrm{s^{-1}}]$	$\gamma = \beta_{\mathrm{fit}}$	First-order; α unused
Torricelli	k	$k = \alpha_{\mathrm{fit}}$	Nonlinear Torricelli block

Table 12: **Confusion matrix, temporal-reasoning VLM on Delfys75.** Rows = ground truth, columns = predicted. Diagonal = correct.

	drop.	**free**	**led**	**pend.**	**slide**	**torr.**	Σ
dropped_ball	**10**	5	0	0	0	0	15
free_fall	0	**0**	0	10	5	0	15
led	0	0	**15**	0	0	0	15
pendulum	2	2	0	**11**	0	0	15
sliding_block	0	0	0	0	**15**	0	15
torricelli	0	0	0	0	0	**15**	15
Acc	66/90 = **73.3%**						

Table 14: **Coupling-coefficient validation (hitting cones).** Latent reconstruction MSE for κ=0 vs. κ=learned; the learned coupling reduces error by 6–19%, confirming it captures real physical interaction rather than noise.

Setting	**κ=0 MSE**	**κ=learned MSE**	**Reduction**
Slow	0.0784	**0.0737**	6.0%
Mid	1.0049	**0.8144**	19.0%
Fast	0.2743	**0.2453**	10.5%

Table 15: **Multi-clip vs. per-clip on IRIS held-out test clips** (MAE; lower better; Δ=multi−per-clip). Multi-clip wins on gravity and angular stiffness; the nonlinear pendulum is the open failure mode.

Phenomenon	**Param**	**GT**	**Per-clip**	**Multi-clip**	**Δ**
Dropping ball	g	9.81	0.207	**0.046**	−0.161
Falling ball	g	9.81	0.246	**0.166**	−0.080
Pendulum	L [m]	0.50	**0.507**	22.06	+21.55
Rotation	β	0.03	**0.657**	0.851	+0.194
Rotation	α	0.10	1.037	**0.577**	−0.459