SWIM2REAL: VLM-Guided System Identification for Sim-to-Real Transfer

Kevin Qiu[1,2], Kyle Walker[3], Mike Y. Michelis[4], Marek Cygan[1,5], Josie Hughes[3]

[1]University of Warsaw [2]IDEAS NCBR [3]EPFL [4]ETH Zurich [5]Nomagic
kevinxqiu@gmail.com

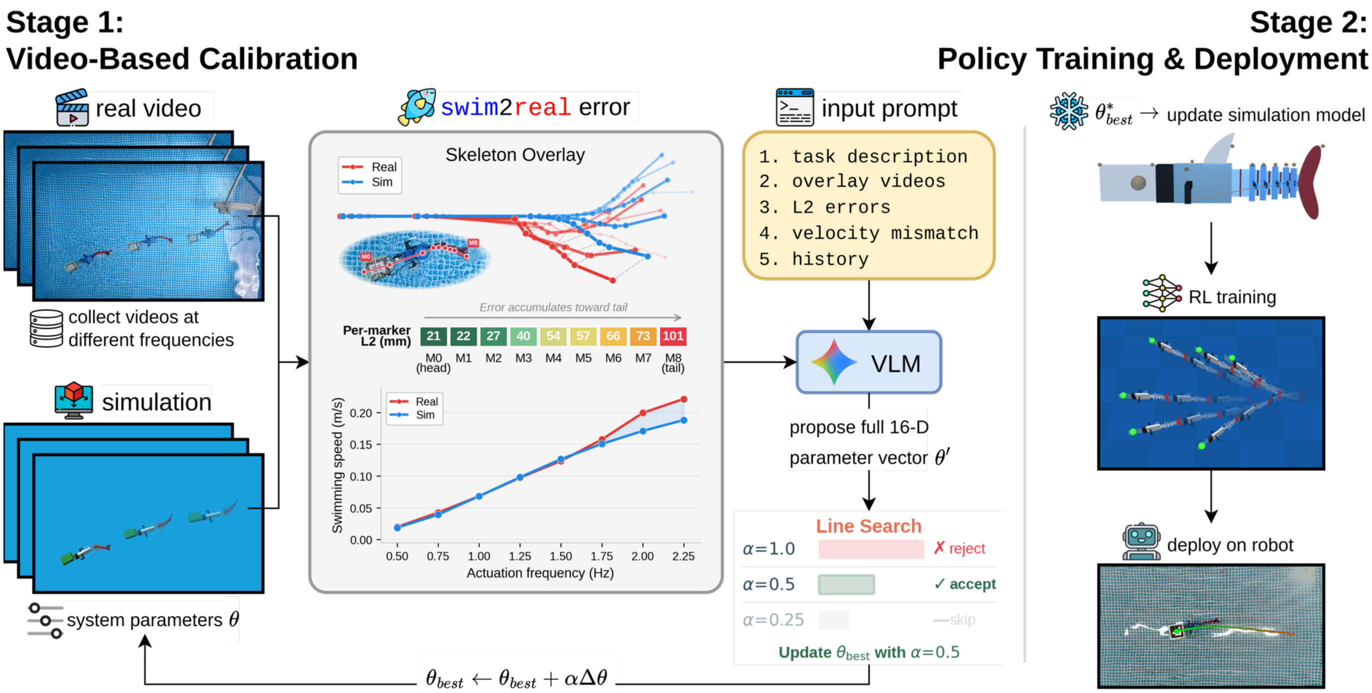

Fig. 1: SWIM2REAL calibrates a 16-parameter robotic fish simulator from video. **Stage 1**: a VLM compares simulated and real swimming videos, proposes parameter adjustments, and a backtracking line search validates the step size, iterating for up to 40 evaluations. **Stage 2**: the calibrated simulator trains an RL policy that transfers to the physical fish at 50 Hz.

***Abstract*—We present SWIM2REAL, a pipeline that calibrates a 16-parameter robotic fish simulator from swimming videos using a vision-language model (VLM) for feedback. Calibrating soft aquatic robots is hard because nonlinear fluid–structure coupling makes the parameter landscape chaotic, simplified fluid models introduce a persistent sim-to-real gap, and controlled aquatic experiments are difficult to reproduce. The VLM compares simulated and real swimming videos and proposes parameter updates with physical reasoning; a backtracking line search (Fig. 1) validates each step size, tripling the accept rate from 14% to 42% by recovering proposals whose direction is correct but whose magnitude overshoots. SWIM2REAL calibrates all 16 parameters simultaneously, matching real fish velocities across motor frequencies more closely than any other method (MAE = 7.4 mm/s, 43% lower than the next best), with zero outlier seeds across five runs. Downstream RL policies trained in the SWIM2REAL-calibrated simulator transfer open-loop to the physical fish at 50 Hz, swimming 12% farther than policies from BayesOpt-calibrated simulators and 90% farther than CMA-ES.**

I. INTRODUCTION

Synthetic data from physics simulators is the primary training source for RL-based robot control [1, 2], but policies transfer to hardware only when the simulator faithfully represents real dynamics. For aquatic soft robots [3, 4], simulator calibration is especially difficult. Hydrodynamic coefficients, joint stiffnesses, and motor parameters interact nonlinearly, creating parameter landscapes where small changes cause large behavioral shifts [5]. Prior work on this platform [6] required a three-stage manual pipeline (FFT-based stiffness fitting, grid search over motor geometry, then BO refinement of fluid coefficients) to handle this complexity.

SWIM2REAL replaces the entire manual pipeline with a VLM [7] that watches paired simulated and real swimming videos and proposes parameter corrections. A backtracking line search [8] validates each VLM proposal at decreasing step sizes, recovering proposals whose direction is correct but whose magnitude overshoots. The calibrated simulator produces synthetic RL training data that transfers directly to hardware: policies trained in the SWIM2REAL-calibrated simulator swim 12% farther than those from BayesOpt-calibrated simulators.

II. METHOD

A. VLM-in-the-Loop Calibration

At each iteration, the VLM receives paired sim-real swimming videos across 8 motor frequencies (0.5–2.5 Hz) and a structured prompt listing the current 16 parameters with their bounds. The VLM diagnoses discrepancies and proposes updated values with natural-language reasoning.

A representative first-round diagnosis: "*The simulation exhibits a critical failure at high frequencies (≥1.25 Hz), where the tail motion is severely attenuated, leading to near-complete loss of thrust. This 'locking up' behavior points to excessive damping forces that become dominant at high angular velocities.*"

B. Backtracking Line Search

Raw VLM proposals often have the correct direction but excessive magnitude, overshooting the optimum. We introduce a backtracking line search: given a proposal, we try the full step, then $0.5\times$, then $0.25\times$, accepting the first step that improves on the current best. This triples the accept rate from 14% to 42%.

Among accepted proposals, 39% use the full step, 33% use a half step, and 27% require a quarter step. This means 61% of successful updates would have been rejected without the line search. Each VLM call costs a mean of 2.5 simulation evaluations (line search overhead), but recovers $3\times$ more proposals.

C. Parameter Space

The fish simulator [9] has 16 tunable parameters: 5 hydrodynamic force coefficients, 1 motor arm length, 5 per-joint hinge stiffnesses, and 5 per-joint hinge dampings. The parameter landscape is chaotic: small stiffness changes (<5%) can shift swimming velocity by >20%.

III. RESULTS

A. Calibration Accuracy

Table I summarizes calibration performance. SWIM2REAL achieves 51.3 ± 1.2 mm with all five seeds falling within a 3 mm range (50.2–53.2 mm; Fig. 2). BayesOpt [10] reaches comparable final error (52.4 ± 2.1 mm) but SWIM2REAL converges faster (AUC 85.9 vs. 94.2 mm). CMA-ES [11] fails catastrophically on 2 of 5 seeds (worst: 254.2 mm) because population size 12 with budget 40 completes only ∼3 generations in 16 dimensions.

TABLE I: Calibration performance across 5 seeds. Best: mean ± std of per-seed final L2 marker error. AUC: mean best-so-far error over 40 evaluations. SWIM2REAL achieves the lowest error with zero outlier seeds.

Method	Best (mm) ↓	Worst (mm) ↓	AUC (mm) ↓
Random	82.0 ± 34.3	141.8	129.3 ± 26.6
CMA-ES	112.7 ± 83.9	254.2	156.4 ± 91.7
BayesOpt	52.4 ± 2.1	55.6	94.2 ± 17.6
SWIM2REAL (Ours)	$\mathbf{51.3 \pm 1.2}$	**53.2**	$\mathbf{85.9 \pm 30.9}$

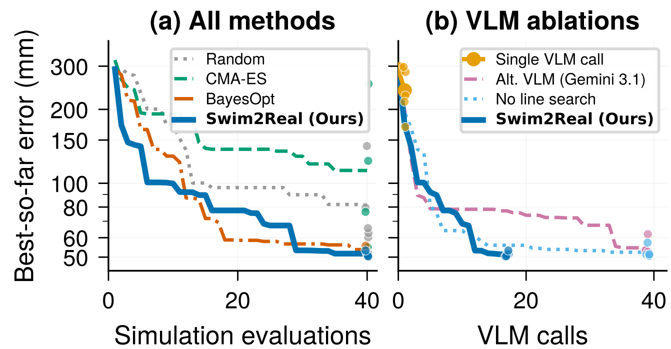

Fig. 2: Best-so-far L2 error (mean across 5 seeds, with dots showing individual seed final values). (a) All five SWIM2REAL seeds fall within 50.2–53.2 mm, while CMA-ES collapses on 2 of 5. (b) The line search triples the accept rate (42% vs. 14%), so SWIM2REAL reaches ∼51 mm in ∼16 VLM calls while the no-line-search ablation requires 39.

B. Downstream RL Transfer

SAC [12] policies trained in the SWIM2REAL-calibrated simulator transfer open-loop to the physical fish at 50 Hz. In forward-swimming evaluations, SWIM2REAL-calibrated policies reach 7.6 ± 0.0 m (3 seeds), compared to 6.8 ± 0.1 m for BayesOpt (+12%), 6.0 ± 0.0 m for random calibration, and 4.0 ± 0.6 m for CMA-ES (+90%). The ranking is monotonic with calibration accuracy: same RL algorithm, same training budget, same reward, but a more faithful simulator produces better hardware behavior.

C. Ablations

Removing the line search increases variance (52.4 ± 2.8 mm) and requires $2.5\times$ more VLM calls to reach similar accuracy. Replacing Gemini 2.5 Pro with Gemini 3.1 Pro yields 54.6 ± 4.2 mm, within seed variance, suggesting the approach is not tied to a specific model version. A single VLM call followed by random search (warm start) produces 91.1 ± 49.5 mm, confirming that iterative VLM feedback is essential.

D. Interpretable Reasoning

Unlike black-box calibration, SWIM2REAL outputs a written physical diagnosis at every iteration. The diagnosis in Section II-A correctly identifies the two dominant error sources (over-damping at high frequency, under-stiffness at low frequency). Acting on it, the VLM decreased fluid drag and

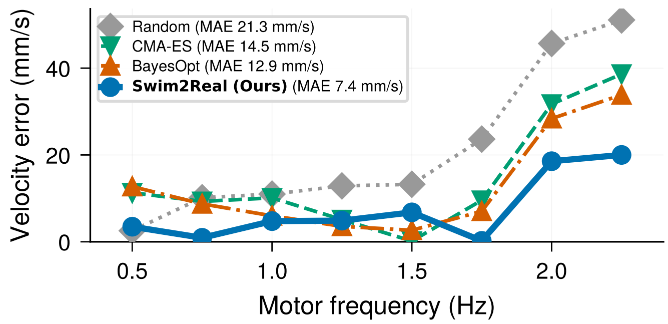

Fig. 3: Velocity error $|v_{\text{sim}} - v_{\text{real}}|$ across motor frequencies. SWIM2REAL tracks the real fish most closely (MAE = 7.4 mm/s, 43% lower than BayesOpt at 12.9 mm/s), with the gap widening at higher frequencies where calibration quality matters most.

increased hinge stiffness; the resulting proposal was accepted at β^0=1.0 and reduced error from 168.8 to 56.5 mm in a single step. A scalar objective cannot attribute an error drop to *which* physical phenomenon changed, which is exactly the signal a practitioner needs when deciding whether to tune further or revisit the model class.

IV. Related Work

Aquatic robot calibration. Michelis et al. [6] calibrate the same fish platform using a three-stage pipeline (coarse grid search, Bayesian optimization, CMA-ES refinement), requiring manual stage design and per-frequency error metrics. SWIM2REAL replaces all three stages with a single VLM-in-the-loop process that requires no hand-designed metrics or stage boundaries. Prior work on tendon-driven underwater robots [3, 4] demonstrated biological swimming behaviors but relied on extensive manual system identification.

Sim2real for soft robots. Domain randomization [1] and system identification [2] are the two main approaches to sim2real transfer. Domain randomization trains policies robust to parameter uncertainty but produces conservative behaviors; SWIM2REAL takes the complementary approach of calibrating the simulator precisely so that policies trained in it transfer directly without robustness margins. Model-based control of soft robots [5] faces the same simulator-fidelity bottleneck, where nonlinear fluid-structure coupling makes parameter identification difficult.

VLMs for physical reasoning. Recent work applies VLMs to robot planning, manipulation, and reward design. SWIM2REAL and its predecessor Vid2Sid [13] are the first to use VLMs for physics parameter estimation, using visual reasoning to diagnose which physical parameters cause observed sim-real discrepancies. The key distinction from scalar-objective sysid is that the VLM treats calibration as a reasoning task rather than a pure minimization.

V. Experimental Details

The platform is a tendon-driven robot fish [9], 0.6 m long, with a compliant tail driven by antagonistic tendons that cross at the midpoint to produce a bio-inspired S-bend. Eleven body markers are tracked via overhead camera (2160×3840, 60 fps) in a 2 m × 3 m pool. The simulator uses MuJoCo [14] at 1 kHz, discretizing the tail as five rigid hinge-connected segments with a stateless ellipsoid fluid model [6].

All methods use a budget of B=40 simulation evaluations across 5 random seeds with identical initializations. A single 8-frequency evaluation takes ∼13 s on one CPU core; each VLM call adds ∼42 s (rendering, upload, inference, parsing). Wall-clock runtime is 19 min for SWIM2REAL vs. 6 min for baselines. The VLM overhead roughly triples wall-clock time but remains minor relative to physical data collection.

VI. Discussion

SWIM2REAL takes a raw swimming video and produces a calibrated simulator ready for RL training, with no platform-specific error metrics or staged pipeline [13].

Out-of-objective validation. L2 marker error is the calibration objective; forward swimming velocity is not. Fig. 3 shows that SWIM2REAL-calibrated parameters also track real fish velocities more closely than every baseline (MAE 7.4 mm/s vs. 12.9 for BayesOpt, 14.5 for CMA-ES, 21.3 for random). This out-of-objective agreement argues that the simulator captures real dynamics rather than overfitting to the training metric through parameter cancellation, and the gap widens at higher frequencies where calibration quality matters most for locomotion control.

The 50 mm error floor. Per-marker analysis reveals a head-to-tail error gradient: head markers (M0–M2) average 23 mm while tail markers (M6–M8) average 80 mm, with the tail tip alone reaching 101 mm. The floor is dominated by the discretized kinematic chain, where five rigid hinge joints approximate a continuously bending tail and the approximation error accumulates along the chain. Closing the floor would require per-segment fluid coefficients or a higher-fidelity fluid model, a decision SWIM2REAL's diagnostics help practitioners make by surfacing exactly which joints drive the residual.

Implications for synthetic data. For a synthetic-data pipeline, the practical question is not "how low is the calibration error" but "does the calibrated simulator produce training data that transfers." The 12% and 90% forward-swimming gaps between methods show that small differences in marker error translate into much larger differences in downstream RL performance, because policy learning amplifies systematic simulator bias. SWIM2REAL's out-of-objective velocity match (Fig. 3) is a more reliable predictor of transfer quality than the L2 marker score it was optimized against, suggesting that sysid pipelines should validate against dynamics quantities the optimizer never saw.

Real-world deployment. Policies are deployed open-loop at 50 Hz without state feedback. The fish produces directed swimming toward targets, though real trajectories exhibit a consistent leftward arc absent in simulation. We attribute this to tendon-routing friction that creates a steering bias invisible to marker-based calibration, which operates in the fish's local body frame. Despite this bias, the deployed policy still drives the fish toward goals, confirming that the calibrated simulator captures enough body-dynamic fidelity for motor-command transfer.

Design rationale. The algorithmic simplicity is deliberate. The VLM provides structured reasoning about physical discrepancies, and the line search corrects the one systematic error (magnitude overestimation) that the VLM consistently makes. This parallels gradient-based optimization: the VLM replaces the gradient with visual-physical reasoning, and the line search determines step size. The 58% rejection rate (even with line search) reflects cases where the VLM adjusts too many parameters simultaneously, creating coupling effects. Analysis of rejected rounds reveals two dominant failure modes: the VLM conflates fluid coefficient effects with joint stiffness effects (both influence tail amplitude through different physical mechanisms), and motor arm length has no direct visual signature in the skeleton overlay, so adjustments to this parameter are guesses informed by overall thrust mismatch rather than a specific visual cue.

Limitations. The backtracking line search averages 2.5 simulation evaluations per VLM call, raising wall-clock time to 19 min vs. 6 min for baselines. The overhead is dominated by VLM inference latency ($\sim$42 s per call) and would shrink with on-device or cached inference. We tested Gemini 2.5 Pro and 3.1 Pro; how VLM scale affects calibration quality beyond this is an open question, as is whether smaller or open-source VLMs with lower latency could achieve comparable accuracy. The open-loop steering bias we observed is invisible to marker-based calibration because it operates in the fish's local body frame; closing the gap for directed control will require either closed-loop state feedback, tendon-level friction modeling, or a calibration objective that includes heading in the world frame.

VLM vs. black-box comparison. Beyond final accuracy, SWIM2REAL and BayesOpt differ in failure characteristics. BayesOpt's 5 seeds cluster tightly (52.4 ± 2.1 mm) because the Gaussian process smooths the search. CMA-ES collapses on 2 of 5 seeds because population-based search in 16 dimensions exhausts the evaluation budget before convergence. SWIM2REAL's seeds are the tightest (51.3 ± 1.2 mm) because the VLM's physical reasoning provides a structured initialization that avoids the catastrophic failures plaguing population-based methods. The practical implication: SWIM2REAL is the only method with zero outlier seeds, which matters for production pipelines where a single failed calibration wastes an entire data-generation run.

Takeaways for synthetic-data workflows. Three practical lessons emerge. First, for 16-dimensional chaotic parameter spaces, iterative reasoning with a line search outperforms single-pass generation or uncurbed gradient-free search at the same evaluation budget. Second, the wall-clock overhead of VLM inference is acceptable when physical data collection is the bottleneck, which is the usual case for aquatic platforms. Third, written diagnoses make the difference between "the calibration is stuck" and "the simulator is wrong," a distinction that determines whether the right next step is more compute or a new simulator.

References

[1] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2017, pp. 23–30.

[2] F. Muratore, F. Ramos, G. Turk, W. Yu, M. Gienger, and J. Peters, "Robot learning from randomized simulations: A review," *Frontiers in Robotics and AI*, vol. 9, p. 799893, 2022.

[3] A. D. Marchese, C. D. Onal, and D. Rus, "Autonomous soft robotic fish capable of escape maneuvers using fluidic elastomer actuators," *Soft Robotics*, vol. 1, no. 1, pp. 75–87, 2014.

[4] R. K. Katzschmann, J. DelPreto, R. MacCurdy, and D. Rus, "Exploration of underwater life with an acoustically controlled soft robotic fish," *Science Robotics*, vol. 3, no. 16, p. eaar3449, 2018.

[5] C. Della Santina, C. Duriez, and D. Rus, "Model-based control of soft robots: A survey of the state of the art and open challenges," *IEEE Control Systems Magazine*, vol. 43, no. 3, pp. 30–65, 2023.

[6] M. Y. Michelis, N. Obayashi, J. Hughes, and R. K. Katzschmann, "Simple models, real swimming: Digital twins for tendon-driven underwater robots," *arXiv preprint arXiv:2602.23283*, 2026.

[7] Gemini Team, R. Anil, S. Borgeaud *et al.*, "Gemini: A family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023.

[8] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. Springer, 2006.

[9] N. Obayashi, A. Anastasiadis, J. Gumowski, K. Junge, K. L. Walker, K. Mülleners, and J. Hughes, "ScaFi: Length-scalable, compliant, parametric robotic fish design for operation in multiple environmental niches," 2025.

[10] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 25, 2012.

[11] N. Hansen, "The CMA evolution strategy: A tutorial," *arXiv preprint arXiv:1604.00772*, 2016.

[12] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. Int. Conf. Machine Learning (ICML)*, 2018, pp. 1861–1870.

[13] K. Qiu, Y. Zhang, M. Cygan, and J. Hughes, "Vid2sid: Videos can help close the sim2real gap," *arXiv preprint arXiv:2602.19359*, 2026.

[14] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2012, pp. 5026–5033.