MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

PRISM: Bridging Multimodality and Real-Time Control in Multisensory Policies

Bhaskar, Amisha; Tokekar, Pratap; Di Cairano, Stefano; Schperberg, Alexander

TR2026-169 November 10, 2026

Abstract

Robotic imitation learning requires policies that capture multimodal action distributions, run at real-time control rates, and exploit available sensors. Existing generative approaches, including diffusion, flow matching, and IMLE, satisfy these requirements only partially. We present PRISM, a single-pass multi-sensory policy built on a calibrated batch-global RS–IMLE objective. We show that the RS–IMLE rejection mechanism, originally designed for few-shot image synthesis, degenerates at policy-scale batch sizes, and introduce the calibration needed to restore effective rejection. PRISM couples a temporal multisensory encoder (RGB, depth, tactile, audio, proprioception) with a linear-attention Performer generator trained under this objective. On real-world loco-manipulation (Unitree Go2 + 7-DoF arm) and tabletop manipulation (UR5), PRISM improves success over diffusion, flow, and IMLE baselines by 10–25 pp while sustaining 30-50 Hz closed-loop control. On language-conditioned CALVIN ABCD, it improves success by about 31 percentage points over diffusion and about 24 pp over flow matching, while reducing trajectory jerk by 20x–70x.

Conference on Robot Learning (CoRL) 2026

© 2026 MERL. This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

PRISM: Bridging Multimodality and Real-Time Control in Multisensory Policies

Amisha Bhaskar
Amazon Robotics
Sunnivale, CA
amishii@amazon.com

Pratap Tokekar
Amazon Robotics
North Reading, MA
tokekar@amazon.com

Stefano Di Cairano
Mitsubishi Electric Research Laboratories (MERL)
Cambridge, MA
dicairano@merl.com

Alexander Schperberg
Mitsubishi Electric Research Laboratories (MERL)
Cambridge, MA
schperberg@merl.com

Abstract: Robotic imitation learning requires policies that capture multimodal action distributions, run at real-time control rates, and exploit available sensors. Existing generative approaches, including diffusion, flow matching, and IMLE, satisfy these requirements only partially. We present PRISM, a single-pass multisensory policy built on a calibrated batch-global RS–IMLE objective. We show that the RS–IMLE rejection mechanism, originally designed for few-shot image synthesis, degenerates at policy-scale batch sizes, and introduce the calibration needed to restore effective rejection. PRISM couples a temporal multisensory encoder (RGB, depth, tactile, audio, proprioception) with a linear-attention Performer generator trained under this objective. On real-world loco-manipulation (Unitree Go2 + 7-DoF arm) and tabletop manipulation (UR5), PRISM improves success over diffusion, flow, and IMLE baselines by 10–25 pp while sustaining 30–50 Hz closed-loop control. On language-conditioned CALVIN ABCD, it improves success by $\sim$31 percentage points over diffusion and $\sim$24 pp over flow matching, while reducing trajectory jerk by $20\times$–$70\times$.[1]

Keywords: Imitation Learning, Multisensory Policies

1 Introduction

Learning robot policies from demonstrations [1, 2, 3, 4, 5, 6] requires policies that are multimodal, reactive, and robust to heterogeneous sensing. Mean-squared behavior cloning is computationally efficient but collapses distinct demonstrated behaviors into averaged actions. Diffusion and flow-based policies [7, 8, 9, 10, 11, 12, 13] better represent multimodal action distributions, but their iterative sampling procedures increase latency and can limit closed-loop control frequency [14, 15, 16]. IMLE and RS–IMLE [17, 18, 19] provide a promising single-pass alternative, yet their rejection mechanism was developed in regimes that differ substantially from policy learning, particularly in batch size, action scale, and conditional trajectory structure.

[1] Project page with videos: https://anonymous227111-png.github.io/

10th Conference on Robot Learning (CoRL 2026), Austin TX, USA.

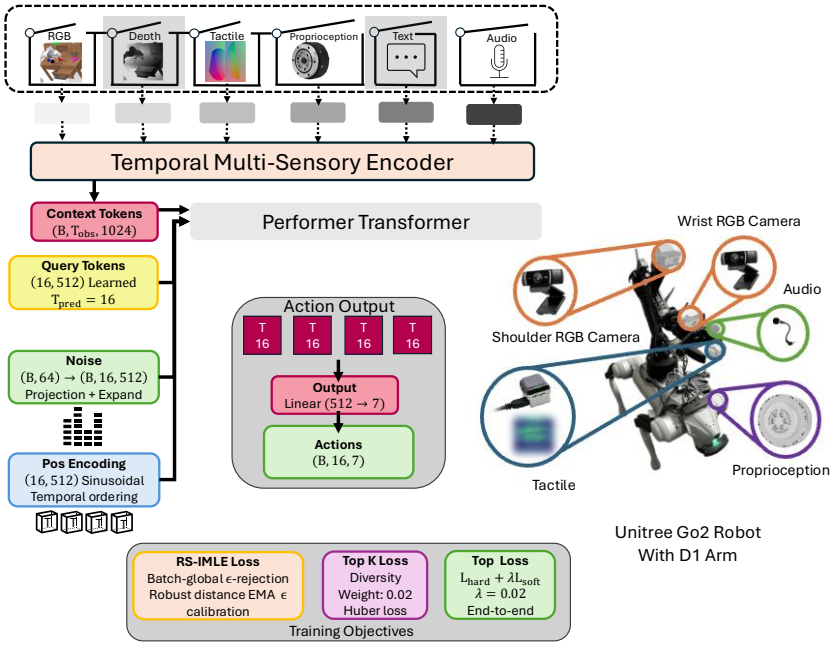


Figure 1: **Overview of Performer RS-IMLE for Multisensory Imitation (PRISM).** PRISM fuses per-timestep multisensory features into temporal context tokens, predicts an action sequence with a bidirectional FAVOR$^+$ generator in one forward pass, and trains with a batch-global RS–IMLE objective whose rejection threshold is calibrated by an EMA statistic.

We show that directly applying RS–IMLE to policy-scale training can produce a degenerate rejection process: candidates are either almost never rejected or almost always rejected, while training losses still appear well behaved. This failure arises from two sources: self-matches in the candidate-target distance matrix and rejection thresholds calibrated to image-generation distance scales rather than action-trajectory distances. To address this, we introduce **Performer RS-IMLE for Multisensory Imitation (PRISM)**, a single-pass multisensory imitation policy trained with a calibrated batch-global RS–IMLE objective. The method excludes diagonal self-matches, rescales the rejection threshold using an EMA-calibrated clamp, and includes a one-pass audit to detect rejection degeneracy before full training. Our contributions are:

Calibrated RS–IMLE for policy learning. We identify two failure modes that arise when RS–IMLE is scaled to large-batch conditional imitation learning and introduce diagonal exclusion, EMA-calibrated rejection thresholds, and a pre-training audit for detecting rejection collapse (Sec. 4.4, App. C.4). Proposition F.2 bounds within-mode candidate suppression under the calibrated objective.

Single-pass multisensory policy architecture. We combine the calibrated objective with per-timestep multisensory fusion, a bidirectional Performer/FAVOR$^+$ generator, and receding-horizon ProxyScore selection. The resulting policy predicts full action chunks in one forward pass and runs at 30–50 Hz on real hardware (Sec. 4.4).

Evaluation across simulation and hardware. We evaluate PRISM on MetaWorld [20], CALVIN ABCD [21], Robomimic PH [22], and real loco-manipulation and tabletop tasks. PRISM improves over diffusion, flow, and IMLE baselines under matched single-pass evaluation, while matching or exceeding AdaFlow on near-unimodal Robomimic PH, indicating that calibrated multimodal training does not penalize accuracy when diversity is limited.

2 Related Work

Behavior cloning and multimodal actions. Behavior cloning provides fast single-pass control by mapping observations directly to actions [23, 24], but unimodal regression losses can average distinct expert strategies in contact-rich tasks. Prior work improves expressivity through discretized action prediction [25, 26], mixture or transformer-based behavior models [22, 27], and implicit energy-based policies [28]. These methods motivate multimodal action distributions over mean actions. Closest to PRISM's architecture, ACT [29] predicts action chunks non-autoregressively via a transformer with learned action query tokens; PRISM differs by replacing the CVAE prior with batch-global RS–IMLE for explicit multimodality control and by using linear attention for latency scaling.

Generative policy learning and IMLE. Diffusion and flow-based policies model complex visuomotor action distributions [7, 9, 30], with recent accelerations based on equivariant transformers, adaptive solvers, distillation, or step reuse [11, 10, 31, 15]. However, reducing the number of sampling steps can make smooth execution and mode coverage harder. Single-step GAN-based policies avoid iterative sampling but can suffer from mode collapse [32, 33]. IMLE instead trains a generator by matching each data point to a nearby generated sample [17], and RS–IMLE improves sample selection through rejection sampling [18]. PRISM extends this to conditional robot imitation with batch-global rejection over minibatch targets.

Multisensory and temporally coherent policies. Robot manipulation benefits from combining RGB, depth, tactile, proprioceptive, and other sensory cues under partial observability and contact uncertainty [4, 5]. At the same time, multimodal policies can switch between valid behaviors during execution, producing jitter or inconsistent action chunks [34, 35]. Attention-based sequence policies can model temporal context, and linear-time variants such as Performer make longer contexts more efficient [36]. PRISM uses per-timestep multisensory fusion with bidirectional linear attention and receding-horizon candidate selection for coherent real-time action sequences.

3 Problem Statement

Setting. We consider imitation learning from demonstrations with temporally aligned multisensory observations. Each episode provides synchronized sensor streams comprising one or more RGB cameras (always wrist; optionally static view), and optionally depth, tactile, proprioception, and audio. Let $\mathcal{D} = \{(\mathbf{O}^{(n)}, \mathbf{A}^{(n)})\}_{n=1}^{N}$ denote N episodes, where $\mathbf{O}^{(n)}$ and $\mathbf{A}^{(n)}$ are sequences of observations and actions.

Observations and actions. At discrete time t, the observation $\mathbf{o}_t$ concatenates all available sensors. The action is $\mathbf{a}_t \in \mathbb{R}^{D_a}$ with each dimension normalized to $[-1, 1]$, where $D_a = 7$ for tabletop manipulation (end-effector deltas + gripper) and $D_a = 14$ for loco-manipulation (absolute pose + base velocities + gripper). Full parameterization details are in Appendix A.2.

Windows. For observation horizon T_o and prediction horizon T_p:

$$\mathbf{O}_{t-T_o+1:t} = \{\mathbf{o}_{t-T_o+1}, \ldots, \mathbf{o}_t\},$$
$$\mathbf{A}_{t+1:t+T_p} = \{\mathbf{a}_{t+1}, \ldots, \mathbf{a}_{t+T_p}\}.$$

Our goal is to learn a single-forward-pass, multimodal policy:

$$\hat{\mathbf{A}}_{t+1:t+T_p} \sim \pi_\theta(\,\cdot \mid \mathbf{O}_{t-T_o+1:t}, z\,), \tag{1}$$

where $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a latent variable.

Assumptions. (i) Sensors are time-aligned; missing modalities may occur. (ii) Actions are bounded and normalized. (iii) Demonstrations are multimodal: multiple valid action sequences may correspond to the same context.

4 Proposed Approach

PRISM is a single-pass imitation policy for multimodal robot control. Given a short history of observations, the model predicts a future action sequence in one forward pass and executes it in a receding-horizon loop. The method has three components: (i) a temporal multisensory encoder that keeps the observation history ordered in time, (ii) a bidirectional linear-attention generator that produces full action chunks non-autoregressively, and (iii) a batch-global RS–IMLE objective that trains the generator to cover multiple demonstrated behaviors without iterative sampling. Figure 1 gives the full architecture. We keep only the core equations in the main paper; training and inference pseudocode are in Appendix.

4.1 Temporal multisensory context

Let T_o denote the observation horizon and d the model width. At each timestep t, available sensor streams are encoded into modality embeddings $\{\mathbf{v}_t^{(m)}\}_{m\in\mathcal{M}_t}$, where $\mathcal{M}_t$ may include wrist RGB, static RGB, depth, tactile, proprioception, audio, or text depending on the benchmark. We fuse modalities independently at each timestep,

$$\mathbf{c}_t = \mathrm{MLP}\left(\left[\mathbf{v}_t^{(m)}\right]_{m\in\mathcal{M}_t}\right) \in \mathbb{R}^d, \tag{2}$$

and stack the resulting tokens into $\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_{T_o}] \in \mathbb{R}^{T_o\times d}$ with learned positional embeddings. This design preserves the temporal order of sensory evidence before action generation. Dataset-specific modalities and encoder details are listed in Appendix A.1 and Appendix B.2.

4.2 Single-pass trajectory generator

The generator maps the context tokens $\mathbf{C}$ and a latent variable $z \sim \mathcal{N}(0, I)$ to an action sequence $\hat{\mathbf{A}} \in \mathbb{R}^{T_p\times D_a}$. We initialize T_p learned action query tokens, add positional embeddings and a projected latent embedding, and process them with bidirectional self-attention over the action queries and cross-attention to $\mathbf{C}$. The generator is non-autoregressive: all action timesteps are predicted jointly, coordinating early and late parts of the trajectory in one forward pass. Both self-attention and cross-attention use FAVOR$^+$ linear attention [36]. This keeps the trajectory generator efficient as the context and prediction horizons grow, reducing attention cost from quadratic to linear in sequence length for a fixed random-feature budget. The stabilized implementation is in Algorithm 1 (App. B.2).

4.3 Batch-global RS–IMLE training

For each training example i in a minibatch of size B, we sample K latents $\{z_{i,k}\}_{k=1}^K \sim \mathcal{N}(0, I)$ and generate candidates $\hat{\mathbf{A}}_i^{(k)} = G_\theta(\mathbf{C}_i, z_{i,k})$ in a single batched forward pass. We compare action sequences using a scale-normalized Charbonnier distance,

$$D_\rho(\hat{\mathbf{A}}, \mathbf{A}) \;=\; \frac{1}{T_p}\sum_{t=1}^{T_p}\sum_{d=1}^{D_a} w_d\sqrt{(\hat{a}_{t,d} - a_{t,d})^2 + \varepsilon_c^2}, \tag{3}$$

where w_d normalizes action dimensions and $\varepsilon_c = 10^{-6}$. The same distance is used for training and candidate scoring.

Standard conditional IMLE selects the best generated candidate for each paired target [17, 19]. PRISM instead compares each candidate against *every non-paired* target in the minibatch and rejects candidates that already lie within $\varepsilon_{\mathrm{RS}}$ of one. Let

$$\tilde{D}_{i,k\to j} \;=\; \begin{cases} \infty & \text{if } j = i, \\ D_\rho\left(\hat{\mathbf{A}}_i^{(k)}, \mathbf{A}_j\right) & \text{otherwise,} \end{cases} \tag{4}$$

denote the off-diagonal candidate-to-target distance tensor. A candidate is rejected if it is too close to any non-paired target:

$$\mathbb{I}_{i,k}^{\mathrm{rej}} \;=\; \mathbb{I}\left[\min_j \tilde{D}_{i,k\to j} \;<\; \varepsilon_{\mathrm{RS}}\right]. \tag{5}$$

The diagonal exclusion in Eq. 4 is essential: if the paired target $\mathbf{A}_i$ were included in the $\min_j$, its distance would by construction dominate and the rejection mask would saturate, making Eq. 5 uninformative (Sec. 4.4).

The hard training loss selects the best non-rejected candidate per example,

$$\mathcal{L}_{\text{hard}} \;=\; \frac{1}{B}\sum_{i=1}^{B} \min_{k\in\mathcal{K}_i} D_\rho\Big(\hat{\mathbf{A}}_i^{(k)}, \mathbf{A}_i\Big), \quad \mathcal{K}_i \;=\; \begin{cases} \{k : \mathbb{I}^{\text{rej}}_{i,k} = 0\}, & \text{if non-empty,} \\ \{1,\dots,K\}, & \text{otherwise.} \end{cases} \tag{6}$$

The fallback avoids dropping the sample when all candidates fall inside the rejection region.

We set ε_{RS} by an exponential moving average of a batch-global quantile over the off-diagonal distance tensor,

$$\tilde{\varepsilon} \;=\; \text{Quantile}_q\Big(\{\tilde{D}_{i,k\to j}\}_{i\neq j,\,k}\Big), \quad \varepsilon_{\text{RS}} \;\leftarrow\; \text{clip}(\alpha\varepsilon_{\text{RS}} + (1{-}\alpha)\tilde{\varepsilon},\; \varepsilon_{\min},\; \varepsilon_{\max}). \tag{7}$$

We use $q = 0.005$, $\alpha = 0.9$, and clamps $(\varepsilon_{\min}, \varepsilon_{\max}) = (10^{-4}, 2.5)$. These values differ from the image-synthesis defaults of Vashist et al. [18]; the rationale is in Sec. 4.4 and an empirical audit across batch sizes in App. C.4.

We additionally use a small top-K' soft-coverage term,

$$\mathcal{L}_{\text{soft}} \;=\; -\frac{1}{B}\sum_{i=1}^{B} \log \sum_{k\in\text{TopK}'(D_{i,\cdot})} \exp(-D_{i,k}/\tau), \tag{8}$$

and optimize $\mathcal{L} = \mathcal{L}_{\text{hard}} + \lambda_{\text{soft}}\mathcal{L}_{\text{soft}}$ with $\lambda_{\text{soft}} \ll 1$. The soft-coverage interpretation and ablations are in App. F.4 and App. C.2.

4.4 Threshold calibration and receding-horizon inference

The policy-scale regime ($B{\approx}96{-}128$) differs from the image-synthesis setting of Vashist et al. [18] ($B{\approx}4{-}16$). We verify this by instrumenting the rejection rate ρ_t throughout training on CALVIN. Under the image-synthesis default $\varepsilon_{\max}{=}0.2$, the clamp pins the threshold an order of magnitude below the action-distance regime (25th percentile ${\approx}1.4$) and $\rho_t{=}0$ at every step: rejection never fires. Raising the clamp to $\varepsilon_{\max}{=}2.5$ without excluding self-targets (Eq. 4) saturates the mask at $\rho_t{=}0.76{-}1.00$, since the paired-target distance dominates the per-candidate minimum. Only with both fixes ($\varepsilon_{\max}{=}2.5$ and diagonal exclusion) does ρ_t stabilize at $0.18{-}0.32$ with $q{=}0.005$, the operating regime used for all results in this paper (App. C.4; Lemma F.1 establishes $\mathcal{O}(1/N)$ variance of the quantile estimator). Proposition F.2 (App. F.3) shows that within-mode candidate suppression scales as $O(B\pi_k q)$ per candidate, which remains small for the operating values $q{=}0.005$ and $B{=}128$ used throughout this paper.

Inference. At test time, PRISM encodes the latest observation window and generates K candidates in one batched forward pass. When proprioception or end-effector pose is available, a single candidate is selected by an observation-only ProxyScore that favors continuity with the previously executed action $\mathbf{a}_{t-1}$,

$$k^\star = \arg\min_k \Big\|\hat{\mathbf{a}}_1^{(k)} - \mathbf{a}_{t-1}\Big\|_2, \tag{9}$$

otherwise a deterministic smoothness tie-break is used. The selected chunk is executed for T_a steps before the window slides and the policy replans (Algorithm 3, App. B.5).

5 Experiments

We evaluate PRISM on MetaWorld [20], CALVIN (full ABCD, language-conditioned) [21], Robomimic PH [22], and two real-robot platforms. Baselines include Diffusion Policy [7], Flow Matching [37], and IMLE Policy [19], reimplemented or reproduced under matched modality, control rate, action parameterization, and data budget (details in App. A.3). On CALVIN we report

per-instruction success averaged over all ABCD language annotations. All simulation results are mean±s.e. over 3 seeds unless noted. Full protocols, modality configurations, and per-task tables are in Appendix A–D. We organize experiments around: isolating the calibration finding from architectural and execution contributions (Sec. 5.1), benchmark comparison (Sec. 5.2–5.3), multisensory robustness and latency (Sec. 5.4–5.5), and hardware transfer (Sec. 5.6).

5.1 Where do PRISM's gains come from?

We address attribution through three controlled comparisons: backbone × objective at matched data and modality (Table 1, left), calibration isolation via the $\varepsilon_{\max}$=0.2 row (Table 1, left), and execution-strategy isolation at matched T_a (Table 1, right); each contributes, none alone. Crossing backbones with objectives (Table 1, left), UNet+MSE-BC reaches 44.6% and Performer+MSE reaches 48.4%, while batch-global RS–IMLE on UNet alone reaches 50.9%; the Performer combined with batch-global RS–IMLE yields 67.9%, exceeding the 54.7% predicted by summing the two effects in isolation by 13.2 points (super-additive), with jerk falling over 30× from 1.666 to 0.052. Crucially, running the same Performer + batch-global configuration with the image-synthesis default $\varepsilon_{\max}$=0.2 reaches only 54.9%, confirming that the calibration corrections in Sec. 4.4 are necessary. The temporal encoder is independently necessary: per-timestep multisensory fusion reaches 67.9% versus 58.6% for early fusion and 50.4% without temporal modeling. Execution strategy stacks on top (Table 1, right): ProxyScore selection adds 5.1% over random candidate selection at matched T_a=8, and shorter receding-horizon chunks lift success to 79.8% at T_a=4 and 83.6% at T_a=1; we report T_a=8 in cross-method comparisons because baselines use the same execution horizon; the T_a=1 row characterizes the execution ceiling.

Table 1: **Attribution on CALVIN.** Left: backbone × objective (jerk over predicted chunks). Right: execution strategy (jerk over executed trajectories; lower T_a replans more often, reducing jerk).

Backbone	Objective	Succ.↑	Jerk↓
UNet	MSE BC	44.6	1.666
UNet	Per-sample RS–IMLE	46.7	4.708
UNet	Batch-global RS–IMLE	50.9	1.129
Performer	MSE BC	48.4	0.920
Performer	Per-sample RS–IMLE	54.6	0.658
Performer	BG RS–IMLE ($\varepsilon_{\max}$=0.2)	54.9	0.670
Performer	Batch-global RS–IMLE	**67.9**	**0.052**

Execution strategy	T_a	Succ.↑	Jerk↓
Open-loop, execute full chunk	16	62.3	0.819
Random candidate selection	8	62.8	0.760
ProxyScore selection	8	67.9	0.613
ProxyScore selection	4	79.8	0.198
ProxyScore selection	1	83.6	0.000

5.2 How does PRISM compare to diffusion, flow, and prior IMLE policies?

Across three benchmarks spanning the multimodality spectrum (Table 2), PRISM consistently improves over the strongest single-pass baselines. All CALVIN methods receive identical language conditioning (frozen CLIP text embedding + parsed directional token), so the comparison isolates the policy objective rather than the text features. On CALVIN, PRISM improves success by 11.7% over IMLE Policy, 23.5% over Flow Matching, and 31.5% over Diffusion Policy, while jerk drops 20× relative to IMLE and 70× relative to Diffusion. On MetaWorld, PRISM leads across all difficulty groups (Easy +20.8%, Medium +25.1%, Hard +14.5%, Very-Hard +9.8% over IMLE Policy), with the largest gains on the splits where demonstrations contain branching strategies and contact-rich phases. PRISM leads on group averages despite using RGB while some baselines (DP3, FlowPolicy) use 3D point clouds; per-task variation favoring point-cloud methods on tasks like Disassemble is documented in App. D.2. On near-unimodal Robomimic PH, PRISM reaches 97.6% with one-step inference (NFE=1), matching or exceeding AdaFlow (95.6%, NFE≈1.2), the prior best on this benchmark. Batch-global RS–IMLE does not pay an accuracy tax when diversity is unnecessary.

Table 2: **Main benchmark results** across CALVIN (ABCD, language-conditioned; all methods use identical CLIP text + directional-token conditioning), MetaWorld (50 tasks by difficulty), and Robomimic PH.

	CALVIN		MetaWorld				Robomimic PH	
Method	Succ.↑	Jerk↓	Easy↑	Med.↑	Hard↑	V-Hard↑	NFE↓	Avg.↑
Diffusion Policy [7]	36.4±9.2	3.783	83.6	31.1	9.0	26.6	15	0.850
Flow Matching [37]	44.4±3.7	1.148	61.4	20.6	13.4	36.0	1	0.852
IMLE Policy [19]	56.2±6.5	1.053	75.6	60.4	43.5	76.0	1	0.870
PRISM	**67.9±4.3**	**0.052**	**96.4**	**85.5**	**58.0**	**85.8**	**1**	**0.976**

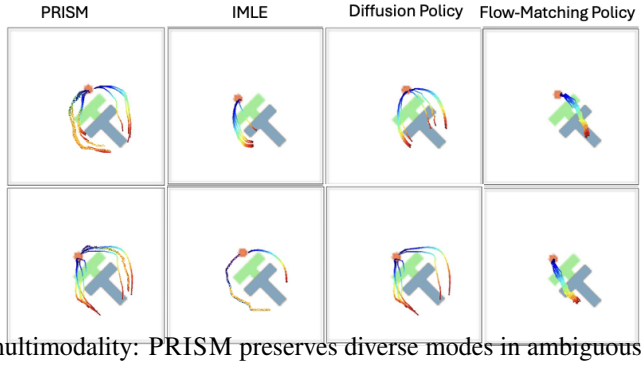


Figure 2: Push-T multimodality: PRISM preserves diverse modes in ambiguous regions, while Flow Matching and IMLE Policy collapse toward unimodal outputs.

5.3 Does PRISM actually preserve multimodal behavior?

We test whether batch-global rejection preserves multimodal coverage using the Push-T diagnostic from Rana et al. [19], sweeping the end-effector start position across the top edge of the T-block where demonstrations branch left or right (Fig. 2). PRISM maintains diverse but smooth trajectory families in the ambiguous central states; Flow Matching collapses to a single trajectory family regardless of start position; IMLE Policy retains partial diversity but with higher trajectory jitter, consistent with the per-sample-vs-batch-global gap in Table 1.

5.4 Is PRISM robust to sensor dropout?

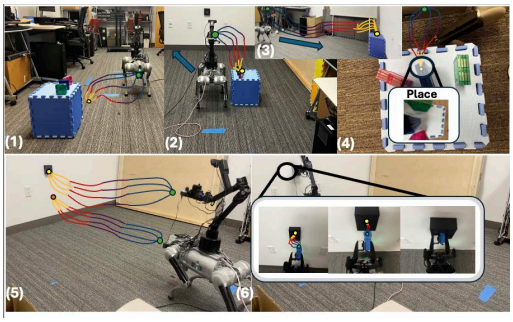
PRISM degrades gracefully under sensory dropout applied at evaluation time. Starting from 67.9% with all modalities on CALVIN, removing depth, static RGB, or tactile individually has modest effect (59–67%), while removing wrist RGB drops success to 41.5% and removing proprioception drops it to 15.8%; removing wrist RGB and proprioception together collapses performance (0%). Full single- and pairwise-dropout results are in Appendix C.1 (Fig. 5). The pattern is mechanistically interpretable: wrist RGB carries local interaction geometry during approach and contact; proprioception anchors the base-frame trajectory and supplies ProxyScore input; depth becomes redundant at close range once wrist RGB resolves contact geometry. Wrist RGB and proprioception form the critical sensing backbone; remaining modalities are auxiliary.

5.5 How fast is PRISM relative to iterative samplers?

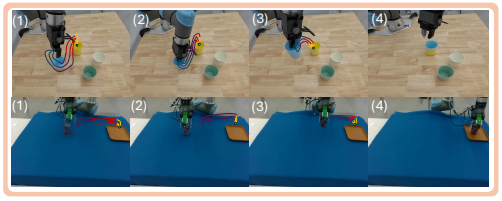
At matched NFE (Table 3), PRISM and Flow Matching are comparably fast at ∼15 ms (both single-pass), while Diffusion Policy at NFE=15 requires 213.5 ms, missing the 30 Hz budget by 6×. The key difference is that Flow Matching at NFE=1 sacrifices multimodal coverage for speed (Sec. 5.3), whereas PRISM retains both. The case for FAVOR$^+$ is latency scaling:

Table 3: Inference latency (ms), A100, T_o=8, T_p=16.

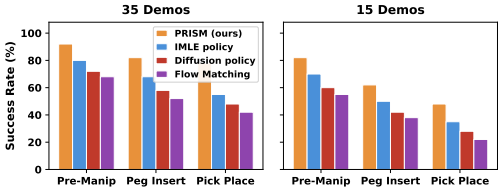
Method	NFE	Mean	p99
Diffusion Policy	15	213.5	252.8
Flow Matching	1	14.8	17.9
PRISM (K=8)	**1**	**15.0**	**16.0**



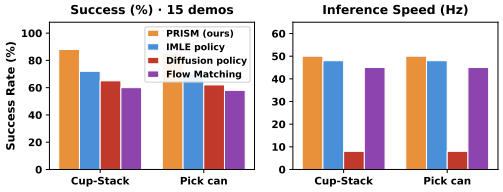
(a) Loco-manipulation platform



(b) Tabletop manipulation setup



(c) Loco-manipulation success rates



(d) Tabletop success & inference speed

Figure 3: PRISM improves success by 10–25 pp over baselines across loco-manipulation (a,c) and tabletop (b,d) tasks while maintaining real-time closed-loop inference (50 trials per method).

softmax attention is quadratic in horizon length, and
PRISM's 30–50 Hz hardware control (Sec. 5.6) would not be feasible under quadratic attention at deployment horizons. FAVOR$^+$ matches softmax accuracy on CALVIN (within $\sim$5 pp, App. C.2) while remaining linear.

5.6 Do simulation gains transfer to hardware?

The simulation trends transfer to hardware on both a Unitree Go2 with a D1 arm and a UR5 tabletop platform, deployed with T_o=8, T_p=16, T_a=8, K=8 across pre-manipulation parking, peg-in-hole insertion, multi-object pick-and-place, cup stacking, and can picking (Figure 3; 50 trials per method per task). PRISM improves success by 10–25 pp over baselines. Performance also scales with demonstration count: increasing from 15 to 35 demos adds $+$10 pp on parking, $+$20 pp on peg-in-hole, and $+$30 pp on pick-and-place. Tabletop results show the same trend. On the deployment RTX 5090, PRISM sustains 30–50 Hz end-to-end closed-loop control (forward pass plus preprocessing and camera sync), consistent with the A100 benchmarks in Table 3.

6 Limitations

PRISM has five limitations that bound the claims in this paper. First, the EMA-calibrated rejection threshold ε_{RS} is stable for $B \geq 64$ (Lemma F.1, Fig. 8) but its absolute scale depends on the action-distance distribution of the dataset; we provide a one-pass audit (Sec. 4.4, App. C.4) that diagnoses the two failure modes we encountered, and a fully dataset-agnostic rule (e.g., local k-NN density rather than a global quantile) is a natural next step. Second, we train visual encoders from scratch to keep the architecture-versus-objective attribution in Table 1 clean; integrating pretrained backbones (DINOv2, R3M) is expected to improve absolute performance and is straightforward as a drop-in encoder swap. Third, despite Robomimic PH being near-unimodal (where batch-global rejection contributes minimal diversity benefit), PRISM matches or exceeds AdaFlow (0.976 vs. 0.956, Table 15), indicating the objective and architecture do not trade off accuracy for diversity; however, PH remains a limited stress test for the multimodal mechanism, and richer multi-strategy benchmarks would be more informative. Fourth, ProxyScore requires observable proprioception or end-effector pose; without it, PRISM falls back to a deterministic smoothness tie-break that captures most but not all of the closed-loop benefit (Table 1, right), and history-aware or uncertainty-aware candidate selection

is a clear direction suggested by our failure analysis (App. G). Fifth, our hardware experiments use 15–35 demonstrations per task on two embodiments (Unitree Go2+D1 and UR5), validating simulation-to-hardware transfer but not generalization at scale; cross-embodiment transfer, larger demonstration budgets, and structured training-time modality dropout (we currently test dropout only at evaluation time) are the most informative next-step experiments.

7 Conclusion

We presented PRISM, a single-pass multisensory policy that generates full action sequences with bidirectional Performer attention and preserves multimodal coverage via batch-global RS–IMLE. Our central finding is that calibrated batch-global rejection, with EMA-calibrated thresholds and diagonal exclusion, is what makes single-pass IMLE viable at policy-scale batches; the Performer backbone amplifies it super-additively (Table 1). Across MetaWorld, CALVIN, and real loco-manipulation and tabletop suites, PRISM improves success by **11–31 pp** in simulation and **10–25 pp** on hardware while cutting trajectory jerk by up to $70\times$ at real-time rates. Future work includes adaptive rejection control, learned feature maps for linear attention, and RL fine-tuning.

Acknowledgments

Pratap Tokekar holds concurrent appointments as an Associate Professor in the Department of Computer Science at the University of Maryland and as an Amazon Scholar. This paper describes work performed at Amazon and is not associated with the University of Maryland or Northeastern University. This paper describes work performed at the University of Maryland and is not associated with Amazon.

References

[1] C. G. Atkeson and S. Schaal. Robot learning from demonstration. In *ICML*, volume 97, pages 12–20, 1997.

[2] B. D. Argall, S. Chernova, M. Veloso, and B. Browning. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483, 2009.

[3] R. Rahmatizadeh, P. Abolghasemi, L. Bölöni, and S. Levine. Vision-based multi-task manipulation for inexpensive robots using end-to-end learning from demonstration. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3758–3765. IEEE, 2018.

[4] Y. Avigal, L. Berscheid, T. Asfour, T. Kröger, and K. Goldberg. Speedfolding: Learning efficient bimanual folding of garments. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–8. IEEE, 2022.

[5] P. Yu, A. Bhaskar, A. Singh, Z. Mahammad, and P. Tokekar. Sketch-to-skill: Bootstrapping robot learning with human drawn trajectory sketches. *arXiv preprint arXiv:2503.11918*, 2025.

[6] A. Bhaskar, R. Liu, V. D. Sharma, G. Shi, and P. Tokekar. Lava: Long-horizon visual action based food acquisition. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8929–8935. IEEE, 2024.

[7] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[8] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. *arXiv preprint arXiv:2403.03954*, 2024.

[9] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

[10] X. Hu, Q. Liu, X. Liu, and B. Liu. Adaflow: Imitation learning with variance-adaptive flow-based policies. *Advances in Neural Information Processing Systems*, 37:138836–138858, 2024.

[11] N. Funk, J. Urain, J. Carvalho, V. Prasad, G. Chalvatzaki, and J. Peters. Actionflow: Equivariant, accurate, and efficient policies with spatially symmetric flow matching. *arXiv preprint arXiv:2409.04576*, 2024.

[12] Q. Zhang, Z. Liu, H. Fan, G. Liu, B. Zeng, and S. Liu. Flowpolicy: Enabling fast and robust 3d flow-based policy via consistency flow matching for robot manipulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 14754–14762, 2025.

[13] L. Yang, Z. Zhang, Z. Zhang, X. Liu, M. Xu, W. Zhang, C. Meng, S. Ermon, and B. Cui. Consistency flow matching: Defining straight flows with velocity consistency. *arXiv preprint arXiv:2407.02398*, 2024.

[14] G. Lu, Z. Gao, T. Chen, W. Dai, Z. Wang, W. Ding, and Y. Tang. Manicm: Real-time 3d diffusion policy via consistency model for robotic manipulation. *arXiv preprint arXiv:2406.01586*, 2024.

[15] A. Prasad, K. Lin, J. Wu, L. Zhou, and J. Bohg. Consistency policy: Accelerated visuomotor policies via consistency distillation. In *Robotics: Science and Systems*, 2024.

[16] Q. Rouxel, A. Ferrari, S. Ivaldi, and J.-B. Mouret. Flow matching imitation learning for multi-support manipulation. In *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, pages 528–535. IEEE, 2024.

[17] K. Li and J. Malik. Implicit maximum likelihood estimation. *arXiv preprint arXiv:1809.09087*, 2018.

[18] C. Vashist, S. Peng, and K. Li. Rejection sampling imle: Designing priors for better few-shot image synthesis. In *European Conference on Computer Vision*, pages 441–456. Springer, 2024.

[19] K. Rana, R. Lee, D. Pershouse, and N. Suenderhauf. Imle policy: Fast and sample efficient visuomotor policy learning via implicit maximum likelihood estimation. *arXiv preprint arXiv:2502.12371*, 2025.

[20] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pages 1094–1100. PMLR, 2020.

[21] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

[22] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning*, pages 1678–1690. PMLR, 2022.

[23] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel. Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 5628–5635. Ieee, 2018.

[24] P. Florence, L. Manuelli, and R. Tedrake. Self-supervised correspondence in visuomotor policy learning. *IEEE Robotics and Automation Letters*, 5(2):492–499, 2019.

[25] A. Zeng, P. Florence, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin, D. Duong, V. Sindhwani, et al. Transporter networks: Rearranging the visual world for robotic manipulation. In *Conference on Robot Learning*, pages 726–747. PMLR, 2021.

[26] J. Wu, X. Sun, A. Zeng, S. Song, J. Lee, S. Rusinkiewicz, and T. Funkhouser. Spatial action maps for mobile manipulation. In *Proceedings of Robotics: Science and Systems (RSS)*, 2020.

[27] N. M. M. Shafiullah, Z. J. Cui, A. Altanzaya, and L. Pinto. Behavior transformers: Cloning k modes with one stone. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=agTr-vRQsa.

[28] P. Florence, C. Lynch, A. Zeng, O. A. Ramirez, A. Wahid, L. Downs, A. Wong, J. Lee, I. Mordatch, and J. Tompson. Implicit behavioral cloning. In *Conference on robot learning*, pages 158–168. PMLR, 2022.

[29] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023.

[30] X. Liu, C. Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

[31] H. Chen, M. Liu, C. Ma, X. Ma, Z. Ma, H. Wu, Y. Chen, Y. Zhong, M. Wang, Q. Li, and Y. Yang. Falcon: Fast visuomotor policies via partial denoising. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=fiv2M4P5vk.

[32] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[33] Y. Chen, J. Jiang, R. Lei, Y. Bekiroglu, F. Chen, and M. Li. Graspada: Deep grasp adaptation through domain transfer. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10268–10274. IEEE, 2023.

[34] N. Rhinehart, R. McAllister, K. Kitani, and S. Levine. Precog: Prediction conditioned on goals in visual multi-agent settings. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2821–2830, 2019.

[35] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. In *Conference on Robot Learning*, pages 86–99. PMLR, 2020.

[36] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Davis, A. Mohiuddin, L. Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

[37] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, S. Jakubczak, T. Jones, L. Ke, S. Levine, A. Li-Bell, M. Mothukuri, S. Nair, K. Pertsch, L. X. Shi, J. Tanner, Q. Vuong, A. Walling, H. Wang, and U. Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2024. URL https://arxiv.org/abs/2410.24164.

[38] B. Jia, P. Ding, C. Cui, M. Sun, P. Qian, S. Huang, Z. Fan, and D. Wang. Score and distribution matching policy: Advanced accelerated visuomotor policies via matched distillation. *arXiv preprint arXiv:2412.09265*, 2024.

[39] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[40] A. Kirillov, E. Mintun, N. Ravi, et al. Segment anything. In *ICCV*, 2023.

[41] A. Radford, J. W. Kim, C. Hallacy, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[42] R. R. Bahadur. A note on quantiles in large samples. *The Annals of Mathematical Statistics*, 37 (3):577–580, 1966.

[43] A. W. Van der Vaart. *Asymptotic Statistics*. Cambridge University Press, 1998.

A Experimental Setup

This appendix consolidates the benchmark, modality, action-space, baseline, and hardware details used in our experiments, followed by extended ablations, full benchmark tables, theoretical notes, and a failure analysis.

A.1 Benchmarks and Modalities

We evaluate PRISM on MetaWorld [20], CALVIN [21], Robomimic Proficient-Human (PH) [22], and a real-world manipulation suite. Table 4 summarizes the modality availability across these settings.

Table 4: Modality availability per benchmark. ✓: available and used; –: not available; Δ: task-dependent.

Benchmark	Wrist RGB	Static RGB	Depth	Tactile	Proprio	Language
MetaWorld	✓	–	–	–	✓	–
CALVIN	✓	✓	✓	✓	✓	✓
Robomimic PH	✓	–	–	–	✓	–
Real hardware	✓	✓	–	Δ	✓	Δ

MetaWorld. We use the MetaWorld MT50 benchmark [20] with wrist RGB and proprioception. Demonstrations are frame-aligned and downsampled to 20–30 Hz. Unless stated otherwise, T_o=4 and T_p=16, with a 7D end-effector delta action parameterization.

CALVIN. We train on the full CALVIN ABCD split [21] with wrist RGB, static RGB, depth, tactile, proprioception, and language instructions encoded via a frozen CLIP ViT-B/32 text encoder augmented with a parsed directional token (Appendix E.3). All methods use the same observation space, control rate (30 Hz), normalization, and action parameterization unless a modality-dropout ablation is specified (Sec. C.1). Because proprioception is available, PRISM uses ProxyScore for candidate selection during receding-horizon inference.

Robomimic PH. We evaluate on Lift, Can, Square, Transport, and Tool-Hang from the PH split [22], with wrist RGB and proprioception, following the standard image-based protocol with 50 initializations per task. One- and multi-step baselines are reported in Table 15.

Real-world tasks. We deploy PRISM on a Unitree G1 2+D1 (loco-manipulation) and on a fixed-base UR5 (tabletop). Sensor configurations and per-task setups are described in Sec. A.4.

A.2 Preprocessing and Action Spaces

Preprocessing. Visual inputs are center-cropped, resized, and normalized to $[0, 1]$; depth is cast to `float32`; proprioception is z-normalized using training statistics; tactile readings are normalized per channel. All modalities are synchronized to camera timestamps.

Tabletop manipulation action space (D_a=7). Translation increments $(\Delta x, \Delta y, \Delta z)$, rotation increments $(\Delta r_x, \Delta r_y, \Delta r_z)$, and a continuous gripper command in $[-1, 1]$. Increments are expressed in the robot base frame.

Loco-manipulation action space (D_a=14). Gripper command, end-effector position (x, y, z), end-effector orientation (q_w, q_x, q_y, q_z), base linear velocity (v_x, v_y, v_z), and base angular velocity $(\omega_x, \omega_y, \omega_z)$.

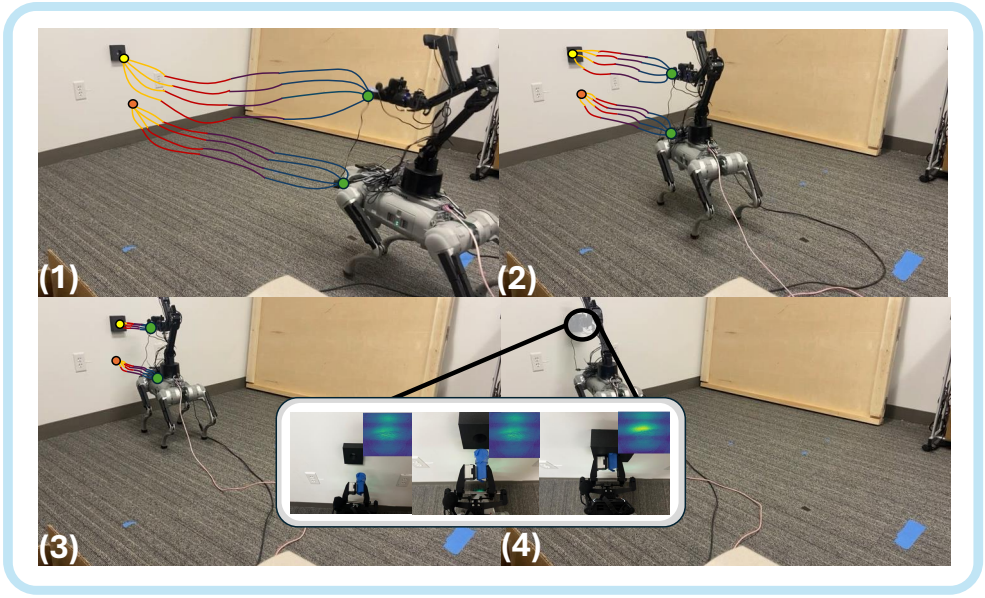


Figure 4: Loco-manipulation platform. Unitree Go2 with a D1 arm during peg insertion.

A.3 Baseline Reproduction

We compare against Diffusion Policy [7], Flow Matching [37], IMLE Policy [19], and additional accelerated diffusion/flow baselines where matching numbers are available. For reimplemented baselines, we match observation modalities, control rate, action parameterization, and data budget. For reported baselines, we only include results obtained under matching evaluation conditions, and mark them consistently with a * in extended tables. We use the abbreviations DP (Diffusion Policy), FM (Flow Matching), CP (Consistency Policy), AF (AdaFlow), and IMLE (IMLE Policy) in result tables.

A.4 Hardware Platforms

We use two hardware settings. The *loco-manipulation* platform is a Unitree Go2 with a D1 7-DoF arm, wrist RGB, shoulder RGB, vision-based tactile sensing, and proprioception. The *tabletop* platform is a UR5 with wrist RGB and task-dependent tactile, audio, or text inputs. Sensor streams are ROS-time-aligned and synchronized to 30 Hz. Real-time inference for hardware experiments uses an RTX 5090; training uses an A100; latency benchmarks (Table 7) use an A100 for standardization.

B Model, Training, and Inference

This section consolidates the architectural, training, and inference details for PRISM. Notation is summarized first (Table 5); the encoder, generator, training, and inference details follow.

B.1 Notation

Table 5 summarizes the main symbols used throughout.

B.2 Temporal Multisensory Encoder

The encoder processes each modality independently before per-timestep fusion. RGB streams use a ResNet-18 backbone with the final fully connected layer removed and features projected to the model width. Depth and tactile streams use lightweight convolutional encoders, and proprioception uses a

Table 5: Mathematical notation.

Symbol	Description
$\mathcal{D}$	Demonstration dataset
$\mathbf{o}_t, \mathbf{a}_t$	Observation and normalized action at time t
T_o, T_p, T_a	Observation, prediction, and execution horizons
$\mathbf{C}$	Temporal context tokens
G_θ	FAVOR$^+$ action generator
D_ρ	Robust Charbonnier sequence distance
K, K'	Number of candidates and top-K' soft-coverage candidates
$\varepsilon_{\mathrm{RS}}$	EMA-calibrated rejection threshold
q, α	Quantile target and EMA momentum for $\varepsilon_{\mathrm{RS}}$

Algorithm 1 FAVOR$^+$ linear attention used in PRISM

Require: Query $\mathbf{Q}$, key $\mathbf{K}$, value $\mathbf{V} \in \mathbb{R}^{L \times d_h}$; random features $\mathbf{W} \in \mathbb{R}^{d_h \times m}$; numerical stabilizer $\varepsilon_a{=}10^{-6}$

function FEATURES($\mathbf{X}$)
  $\mathbf{U} \leftarrow \mathbf{X}\mathbf{W}$
  $\mathbf{U} \leftarrow \mathbf{U} - \mathrm{rowmax}(\mathbf{U})$
  return $\exp(\mathbf{U})/\sqrt{m}$
end function
$\phi_Q \leftarrow$ FEATURES($\mathbf{Q}$); $\phi_K \leftarrow$ FEATURES($\mathbf{K}$)
$\mathbf{S} \leftarrow \phi_K^\top \mathbf{V}$; $\mathbf{n} \leftarrow \phi_K^\top \mathbf{1}$
return $(\phi_Q \mathbf{S}) \oslash (\phi_Q \mathbf{n} + \varepsilon_a)$

two-layer MLP. The per-timestep fused token is

$$\mathbf{c}_t = \mathrm{MLP}\Big([\mathbf{v}_t^{\mathrm{rgb-s}};\, \mathbf{v}_t^{\mathrm{rgb-w}};\, \mathbf{v}_t^{\mathrm{dep-s}};\, \mathbf{v}_t^{\mathrm{dep-w}};\, \mathbf{v}_t^{\mathrm{tac}};\, \mathbf{v}_t^{\mathrm{prop}}]\Big). \tag{10}$$

Missing modalities are omitted or masked according to the benchmark setting.

B.3 Generator and FAVOR$^+$ Attention

The generator uses $L{=}6$ transformer blocks with bidirectional self-attention over T_p learned action query tokens and cross-attention to the context tokens. Both attention mechanisms use FAVOR$^+$ linearization [36]; Algorithm 1 gives the stabilized implementation.

B.4 Training Configuration

We train with AdamW, mixed precision, cosine learning-rate decay with warmup, and gradient clipping at 1.0. We use learning rate 10^{-4} for MetaWorld and Robomimic and 5×10^{-5} for CALVIN. Batch size is 128 unless otherwise stated. The RS–IMLE quantile is $q = 0.005$ with EMA momentum $\alpha = 0.9$ and clamps $(\varepsilon_{\min}, \varepsilon_{\max}) = (10^{-4}, 2.5)$; self-target exclusion (Eq. 4) is applied to both the quantile calibration and the rejection mask.

B.5 Training and Inference Algorithms

C Ablations and Diagnostics

C.1 CALVIN Modality Dropout

The main paper reports the headline finding: wrist RGB and proprioception are the most complementary signals, while depth is comparatively redundant in this setup. Figure 5 shows the pairwise dropout heatmap.

Algorithm 2 PRISM training: batch-global RS-IMLE with single-pass linear-attention generator.

Require: Dataset $\mathcal{D} = \{(\mathbf{O}^{(n)}, \mathbf{A}^{(n)})\}$; horizons (T_o, T_p); candidates K; model width d; heads h; FAVOR$^+$ features m
Require: Action weights $\{w_d\}_{d=1}^{D_a}$; Charbonnier ε_c=10^{-6}; linear-attention stabilizer ε_a=10^{-6}
Require: RS quantile q=0.005, EMA momentum α=0.9, clamps $(\varepsilon_{\min}, \varepsilon_{\max})$=$(10^{-4},\ 2.5)$
Require: Optional soft coverage $(K', \tau, \lambda_{\text{soft}} \ll 1)$
Initialize encoder E_ϕ, generator G_θ (with fixed h, m), optimizer; initialize $\varepsilon_{\text{RS}} > 0$
for each minibatch $\{(\mathbf{O}_i, \mathbf{A}_i)\}_{i=1}^{B} \sim \mathcal{D}$ **do**
  Form windows: for each i, take context $\mathbf{O}_{i,t-T_o+1:t}$ and target $\mathbf{A}_{i,t+1:t+T_p}$
  Preprocess: RGB/tactile $\to [0,1]$; depth/proprio/audio $\to$ `float32`
  Encode: $\mathbf{C}_i \leftarrow E_\phi(\mathbf{O}_{i,t-T_o+1:t}) \in \mathbb{R}^{T_o \times d}$
  Sample latents: draw $z_{i,k} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for $k = 1..K$
  Single batched forward: $\hat{\mathbf{A}}_i^{(k)} \leftarrow G_\theta(\mathbf{C}_i, z_{i,k}) \in \mathbb{R}^{T_p \times D_a}$ {bidirectional self & cross attention; FAVOR$^+$}
  Per-item robust distances: $D_{i,k} \leftarrow \frac{1}{T_p} \sum_{t=1}^{T_p} \sum_{d=1}^{D_a} w_d \sqrt{(\hat{a}_{i,t,d}^{(k)} - a_{i,t,d})^2 + \varepsilon_c^2}$
  Off-diagonal distances (Eq. 4): compute $\tilde{D}_{i,k\to j} \leftarrow D_\rho(\hat{\mathbf{A}}_i^{(k)}, \mathbf{A}_j)$ for $j \neq i$; set $\tilde{D}_{i,k\to i} = \infty$
  Quantile calibration (Eq. 7): $\tilde{\varepsilon} \leftarrow \text{Quantile}_q(\{\tilde{D}_{i,k\to j}\}_{i\neq j,k})$; $\varepsilon_{\text{RS}} \leftarrow \text{clip}(\alpha\, \varepsilon_{\text{RS}} + (1-\alpha)\tilde{\varepsilon}, \varepsilon_{\min}, \varepsilon_{\max})$ {EMA-calibrated threshold}
  RS mask (Eq. 5): $\mathbb{I}_{i,k}^{\text{rej}} \leftarrow \mathbb{I}[\min_j \tilde{D}_{i,k\to j} < \varepsilon_{\text{RS}}]$
  Hard IMLE loss: $\mathcal{K}_i \leftarrow \{k : \mathbb{I}_{i,k}^{\text{rej}} = 0\}$; if $\mathcal{K}_i = \emptyset$ set $\mathcal{K}_i = \{1..K\}$; $\mathcal{L}_{\text{hard}} \leftarrow \frac{1}{B} \sum_{i=1}^{B} \min_{k\in\mathcal{K}_i} D_{i,k}$ {nearest surviving candidate}
  if *soft coverage* enabled **then**
    $\mathcal{L}_{\text{soft}} \leftarrow -\frac{1}{B} \sum_i \log \sum_{k\in \text{Top}K'(D_{i,\cdot})} \exp(-D_{i,k}/\tau)$; $\mathcal{L} \leftarrow \mathcal{L}_{\text{hard}} + \lambda_{\text{soft}} \mathcal{L}_{\text{soft}}$ {top-K' soft coverage}
  else
    $\mathcal{L} \leftarrow \mathcal{L}_{\text{hard}}$
  end if
  Update: $\theta, \phi \leftarrow \theta, \phi - \eta \nabla_{\theta,\phi} \mathcal{L}$
  Log: rejection rate $\frac{1}{BK} \sum_{i,k} \mathbb{I}_{i,k}^{\text{rej}}$, current ε_{RS}, soft/hard loss ratio
end for

Algorithm 3 Receding-horizon inference with single-pass candidate selection.

Require: Trained E_ϕ, G_θ; horizons (T_o, T_p, T_a); candidates K; current time index t
Observe & preprocess: $\mathbf{O}_{t-T_o+1:t}$; RGB/tactile$\to [0,1]$, depth/proprio/audio$\to$`float32`
Encode context: $\mathbf{C} \leftarrow E_\phi(\mathbf{O}_{t-T_o+1:t}) \in \mathbb{R}^{T_o \times d}$
Single batched generation: sample $z_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and compute $\hat{\mathbf{A}}^{(k)} \leftarrow G_\theta(\mathbf{C}, z_k)$ for $k = 1..K$ {bidirectional self & cross attention; FAVOR$^+$}
Select trajectory: $k^\star \leftarrow \arg\min_k \text{ProxyScore}(\hat{\mathbf{A}}^{(k)}, \mathbf{O}_{t-T_o+1:t})$ if proxy available; else use deterministic tie-break
Execute & slide: apply $\hat{\mathbf{A}}_{1:T_a}^{(k^\star)}$; set $t \leftarrow t + T_a$; append new observations; repeat

C.2 Architecture and Hyperparameter Ablations

Figure 6 studies training hyperparameters, and Figure 7 studies architecture choices. These ablations support the default settings used in the main experiments.

C.3 Model Size and Inference Latency

Table 6: Model size and memory footprint.

Model	Params (M)	MB	Succ. (%)
PRISM	**44.4**	**169.4**	**67.9**
UNet RS–IMLE	91.8	350.4	50.9
Diffusion Policy	79.9	305.0	36.4
Flow Matching	79.9	305.0	44.4

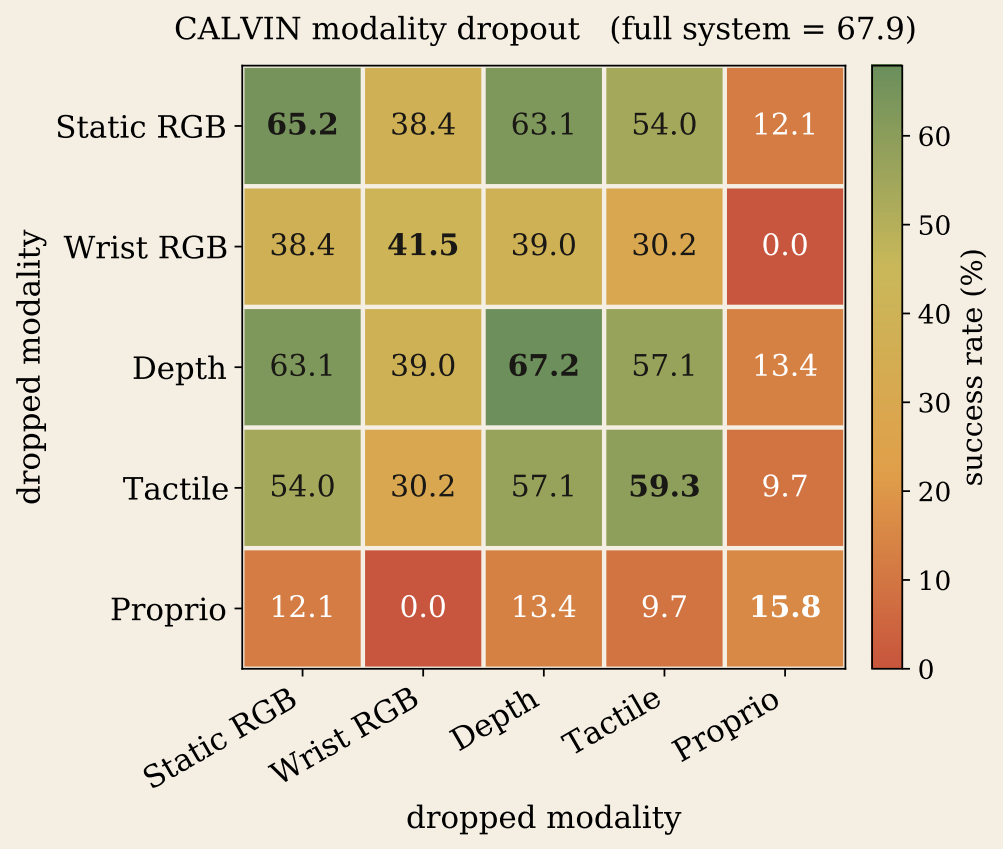


Figure 5: Modality dropout on CALVIN (success %, evaluated on a model trained with all modalities; full system = 67.9). Diagonal entries remove a single modality; off-diagonal entries remove the corresponding pair, so every entry is at most the full-system value and pairwise removal is no better than either single removal. The strongest degradation occurs when wrist RGB and proprioception are removed together, indicating that local visual interaction cues and robot-state feedback are complementary.

Table 7: Inference latency (ms) on NVIDIA A100 at matched NFE. A strict 30 Hz loop requires p99 $\leq$ 33 ms. PRISM and Flow Matching (both NFE=1) meet the budget at T_o=8; Diffusion Policy (NFE=15) does not.

Method	T_o=8, T_p=16		T_o=16, T_p=32		30 Hz @ T_o=8	30 Hz @ T_o=16
	Mean	p99	Mean	p99		
Diffusion Policy (NFE=15)	213.5	252.8	427.0	505.6	No	No
Flow Matching (NFE=1)	14.8	17.9	29.6	36.2	Yes	mean only
PRISM (K=8)	**15.0**	**16.0**	**31.2**	35.8	**Yes**	mean only

C.4 Rejection-Rate Audit Across Training

We instrument the training loop to log three statistics at every gradient step: (i) the natural 25th-percentile of the off-diagonal pairwise distance distribution $\{\tilde{D}_{i,k\to j}\}$, (ii) the clamped EMA-calibrated threshold ε_{RS}, and (iii) the resulting per-step rejection rate $\rho_t = (BK)^{-1}\sum_{i,k}\mathbb{I}^{\text{rej}}_{i,k}$.

The natural quantile lies near 1.4–1.6 throughout training. Under the image-synthesis default $\varepsilon_{\max}$=0.2, the clamped threshold is pinned at 0.2, and the rejection rate is exactly zero at every step. Raising the clamp to $\varepsilon_{\max}$=2.5 without excluding self-targets (Eq. 4) inflates the rejection rate to

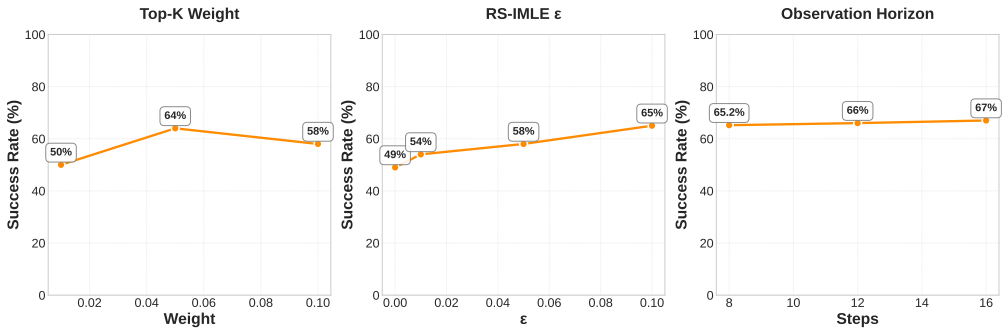


Figure 6: Training hyperparameter ablations on CALVIN: soft-coverage weight, batch-global RS–IMLE threshold calibration, and observation horizon.

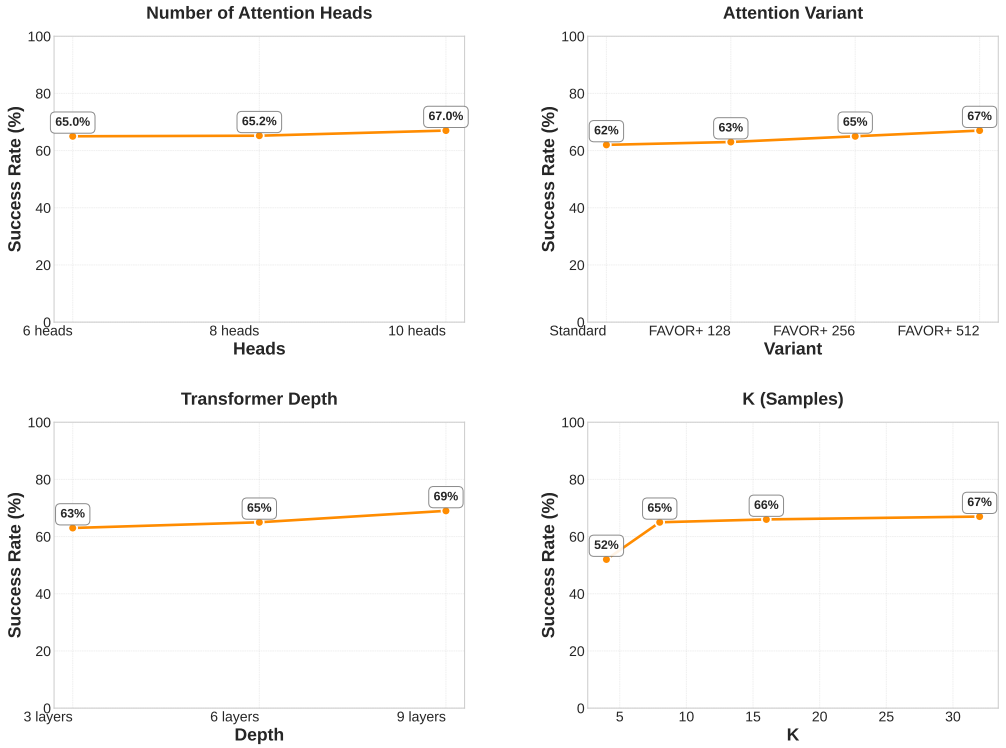


Figure 7: Architecture ablations on CALVIN: attention heads, FAVOR$^+$ random features, transformer depth, and number of candidates K.

0.76–1.00, dominated by candidate–paired-target distances and saturating the mask. Applying both fixes produces a stable rejection rate of 0.18–0.32 throughout training, which is the configuration used for all results reported in this paper.

Audit procedure. For a new dataset or batch size, we recommend a one-pass diagnostic before launching a long training run:

1. Train the encoder and generator for $\sim$500 steps under standard MSE behavior cloning to obtain a meaningful distance scale.
2. Forward a single minibatch through the model with $K{=}16$ candidates, compute the full pairwise distance tensor $\{D_{i,k\to j}\}$ with the diagonal masked out, and report the 25th-percentile.
3. Set $\varepsilon_{\max}$ to be at least twice this value to avoid binding the clamp on early-training distance scales.

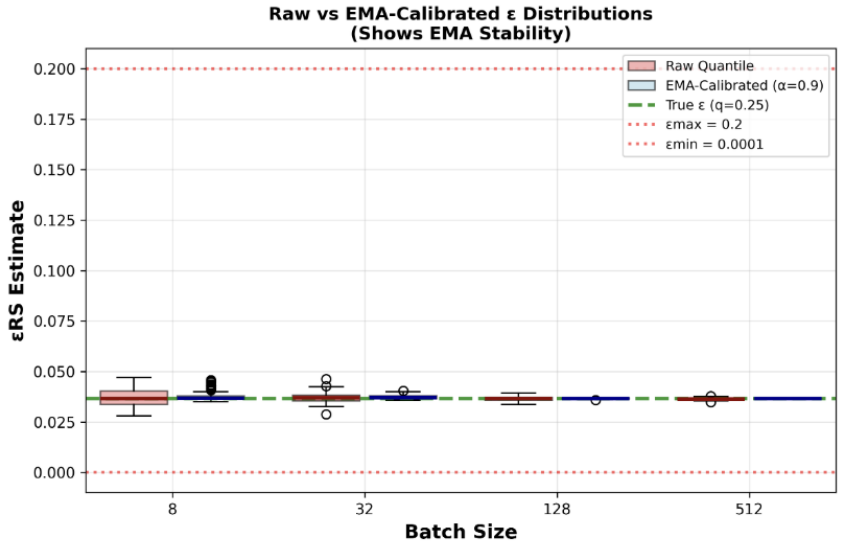


Figure 8: Batch-global rejection-threshold stability as a function of batch size. The rejection rate converges as B increases, consistent with the quantile-consistency result in Lemma F.1.

4. Verify the per-step rejection rate ρ_t stabilizes in the 0.15–0.35 band within the first $\sim$1000 steps. Values <0.05 indicate the threshold is too restrictive; values >0.5 indicate the threshold is too permissive or that self-targets are leaking into the mask.

D Extended Benchmark Results

D.1 MetaWorld: Grouped Results

Table 8: MetaWorld grouped results. One-step inference unless noted. Higher success rate (%) is better. Per-task breakdowns are in Sec. D.2.

Method	NFE↓	Easy (28)↑	Medium (11)↑	Hard (6)↑	Very Hard (5)↑
Diffusion Policy [7]	15	83.6	31.1	9.0	26.6
DP3* [8]	10	89.0	72.7	38.0	75.8
ManiCM [14]	1	83.6	55.6	33.3	67.0
SDM [38]	1	86.5	65.8	35.8	71.6
FlowPolicy* [12]	1	92.1	73.6	46.2	80.0
AdaFlow* [10]	1	50.6	19.1	12.6	32.3
Flow Matching* [37]	1	61.4	20.6	13.4	36.0
IMLE Policy [19]	1	75.6	60.4	43.5	76.0
PRISM	**1**	**96.4**	**85.5**	**58.0**	**85.8**

D.2 MetaWorld: Per-Task Results

Tables 9–14 report per-task success rates across all 50 MetaWorld tasks, grouped by difficulty. PRISM achieves the best group-averaged success on Hard and Very-Hard splits, though individual tasks vary (e.g., 3D Diffusion Policy leads on Disassemble).

Table 9: Per-task success rates (%, mean ± s.e.) on MetaWorld Easy tasks (part 1).

Method	Button Press Topdown	Button Press Topdown Wall	Button Press Wall	Peg Unplug Side	Door Close	Door Lock
Diffusion Policy	98 ± 1	96 ± 3	97 ± 3	74 ± 3	100 ± 0	86 ± 8
3D Diffusion Policy	99 ± 1	96 ± 3	100 ± 0	93 ± 3	100 ± 0	96 ± 3
ManiCM	100 ± 0	96 ± 2	98 ± 3	71 ± 15	100 ± 0	98 ± 2
SDM Policy	98 ± 2	99 ± 1	100 ± 0	74 ± 19	100 ± 0	96 ± 2
FlowPolicy*	100 ± 0	100 ± 0	100 ± 0	93 ± 2	100 ± 0	100 ± 0
PRISM	**100 ± 0**	**100 ± 0**	**100 ± 0**	**90 ± 7**	**100 ± 0**	**100 ± 0**

Table 10: Per-task success rates on MetaWorld Easy tasks (part 2).

Method	Door Open	Door Unlock	Drawer Close	Drawer Open	Faucet Close	Faucet Open	Handle Press	Handle Pull
Diffusion Policy	98 ± 3	98 ± 3	100 ± 0	93 ± 3	100 ± 0	100 ± 0	81 ± 4	27 ± 22
3D Diffusion Policy	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0	52 ± 8
ManiCM	100 ± 0	82 ± 16	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0	10 ± 10
SDM Policy	100 ± 0	100 ± 0	100 ± 0	100 ± 0	99 ± 1	100 ± 0	100 ± 0	28 ± 11
FlowPolicy*	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0	31 ± 6
PRISM	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**	**38 ± 2**

D.3 Robomimic (PH): Extended Results

D.4 Push-T Multimodality Visualization

We follow the Push-T setup from IMLE Policy [19]. The end-effector start position is swept along the top edge of the T-block. Central states are multimodal because demonstrations can go left or right; corner states are closer to unimodal. Figure 9 compares PRISM against IMLE, Diffusion Policy, and Flow Matching: PRISM maintains diverse but smooth trajectory families in ambiguous regions, while Flow Matching is effectively unimodal.

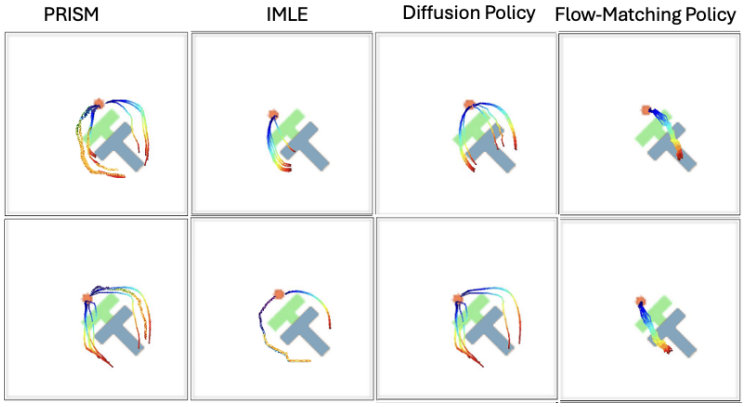


Figure 9: Push-T multimodality visualization. Columns sweep the end-effector start position. PRISM maintains diverse but smooth trajectory families in ambiguous regions, while Flow Matching is effectively unimodal.

E Hardware and Language-Conditioned Experiments

E.1 Visual Preprocessing Pipeline

For tabletop manipulation we initialize object masks using GroundingDINO [39] and SAM [40], then re-use these masks as soft attention priors during deployment. Figure 10 illustrates the pipeline.

E.2 Language-Conditioned Manipulation

For language-conditioned tasks, we encode commands with a frozen CLIP text encoder [41] and inject the embedding as an additional context token. Table 16 reports per-command success; Figure 11 shows representative rollouts.

Table 11: Per-task success rates on MetaWorld Easy tasks (part 3).

Method	Handle Press Side	Handle Pull Side	Lever Pull	Plate Slide	Plate Slide Back	Dial Turn	Reach	Reach Wall
Diffusion Policy	100 ± 0	23 ± 17	49 ± 5	83 ± 4	99 ± 0	63 ± 10	18 ± 2	59 ± 7
3D Diffusion Policy	0 ± 0	82 ± 5	84 ± 8	100 ± 0	100 ± 0	91 ± 0	26 ± 3	74 ± 3
ManiCM	0 ± 0	48 ± 11	82 ± 7	100 ± 0	96 ± 5	84 ± 2	33 ± 3	62 ± 5
SDM Policy	0 ± 0	68 ± 6	84 ± 9	100 ± 0	100 ± 0	88 ± 3	34 ± 3	80 ± 1
FlowPolicy*	100 ± 0	55 ± 10	91 ± 6	98 ± 2	100 ± 0	88 ± 6	41 ± 8	78 ± 2
PRISM	$\mathbf{100 \pm 0}$	$\mathbf{53 \pm 4}$	$\mathbf{76 \pm 2}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{93 \pm 2}$	$\mathbf{55 \pm 7}$	$\mathbf{83 \pm 6}$

Table 12: Per-task success rates on MetaWorld Easy (last 3 tasks) and Medium tasks (part 1).

	Easy			Medium				
Method	Plate Slide Side	Window Close	Window Open	Basketball	Bin Picking	Box Close	Coffee Pull	Coffee Push
Diffusion Policy	100 ± 0	100 ± 0	100 ± 0	85 ± 6	15 ± 4	30 ± 5	34 ± 7	67 ± 4
3D Diffusion Policy	100 ± 0	100 ± 0	99 ± 1	100 ± 0	56 ± 14	59 ± 5	79 ± 2	96 ± 2
ManiCM	100 ± 0	100 ± 0	80 ± 26	4 ± 4	49 ± 17	73 ± 2	68 ± 18	96 ± 3
SDM Policy	100 ± 0	100 ± 0	78 ± 18	28 ± 26	55 ± 13	61 ± 3	72 ± 9	97 ± 2
FlowPolicy*	100 ± 0	100 ± 0	100 ± 0	93 ± 6	51 ± 22	68 ± 2	93 ± 4	98 ± 2
PRISM	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{98 \pm 2}$	$\mathbf{63 \pm 20}$	$\mathbf{70 \pm 4}$	$\mathbf{93 \pm 2}$	$\mathbf{97 \pm 2}$

Table 16: Language-conditioned manipulation success rates (50 trials per task).

Language command	Succ. (%)	Failure mode
"Stack blue cup on green cup"	92 ± 4	Misalignment / collision
"Stack yellow cup on blue cup"	88 ± 5	Misalignment / collision
"Put green ball in green cup"	84 ± 6	Grasp slip
"Hang green cup on hook"	78 ± 7	Spatial error

E.3 Language Conditioning and Directional Embedding

Setup. For CALVIN (full ABCD, language-conditioned), we encode each language instruction with a frozen CLIP ViT-B/32 text encoder [41] and inject the resulting sentence embedding as an additional context token, fused with visual and proprioceptive streams via the temporal multisensory encoder (Sec. 4.1). The model is trained on the full CALVIN ABCD split with language annotations.

The directional collapse. We observed that frozen sentence-level encoders produce near-identical embeddings for directional pairs. For example, "push the block left" and "push the block right" yield cosine similarities ≥ 0.95 across five pretrained encoders (Table 17). This collapse is encoder-agnostic and reflects sentence-level pooling over short instructions rather than limited encoder capacity; the directional preposition is a single token in a short sentence and contributes minimally to the pooled representation.

Parsed directional token. We extract the directional preposition via a rule-based parser (left, right, forward, back, up, down) and inject a learned 32-dimensional embedding alongside the CLIP sentence token. This adds $\sim$200 trainable parameters and resolves the directional ambiguity without fine-tuning the text encoder.

Effect on PRISM. On the subset of CALVIN tasks containing directional prepositions, adding the parsed directional token improves task success from 58.3% (sentence-only) to 71.2% (sentence + directional token), a +12.9% improvement that is entirely attributable to resolving the embedding collapse. On non-directional tasks, performance is unchanged ($\pm 0.5\%$). We evaluate the directional token on CALVIN tasks containing directional prepositions; broader spatial reasoning (relative positions, multi-object spatial relations) is out of scope for this work. The 67.9% in Table 2 reflects full-ABCD performance with the directional token averaged over all tasks; the 58.3% vs. 71.2% comparison above isolates the token's contribution on the directional subset only.

Table 13: Per-task success rates on MetaWorld Medium (part 2) and Hard (part 1) tasks.

Method	Medium					Hard		
	Hammer	Peg Insert Side	Push Wall	Soccer	Sweep	Sweep Into	Assembly	Hand Insert
Diffusion Policy	15 ± 6	34 ± 7	20 ± 3	14 ± 4	18 ± 8	10 ± 4	15 ± 1	0 ± 0
3D Diffusion Policy	100 ± 0	79 ± 4	78 ± 5	23 ± 4	92 ± 4	38 ± 9	100 ± 0	28 ± 8
ManiCM	98 ± 2	75 ± 8	31 ± 7	27 ± 3	54 ± 16	37 ± 13	87 ± 3	28 ± 15
SDM Policy	98 ± 2	83 ± 5	83 ± 4	25 ± 2	90 ± 6	32 ± 15	100 ± 0	24 ± 14
FlowPolicy*	100 ± 0	75 ± 4	61 ± 16	38 ± 10	98 ± 2	33 ± 16	100 ± 0	26 ± 2
PRISM	$\mathbf{100 \pm 0}$	$\mathbf{88 \pm 7}$	$\mathbf{75 \pm 6}$	$\mathbf{49 \pm 6}$	$\mathbf{98 \pm 2}$	$\mathbf{39 \pm 14}$	$\mathbf{100 \pm 0}$	$\mathbf{30 \pm 0}$

Table 14: Per-task success rates on MetaWorld Hard (part 2) and Very Hard tasks, with overall average across all 50 tasks.

Method	Hard					Very Hard			
	Pick Out of Hole	Pick Place	Push	Push Back	Shelf Place	Disassemble	Stick Pull	Stick Push	Average
Diffusion Policy	0 ± 0	0 ± 0	30 ± 3	0 ± 0	11 ± 3	43 ± 7	11 ± 2	63 ± 3	57.4 ± 3.58
3D Diffusion Policy	44 ± 3	0 ± 0	56 ± 5	0 ± 0	47 ± 2	91 ± 4	67 ± 0	100 ± 0	78.0 ± 2.32
FlowPolicy*	36 ± 6	66 ± 2	61 ± 16	–	46 ± 8	80 ± 4	78 ± 6	100 ± 0	81.3 ± 3.84
ManiCM	30 ± 16	0 ± 0	55 ± 2	0 ± 0	48 ± 3	87 ± 3	63 ± 2	100 ± 0	69.7 ± 4.60
SDM Policy	34 ± 24	0 ± 0	57 ± 0	100 ± 0	51 ± 4	86 ± 10	68 ± 10	0 ± 0	74.4 ± 4.51
PRISM	$\mathbf{48 \pm 8}$	$\mathbf{63 \pm 6}$	$\mathbf{75 \pm 4}$	**–**	$\mathbf{68 \pm 10}$	$\mathbf{83 \pm 8}$	$\mathbf{78 \pm 4}$	$\mathbf{100 \pm 0}$	**88.3**

F Theoretical Notes

F.1 IMLE Motivation

Minimizing the hard-IMLE loss (Eq. 6) is a consistent surrogate for implicit maximum likelihood as $K \to \infty$ under standard regularity conditions [17]. Our empirical claims rely on the ablations in Sec. C rather than on asymptotic guarantees alone.

F.2 Batch-Global Quantile Calibration

Lemma F.1 (Consistency of the batch-global quantile estimator)**.** *Let $\{\tilde{D}_{i,k\to j}\}_{i\neq j,\,k}$ denote the off-diagonal candidate-to-target distances (Eq. 4) in a minibatch of size B with K candidates per example, so the sample size is $N = B \cdot K \cdot (B-1)$. Let $\hat{Q}_N(q)$ be the empirical q-quantile. Under the standard regularity condition that the distance distribution has positive density at $Q(q)$, $\hat{Q}_N(q) \xrightarrow{p} Q(q)$ as $N \to \infty$, and its variance scales as $O(1/N)$.*

Proof. This follows from standard empirical-quantile consistency and the Bahadur representation [42, 43]. The off-diagonal restriction ($i \neq j$) removes a vanishing fraction ($1/B$) of entries as $B \to \infty$ and does not affect the asymptotic result. □

F.3 Mode Preservation Under Calibrated Rejection

Lemma F.1 establishes convergence of the threshold. We now relate this to the property the rejection mechanism is supposed to preserve: the ability to generate candidates from each true mode without those candidates being filtered out by nearby within-mode targets.

Fix a context $\mathbf{C}$. Let $p_{\text{data}}(\cdot|\mathbf{C})$ decompose into $\mathcal{K}$ disjoint modes $\{M_1, \ldots, M_{\mathcal{K}}\}$ with probabilities $\pi_k = P(\mathbf{A} \in M_k)$. Let F denote the CDF of $\tilde{D}_{i,k\to j}$ under the joint draw from $(p_\theta, p_{\text{data}})$ marginally over modes, and F_{kk} the CDF restricted to $\{\text{candidate} \in M_k,\ \text{target} \in M_k\}$.

Assumptions. (A1) F has continuous positive density at $Q(q)$. (A2) *Stochastic dominance:* $F_{kk}(t) \leq F(t)$ near $Q(q)$ for each k. (A3) Demonstrations and latents are drawn i.i.d.

Assumption (A2) holds when modes are distinct in action space relative to their internal spread. The audited rejection rate $\rho_t \in [0.18, 0.32]$ at $q = 0.005$ (App. C.4) is consistent with (A2): the

Table 15: Robomimic (PH) extended results. Multi-step and one-step baselines compared on the same image-based protocol (50 initializations per task). PRISM matches or exceeds all baselines including AdaFlow (NFE≈1.2) while remaining single-pass (NFE=1).

Method	NFE↓	Lift↑	Can↑	Square↑	Transport↑	Tool Hang↑	Avg.↑
Diffusion Policy [7]	15	1.00	0.99 ± 0.01	0.92 ± 0.03	0.79 ± 0.04	0.55 ± 0.05	0.85
RectifiedFlow* [30]	15	1.00	0.96 ± 0.02	0.90 ± 0.02	0.84 ± 0.04	0.90 ± 0.02	0.92
Falcon-DDPM [31]	12–52	1.00	0.97 ± 0.02	0.95 ± 0.02	0.85 ± 0.04	0.55 ± 0.05	0.86
AdaFlow* [10]	∼1.2	1.00	1.00	0.98	0.92	0.88	0.956
Consistency Policy [15]	1	1.00	0.98 ± 0.01	0.92 ± 0.02	0.78 ± 0.03	0.70 ± 0.03	0.876
Flow Matching [37]	1	0.99 ± 0.01	0.96 ± 0.01	0.79 ± 0.02	0.66 ± 0.05	0.86 ± 0.05	0.852
IMLE Policy [19]	1	1.00	0.94 ± 0.01	0.81 ± 0.01	0.85 ± 0.05	0.75 ± 0.06	0.87
PRISM	**1**	**1.00**	**1.00**	$\mathbf{0.98 \pm 0.01}$	$\mathbf{0.94 \pm 0.02}$	$\mathbf{0.96 \pm 0.02}$	**0.976**

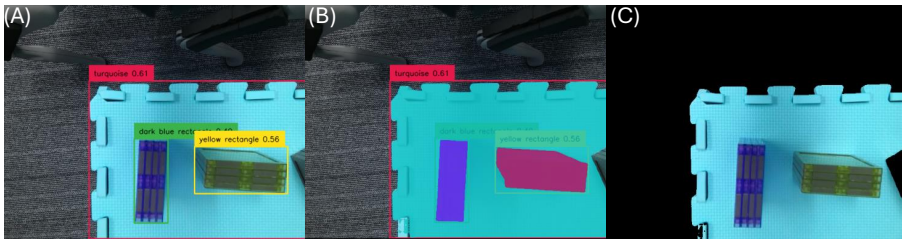


Figure 10: Visual preprocessing pipeline used in tabletop manipulation experiments.

mechanism rejects a stable, non-saturating fraction of candidates rather than collapsing to zero (the threshold never binds) or to one (within-mode distances dominate and the mask saturates).

Proposition F.2 (Rejection-rate concentration and within-mode suppression)**.** *Under (A1)–(A3):*

(a) Rejection-rate concentration. *The per-pair fraction of off-diagonal distances below ε_{RS} equals q by calibration. The per-candidate rejection rate $\bar{\rho} = \mathbb{E}[(BK)^{-1} \sum_{i,k} \mathbb{I}^{\text{rej}}_{i,k}]$ rejects a candidate when at least one of its $B-1$ off-diagonal distances falls below ε_{RS}, so $\bar{\rho} \le 1-(1-q)^{B-1} = (B-1)q\,(1+o(1))$ and concentrates around its mean with fluctuations $O(1/\sqrt{N})$. Within-context correlation of candidate distances makes the realized rate smaller than this i.i.d. bound (audited at* 0.18–0.32*, App. C.4).*

(b) Within-mode suppression bound. *For any mode M_k and any candidate $\hat{\mathbf{A}}_{i,k'} \in M_k$,*

$$P(\textit{within-mode suppression} \mid \hat{\mathbf{A}}_{i,k'} \in M_k) \;\le\; (B-1)\,\pi_k\,q \cdot (1+o(1)).$$

(c) Operating regime. *At $q = 0.005$ and $B = 128$, a mode of mass $\pi_k = 0.1$ has per-candidate within-mode suppression probability $\lesssim 0.064$. The bound scales linearly in q, justifying the small operating value: halving q halves the suppression probability for every mode. The $K = 8$ candidates per example share a context, generator, and threshold, so we do not assume independence across candidates; empirically the rejection-rate audit (App. C.4) shows ρ_t never approaches 1, so total mode collapse does not occur in practice.*

Proof. Part (a). By construction, ε_{RS} is the EMA-smoothed q-quantile of $\{\tilde{D}_{i,k\to j}\}_{i\neq j,k}$. By Lemma F.1, the empirical per-pair fraction below ε_{RS} converges to q at rate $O(1/\sqrt{N})$. A candidate is rejected iff the minimum of its $B-1$ off-diagonal distances is below ε_{RS}; a union bound gives per-candidate rejection probability at most $1-(1-q)^{B-1} = (B-1)q\,(1+o(1))$, and the $O(1/\sqrt{N})$ quantile fluctuations give the stated concentration.

Part (b). Fix candidate $\hat{\mathbf{A}} \in M_k$. Within-mode suppression requires $\exists\, j \neq i$ with $\mathbf{A}_j \in M_k$ and $\tilde{D}_{i,k'\to j} < \varepsilon_{\text{RS}}$. Two bounds hold simultaneously: (i) *Mode-mass bound:* by union over $B-1$ targets, each in M_k with probability π_k, and each satisfying the distance condition with probability $\le q$ by (A2) and Part (a), giving $P \le (B-1)\pi_k \cdot q \cdot (1+o(1))$. (ii) *Global-rate bound:* within-mode suppression $\subseteq$ global rejection, so $P \le 1-(1-q)^{B-1} = (B-1)q(1+o(1))$. Since $\pi_k \le 1$, the mode-mass bound (i) is the tighter of the two: $P \le (B-1)\pi_k q \cdot (1+o(1))$.

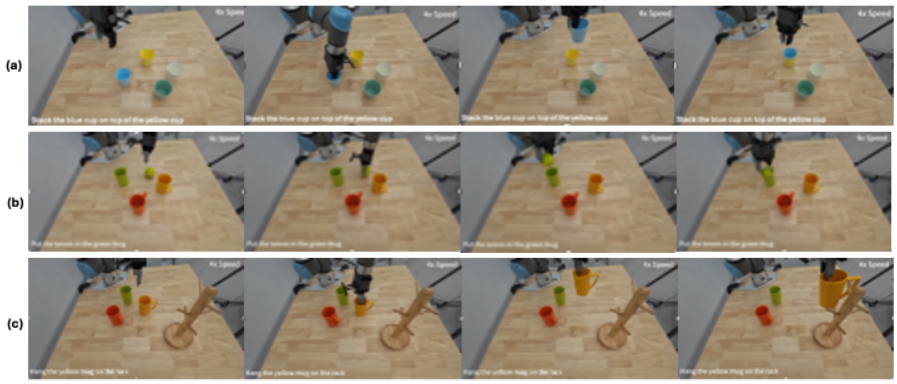


Figure 11: Language-conditioned manipulation rollouts. Commands match Table 16.

Table 17: Cosine similarity between "push [object] left" and "push [object] right" across frozen text encoders. All encoders collapse the directional distinction.

Text Encoder	Cosine Sim. (↑ = more collapsed)
CLIP ViT-B/32	0.97
CLIP ViT-L/14	0.96
Sentence-BERT	0.95
MiniLM-L6	0.96
all-mpnet-base-v2	0.95

Part (c). Direct substitution of $q = 0.005$ and $B = 128$ into (b). □

Remark. The audited $\rho_t \in [0.18, 0.32]$ substantially exceeds the within-mode bound $(B-1)\pi_k q \approx 0.064$ for typical modes, indicating that most rejections fire on cross-mode candidates, which is the intended effect of the mechanism.

F.4 Soft Coverage

The top-K' soft-coverage term (Eq. 8) is a temperature-controlled smoothing of nearest-candidate assignment: as $\tau \to 0^+$, $-\tau \log \sum \exp(-D_k/\tau)$ recovers $\min_k D_k$, while $\tau > 0$ distributes gradient across the top-K' candidates with weights proportional to $\exp(-D_k/\tau)$. Its empirical effect is evaluated in App. C.2.

F.5 Performer Approximation

FAVOR$^+$ provides an unbiased positive-random-feature estimate of the softmax attention kernel, with variance decreasing as the number of random features increases [36]. This gives PRISM an efficient full-sequence generator. It does not by itself guarantee closed-loop temporal stability; stability is evaluated empirically through the execution-horizon and ProxyScore ablations.

G Failure Analysis

We analyze CALVIN failures to identify remaining failure modes. Failures are rarely due to kinematically invalid action outputs; they more often arise from indecision among plausible candidates during receding-horizon execution. This motivates future work on more history-aware candidate selection and uncertainty-aware replanning.

Failure categories. Mode-switching instability accounts for roughly 10–15% of failures, high-jerk transitions for 5–10%, and sensor occlusion for 8–12%. These categories overlap, so they are reported

as diagnostic ranges rather than as a partition summing to 100%. Remaining failures ($\sim$55–75%) involve compounded factors that do not cleanly attribute to a single dominant category.