MIRROR: Multisensory Implicit Rejection-sampled RObotic policy

Amisha Bhaskar [1] **Pratap Tokekar** [1] **Stefano Di Cairano** [2] **Alexander Schperberg** [2]

Abstract

Robotic imitation learning typically requires models that capture multimodal action distributions while operating at real-time control rates and accommodating multiple sensing modalities. Although recent generative approaches such as diffusion models, flow matching, and Implicit Maximum Likelihood Estimation (IMLE) have achieved promising results, they often satisfy only a subset of these requirements. To address this, we introduce MIRROR, a single-pass policy based on a batch-global rejection-sampling variant of IMLE. MIRROR couples a temporal multisensory encoder (integrating RGB, Depth, tactile, audio, and proprioception) with a linear-attention generator using a Performer architecture. We demonstrate the efficacy of MIRROR on a diverse real-world hardware suite, including **loco-manipulation using a Unitree GO2 with a 7-DoF arm D1** and **tabletop manipulation with a UR5 manipulator**. Across challenging physical tasks such as pre-manipulation parking, high-precision insertion, and multi-object pick-and-place, MIRROR outperforms state-of-the-art diffusion policies by **10–25% in success rate** while maintaining high-frequency (**30–50 Hz**) closed-loop control. We further validate our approach on large-scale simulation benchmarks, including CALVIN, MetaWorld, and Robomimic. In CALVIN (10% data split), MIRROR improves success rates by $\sim$25% over diffusion and $\sim$20% over flow matching, while simultaneously **reducing trajectory jerk by** $20\times$**–**$50\times$. These results position MIRROR as a fast, accurate, and multisensory imitation policy that retains multimodal action coverage without the latency of iterative sampling.website

[1]University of Maryland, College Park, College Park, MD 20742, USA [2]Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139, USA. Correspondence to: Amisha Bhaskar <amishab@umd.edu>, Pratap Tokekar <tokekar@umd.edu>, Stefano Di Cairano <dicairano@merl.com>, Alexander Schperberg <schperberg@merl.com>.

Proceedings of the 43^{rd} International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026.

1. Introduction

Imitation learning provides a practical route for acquiring robot policies from demonstrations (Atkeson & Schaal, 1997; Argall et al., 2009; Rahmatizadeh et al., 2018; Avigal et al., 2022; Yu et al., 2025; Bhaskar et al., 2024). For real-world manipulation, however, a policy must satisfy several constraints simultaneously. It should represent multiple valid behaviors in the demonstrations, such as different grasps or approach paths; it should run fast enough for closed-loop control; and it should make effective use of sensory histories from RGB, depth, tactile sensing, and proprioception when these modalities are available. Standard behavior cloning is efficient, but mean-squared losses can average distinct behaviors into low-quality actions. Diffusion and flow-based policies improve multimodal action modeling (Chi et al., 2025; Ze et al., 2024; Song et al., 2021; Hu et al., 2024; Funk et al., 2024; Zhang et al., 2025; Yang et al., 2024), but iterative denoising or numerical integration increases inference cost and can limit control frequency (Lu et al., 2024; Prasad et al., 2024; Rouxel et al., 2024).

IMLE offers an alternative by training a generator so that each demonstration is close to at least one generated sample (Li & Malik, 2018). Recent IMLE Policy work applies this idea to conditional visuomotor imitation and uses RS–IMLE to improve sample diversity (Rana et al., 2025; Vashist et al., 2024). This gives single-pass inference, but two issues remain important for robot deployment. First, per-sample rejection compares candidates only to the paired target trajectory, which does not fully exploit the minibatch distribution and can still produce averaged behaviors in multimodal settings. Second, independently generated action chunks can switch between valid modes across replanning steps, producing jitter or inconsistent execution (Rhinehart et al., 2019; Chai et al., 2020; Rana et al., 2025).

We present **Multisensory Implicit Rejection-sampled RObotic policy** (MIRROR), a single-pass multisensory imitation policy for multimodal robot control. MIRROR encodes short histories of available sensory inputs into per-timestep context tokens and predicts a full future action

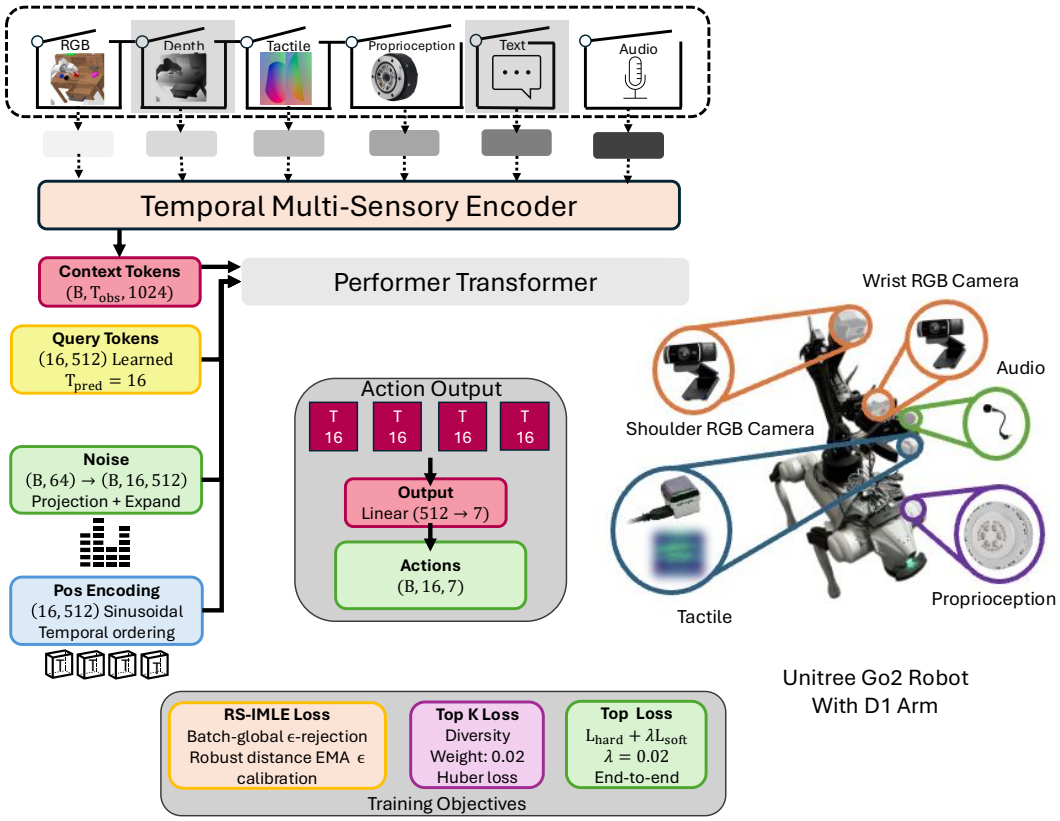

Figure 1. **Overview of Multisensory Implicit Rejection-sampled RObotic policy (MIRROR).** Per-timestep features from the available sensors are fused into temporal context tokens. A bidirectional FAVOR$^+$ generator uses learned action query tokens to output a full action sequence in one forward pass. Training uses a batch-global RS–IMLE objective with robust sequence distances and EMA-calibrated rejection to preserve multimodal demonstrations without iterative sampling. At inference, MIRROR samples multiple latent-conditioned action sequences in a batched pass and selects one for receding-horizon execution.

sequence using learned query tokens with bidirectional linear attention (FAVOR$^+$) (Choromanski et al., 2020). The policy is trained with a batch-global RS-IMLE objective: generated candidates are compared against all target trajectories in the minibatch, and an EMA-calibrated rejection threshold discourages repeatedly fitting already-covered targets. At inference, the model produces multiple trajectory candidates in one batched forward pass and uses receding-horizon execution to replan from latest observations. Our contributions:

- We introduce MIRROR, a single-pass sequence policy that combines per-timestep multisensory encoding with a Performer/FAVOR$^+$ action generator for non-autoregressive trajectory prediction.
- We propose a batch-global RS–IMLE objective for conditional imitation learning, using robust sequence distances and EMA-calibrated rejection to improve coverage of multimodal demonstrations without adding iterative sampling at test time.
- We report results on MetaWorld (Yu et al., 2020), CALVIN (Mees et al., 2022), Robomimic Proficient-Human tasks (Mandlekar et al., 2022), and real-robot loco-manipulation and tabletop manipulation. Across these settings, MIRROR is competitive with or improves over diffusion, flow-matching, and IMLE-style baselines, with the largest gains appearing in multimodal and closed-loop settings.

2. Related Works

Behavior cloning and multimodal actions. Behavior cloning provides fast single-pass control by mapping observations directly to actions (Zhang et al., 2018; Florence et al., 2019), but unimodal regression losses can average distinct expert strategies in contact-rich tasks. Prior work improves expressivity through discretized action prediction (Zeng et al., 2021; Wu et al., 2020), mixture or transformer-based behavior models (Mandlekar et al., 2022; Shafiullah et al., 2022), and implicit energy-based policies (Florence et al.,

2022). These methods motivate modeling multimodal action distributions rather than a single mean action. MIRROR follows this direction while preserving single-pass inference through batch-global RS–IMLE.

Generative policy learning and IMLE. Diffusion and flow-based policies model complex visuomotor action distributions (Chi et al., 2025; Song et al., 2021; Liu et al., 2023b), with recent accelerations based on equivariant transformers, adaptive solvers, distillation, or step reuse (Funk et al., 2024; Hu et al., 2024; Chen et al., 2025; Prasad et al., 2024). 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 (Goodfellow et al., 2014; Chen et al., 2023). IMLE instead trains a generator by matching each data point to a nearby generated sample (Li & Malik, 2018), and RS–IMLE improves sample selection through rejection sampling (Vashist et al., 2024). MIRROR extends this line 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 (Avigal et al., 2022; Yu et al., 2025). At the same time, multimodal policies can switch between valid behaviors during execution, producing jitter or inconsistent action chunks (Rhinehart et al., 2019; Chai et al., 2020). Attention-based sequence policies can model temporal context, and linear-time variants such as Performer make longer contexts more efficient (Choromanski et al., 2020). MIRROR uses per-timestep multisensory fusion with bidirectional linear attention and receding-horizon candidate selection to produce coherent action sequences in real time.

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:

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

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

MIRROR 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 $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_{T_o}] \in \mathbb{R}^{T_o \times d}$ 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, allowing the model to coordinate early and late parts of the trajectory within a single forward pass.

Both self-attention and cross-attention use FAVOR$^+$ linear attention (Choromanski et al., 2020). This keeps the trajectory generator efficient as the context and prediction horizons grow, reducing attention cost from quadratic in sequence length to linear in sequence length for a fixed random-feature budget. The stabilized implementation is given in Algorithm 1 in Appendix B.2.

4.3. Batch-global RS–IMLE training

For each training example i in a minibatch of size B, we sample K latents and generate candidates $\{\hat{\mathbf{A}}_i^{(k)}\}_{k=1}^K$. We compare action sequences using a scale-normalized Charbonnier distance,

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

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 (Li & Malik, 2018; Rana et al., 2025). In contrast, MIRROR computes distances from every generated candidate to every target in the minibatch. Let $D_{i,k\to j} = D_\rho(\hat{\mathbf{A}}_i^{(k)}, \mathbf{A}_j)$. A candidate is rejected if it is already too close to any target in the batch:

$$\mathbb{I}_{i,k}^{\text{rej}} = \mathbb{I}\left[\min_j D_{i,k\to j} < \varepsilon_{\text{RS}}\right]. \quad (4)$$

The hard training loss is then

$$\mathcal{L}_{\text{hard}} = \frac{1}{B} \sum_{i=1}^{B} \min_{k \in \mathcal{K}_i} D_\rho(\hat{\mathbf{A}}_i^{(k)}, \mathbf{A}_i), \quad (5)$$

where

$$\mathcal{K}_i = \begin{cases} \{k : \mathbb{I}_{i,k}^{\text{rej}} = 0\}, & \text{if nonempty,} \\ \{1, \ldots, K\}, & \text{otherwise.} \end{cases} \quad (6)$$

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

We set ε_{RS} using an exponential moving average of a batch-global distance quantile,

$$\tilde{\varepsilon} = \text{Quantile}_q\left(\{D_{i,k\to j}\}_{i,k,j}\right). \quad (7)$$

$$\varepsilon_{\text{RS}} \leftarrow \text{clip}\left(\alpha\varepsilon_{\text{RS}} + (1-\alpha)\tilde{\varepsilon}, \varepsilon_{\min}, \varepsilon_{\max}\right). \quad (8)$$

We use $q \in [0.2, 0.35]$ and clamps $(\varepsilon_{\min}, \varepsilon_{\max}) = (10^{-4}, 0.2)$. Appendix F.2 provides the corresponding consistency argument and empirical calibration results.

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

$$\mathcal{L}_{\text{soft}} = -\frac{1}{B} \sum_{i=1}^{B} \log \sum_{k \in \text{Top}K'(D_{i,\cdot})} \exp(-D_{i,k}/\tau), \quad (9)$$

and optimize $\mathcal{L} = \mathcal{L}_{\text{hard}} + \lambda_{\text{soft}}\mathcal{L}_{\text{soft}}$ with $\lambda_{\text{soft}} \ll 1$. This term supplies gradients to more than one high-quality candidate while avoiding a full average over all modes. The soft-coverage interpretation and ablations are provided in Appendix F.3 and Appendix C.2.

4.4. Receding-horizon inference

At test time, MIRROR encodes the latest observation window, samples K latents, and generates K candidate action sequences in one batched forward pass. A single candidate is selected using an observation-only ProxyScore when proprioception or end-effector pose is available. The proxy scores each candidate by the consistency of its induced first-step motion with the current robot state or the previously executed action. When this information is unavailable, we use a deterministic smoothness tie-break based on the first predicted action. The selected trajectory is executed for T_a steps, after which the observation window is updated and the policy replans. This produces closed-loop behavior while retaining single-pass action generation. The full inference procedure is in Appendix B.5.

5. Experiments

We evaluate MIRROR on MetaWorld (Yu et al., 2020), CALVIN (Mees et al., 2022), Robomimic Proficient-Human tasks (Mandlekar et al., 2022), and real-robot manipulation tasks. The experiments stress three properties required for deployment: multimodal action generation, multisensory robustness, and low-latency closed-loop execution. We begin with controlled ablations on CALVIN to isolate the contribution of each component. We then compare against diffusion policies (Chi et al., 2025), flow-matching policies (Black et al., 2026), and IMLE Policy (Rana et al., 2025). Full benchmark protocols, modality configurations, per-task tables, and hyperparameter sweeps are provided in Appendix A–D.

5.1. Where do the gains come from?

Before comparing across benchmarks, we isolate the components of MIRROR on CALVIN (Mees et al., 2022), since it includes multimodal observations, long-horizon manipulation, and motion-quality metrics. Table 1 separates the effect of the backbone, training objective, and execution

Table 1. **Attribution of the main gains on CALVIN.** Left: architecture and objective ablation. Batch-global RS–IMLE improves over per-sample RS–IMLE for both UNet and Performer backbones, and the best result requires the Performer backbone with the batch-global objective. Right: execution ablation using the same trained MIRROR model. ProxyScore improves candidate selection at matched execution horizon, and shorter receding-horizon chunks further improve closed-loop correction.

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	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

Table 2. **Main benchmark results.** CALVIN uses 10% of Env D with wrist RGB, static RGB, depth, tactile, and proprioception. MetaWorld reports success across 50 tasks grouped by difficulty. Robomimic reports average success over PH image-based tasks. Higher success is better; lower jerk and switch rate are better.

	CALVIN			MetaWorld				Robomimic PH	
Method	Succ.↑	Jerk↓	Switch↓	Easy↑	Med.↑	Hard↑	V-Hard↑	NFE↓	Avg.↑
Diffusion Policy (Chi et al., 2025)	36.4±9.2	3.783	0.515	83.6	31.1	9.0	26.6	15	0.85
Flow Matching (Black et al., 2026)	44.4±3.7	1.148	0.586	61.4	20.6	13.4	36.0	1	0.85
IMLE Policy (Rana et al., 2025)	56.2±6.5	1.053	0.276	75.6	60.4	43.5	76.0	1	0.87
MIRROR	**65.2±4.3**	**0.052**	**0.101**	**96.4**	**85.5**	**58.0**	**85.8**	**1**	**0.922**

strategy while keeping the remaining setup fixed. The backbone ablation compares a UNet-style trajectory generator with the Performer/FAVOR$^+$ attention backbone (Choromanski et al., 2020); the objective ablation compares MSE behavior cloning, per-sample RS–IMLE, and our batch-global variant, building on IMLE (Li & Malik, 2018) and RS–IMLE (Vashist et al., 2024).

The attribution results show that the improvement is not explained by a single implementation choice. With the Performer backbone fixed, batch-global RS–IMLE improves success from 54.6% to 67.9% over per-sample RS–IMLE. With the objective fixed to batch-global RS–IMLE, replacing the UNet with the Performer improves success from 50.9% to 67.9%. The time-preserving encoder is also necessary: per-timestep fusion reaches 67.9%, compared with 58.6% for early fusion and 50.4% without temporal modeling. These results support the intended division of labor in MIRROR: the encoder preserves temporal multisensory structure, the Performer generates full action sequences efficiently, and batch-global RS–IMLE prevents the generator from collapsing toward averaged trajectories.

The execution ablation addresses whether the gains come only from candidate selection at test time. At the same execution horizon (T_a=8), ProxyScore improves success by 5.1% over random candidate selection. Receding-horizon execution provides an additional benefit by replanning more frequently: success increases to 79.8% at T_a=4 and 83.6% at T_a=1. We report T_a=8 in the main benchmark comparisons because the baselines use the same action-execution horizon; the smaller-T_a rows show the available closed-loop performance tradeoff when additional replanning is allowed.

5.2. Benchmark comparison

Having isolated the main sources of improvement, we compare MIRROR against diffusion policies (Chi et al., 2025), flow-matching policies (Black et al., 2026), and IMLE Policy (Rana et al., 2025) across standard manipulation benchmarks. Table 2 summarizes the main results; full per-task tables and additional baselines are in Appendix D.

On CALVIN, MIRROR improves success by 9.0% over IMLE Policy, 20.8% over Flow Matching, and 28.8% over Diffusion Policy. The larger difference is in motion quality: MIRROR reduces jerk by roughly 20× relative to IMLE Policy and by more than 70× relative to Diffusion Policy, while also reducing the mode-switch rate. This is consistent with the attribution results above: batch-global RS–IMLE preserves multimodal coverage, but the full sequence generator and receding-horizon execution are needed to convert that coverage into smooth closed-loop behavior.

On MetaWorld, MIRROR achieves the best success across all difficulty groups. The gains are largest on Medium, Hard, and Very-Hard tasks, where demonstrations contain more branching strategies and longer contact-rich interactions. Compared with IMLE Policy, MIRROR improves success by 25.1% on Medium tasks, 14.5% on Hard tasks, and

9.8% on Very-Hard tasks. Compared with one-step Flow Matching, the gains are substantially larger, supporting that one-step generation alone is insufficient unless the training objective preserves distinct modes.

Robomimic PH (Mandlekar et al., 2022) provides a useful counterpoint because the demonstrations are more structured and closer to unimodal image-based control. MIRROR remains competitive, reaching 92.2% average success with one-step inference. We do not emphasize Robomimic as the primary evidence for multisensory learning, since PH does not contain the full set of modalities used in CALVIN or the hardware experiments. Instead, it shows that the proposed objective and generator remain accurate even when the data are less multimodal. Extended comparisons, including AdaFlow and other multi-step baselines, are reported in Appendix D.

5.3. Multisensory robustness

We next test whether the temporal multisensory encoder contributes beyond simply adding more input channels. On CALVIN, removing depth has little effect on MIRROR (66.2% versus 65.2% with all modalities), while removing wrist RGB or proprioception causes the largest degradation. Pairwise dropouts show that removing wrist RGB and proprioception together leads to near-complete failure. These results indicate that MIRROR uses modalities unevenly but meaningfully: wrist RGB provides local interaction geometry, proprioception anchors the robot state, and auxiliary modalities provide redundancy when informative. We move the full single- and pairwise-dropout plots to Appendix 4 to keep the main paper focused on the causal takeaway.

5.4. Real-world robot evaluation

Finally, we evaluate whether the simulation gains transfer to hardware. We deploy MIRROR on a Unitree GO2 with a D1 arm and parallel gripper using history $H{=}8$, prediction horizon $T{=}16$, execution chunk $T_a{=}8$, and $K{=}5$ candidates per replanning step. The tasks include pre-manipulation parking, peg-in-hole insertion, and multi-object pick-and-place, followed by tabletop manipulation tasks such as cup stacking and can picking. Figure 2 summarizes the hardware setup and success rates. Across loco-manipulation tasks, MIRROR improves success by 10–25% over the strongest baselines while maintaining real-time execution. Performance also improves as demonstrations increase from 15 to 35: approximately +10% for parking, +20% for peg-in-hole, and +30% for pick-and-place. The tabletop manipulation results show the same trend on more precise manipulation tasks. These hardware results are the deployment check for the earlier ablations: the gains from batch-global training, temporal fusion, and receding-horizon candidate selection translate to physical robot control rather than remaining simulation-only effects.

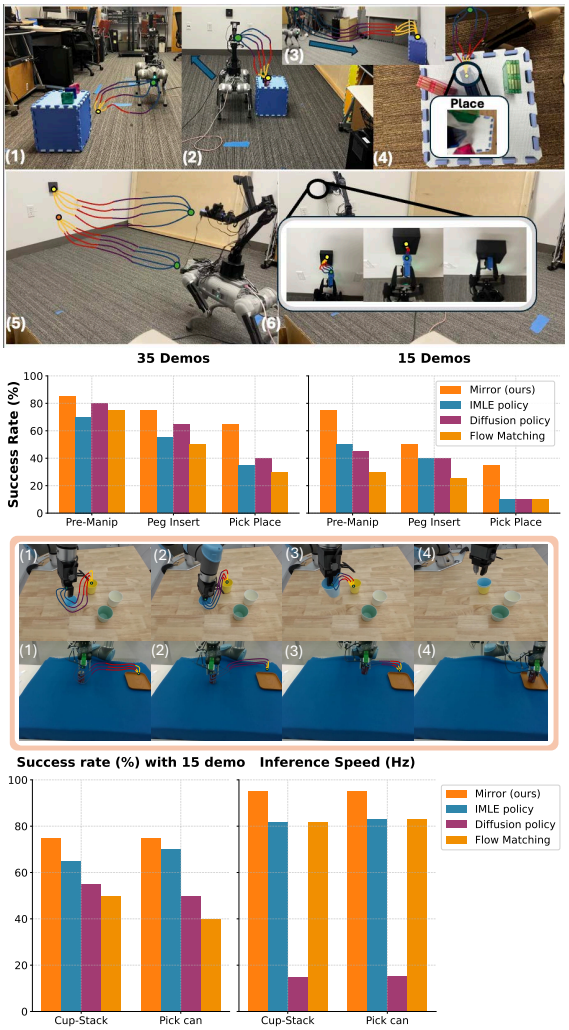

Figure 2. **Real-world hardware evaluation.** From top to bottom: Unitree GO2+D1 loco-manipulation setup and tasks; loco-manipulation success rates over 50 trials per method; tabletop manipulation setup; and tabletop success/inference-speed results. MIRROR improves success by 10–25% over baselines while maintaining real-time closed-loop inference.

6. Conclusion

We presented MIRROR, a single-pass visuomotor policy that fuses multisensory history, generates full action sequences with bidirectional Performer attention, and preserves multimodal coverage via batch-global RS–IMLE. Across MetaWorld, CALVIN, and real loco-manipulation and tabletop suites, MIRROR improves success by **4–25%** over strong diffusion, flow, and prior IMLE baselines while reducing trajectory jerk by up to $20\times$ at real-time rates. Future work includes adaptive rejection control, learned feature maps for linear attention, and RL fine-tuning.

References

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

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

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

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

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

Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L. X., Tanner, J., Vuong, Q., Walling, A., Wang, H., and Zhilinsky, U. π_0: A vision-language-action flow model for general robot control, 2026. URL https://arxiv.org/abs/2410.24164.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Mandlekar, A., Xu, D., Wong, J., Nasiriany, S., Wang, C., Kulkarni, R., Fei-Fei, L., Savarese, S., Zhu, Y., and

Martín-Martín, R. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning*, pp. 1678–1690. PMLR, 2022.

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

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

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

Rahmatizadeh, R., Abolghasemi, P., Bölöni, L., and Levine, S. 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)*, pp. 3758–3765. IEEE, 2018.

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

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

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

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

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

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

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

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

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

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

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

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

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

Zhang, Q., Liu, Z., Fan, H., Liu, G., Zeng, B., and Liu, S. 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, pp. 14754–14762, 2025.

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

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 MIRROR on MetaWorld (Yu et al., 2020), CALVIN (Mees et al., 2022), Robomimic Proficient-Human (PH) (Mandlekar et al., 2022), and a real-world manipulation suite. Table 3 summarizes the modality availability across these settings.

Table 3. Modality availability per benchmark. ✓: available and used; –: not available; Δ: task-dependent.

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

MetaWorld. We use the MetaWorld MT50 benchmark (Yu et al., 2020) 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. CALVIN provides wrist RGB, static RGB, depth, tactile, and proprioception (Mees et al., 2022). 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, MIRROR 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 (Mandlekar et al., 2022), 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 14.

Real-world tasks. We deploy MIRROR on a Unitree GO2+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 $(\omega_x, \omega_y, \omega_z)$.

A.3. Baseline Reproduction

We compare against Diffusion Policy (Chi et al., 2025), Flow Matching (Black et al., 2026), IMLE Policy (Rana et al., 2025), 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 6) use an A100 for standardization.

B. Model, Training, and Inference

This section consolidates the architectural, training, and inference details for MIRROR. Notation is summarized first (Table 4); the encoder, generator, training, and inference details follow.

B.1. Notation

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

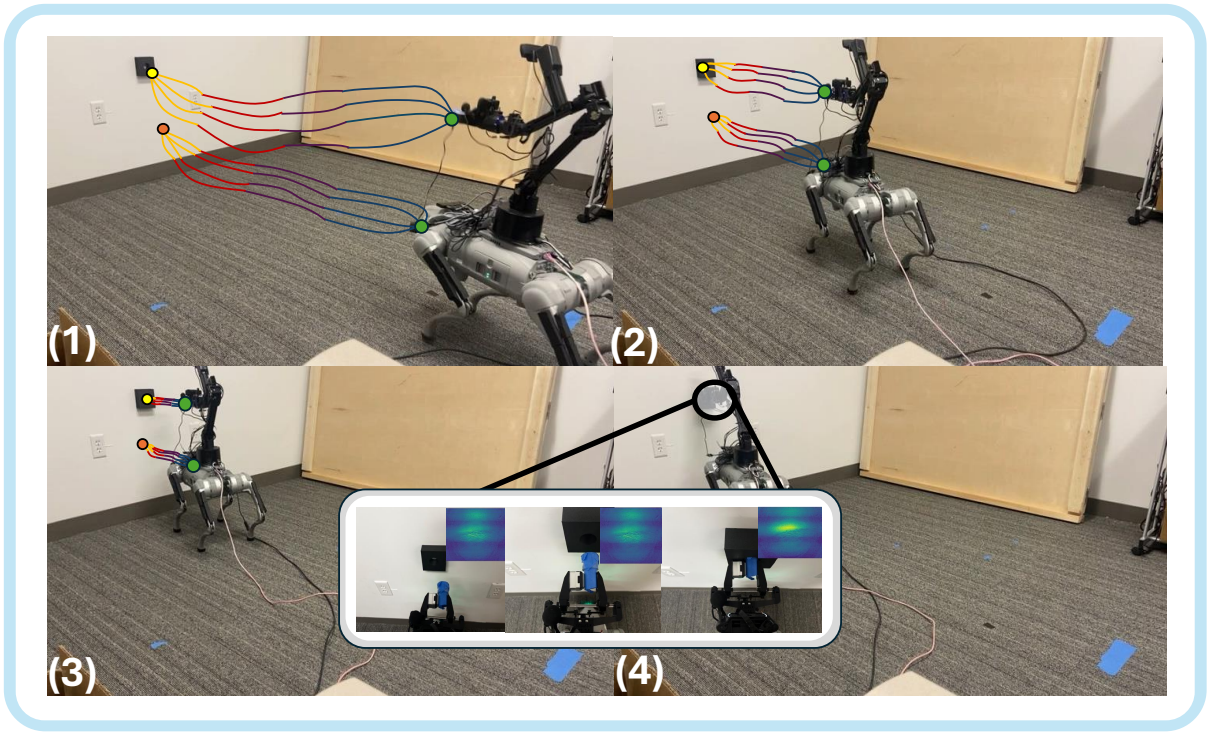

Figure 3. Loco-manipulation platform. Unitree GO2 with a D1 arm during peg insertion.

Table 4. 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
ε_{RS}	EMA-calibrated rejection threshold
q, α	Quantile target and EMA momentum for ε_{RS}

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

Require: Query $\mathbf{Q}$, key $\mathbf{K}$, value $\mathbf{V} \in \mathbb{R}^{L \times d_h}$; random features $\mathbf{W} \in \mathbb{R}^{d_h \times m}$

function FEATURES($\mathbf{X}$)
  $\mathbf{U} \leftarrow \mathbf{X}\mathbf{W}$
  $\mathbf{U} \leftarrow \mathbf{U} - \text{rowmax}(\mathbf{U})$
  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 $(\phi_Q \mathbf{S}) \oslash (\phi_Q \mathbf{n} + \varepsilon_a)$

streams use lightweight convolutional encoders, and proprioception uses a two-layer MLP. The per-timestep fused token is

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

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 (Choromanski et al., 2020); 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 $q \in [0.2, 0.35]$ with EMA momentum $\alpha = 0.9$ and clamps $(\varepsilon_{\min}, \varepsilon_{\max}) = (10^{-4}, 0.2)$.

Algorithm 2 MIRROR 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; linear-attention clamp ε_a
Require: RS quantile $q \in [0.2, 0.35]$, EMA momentum $\alpha \in [0, 1)$, clamps $(\varepsilon_{\min}, \varepsilon_{\max}) = (10^{-4}, 0.2)$
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: $D_{i,k} \leftarrow \frac{1}{T_p} \sum_{t=1}^{T_p} \sum_{d=1}^{D_a} w_d \sqrt{(\hat{a}_{i,t,d}^{(k)} - a_{i,t,d})^2 + \varepsilon_c^2}$
  Batch-global distances: compute $D_{i,k \to j} \leftarrow D_\rho(\hat{\mathbf{A}}_i^{(k)}, \mathbf{A}_j)$ for all i, k, j
  Quantile calibration: $\tilde{\varepsilon} \leftarrow \text{Quantile}_q(\{D_{i,k \to j}\})$; $\varepsilon_{\text{RS}} \leftarrow \text{clip}(\alpha\, \varepsilon_{\text{RS}} + (1-\alpha)\tilde{\varepsilon}, \varepsilon_{\min}, \varepsilon_{\max})$ {Lemma F.2}
  RS mask: $\mathbb{I}_{i,k}^{\text{rej}} \leftarrow \mathbb{I}[\min_j D_{i,k \to j} < \varepsilon_{\text{RS}}]$
  Hard IMLE loss: $\mathcal{K}_i \leftarrow \{k : \mathbb{I}_{i,k}^{\text{rej}} = 0\}$; if $\mathcal{K}_i = \emptyset$ set $\mathcal{K}_i = \{1..K\}$; $\mathcal{L}_{\text{hard}} \leftarrow \frac{1}{B} \sum_{i=1}^{B} \min_{k \in \mathcal{K}_i} D_{i,k}$ {Lemma F.1}
  if *soft coverage* enabled **then**
    $\mathcal{L}_{\text{soft}} \leftarrow -\frac{1}{B} \sum_i \log \sum_{k \in \text{Top}K'(D_{i,\cdot})} \exp(-D_{i,k}/\tau)$; $\mathcal{L} \leftarrow \mathcal{L}_{\text{hard}} + \lambda_{\text{soft}} \mathcal{L}_{\text{soft}}$ {Lemma F.3}
  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

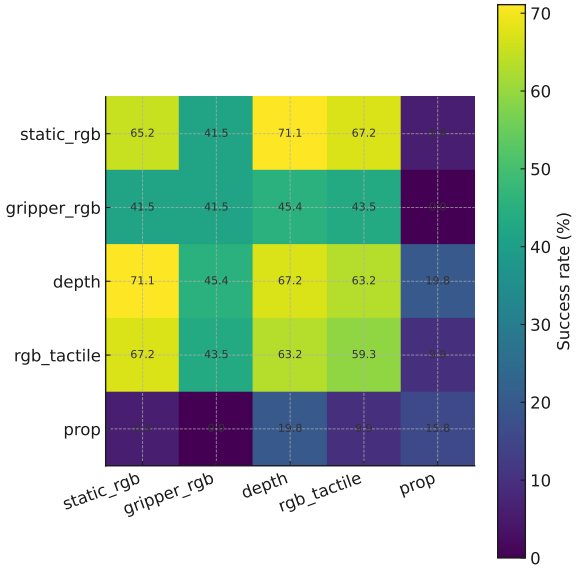

Figure 4. Pairwise modality dropout on CALVIN. The strongest degradation occurs when wrist RGB and proprioception are removed together, indicating that local visual interaction cues and robot-state feedback are complementary.

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 4 shows the pairwise dropout heatmap.

C.2. Architecture and Hyperparameter Ablations

Figure 5 studies training hyperparameters, and Figure 6 studies architecture choices. These ablations support the default settings used in the main experiments.

C.3. Model Size and Inference Latency

D. Extended Benchmark Results

D.1. MetaWorld: Grouped Results

D.2. MetaWorld: Per-Task Results

Tables 8–13 report per-task success rates across all 50 MetaWorld tasks, grouped by difficulty. MIRROR is best or tied-best on the harder tasks, where multi-modal coverage matters most.

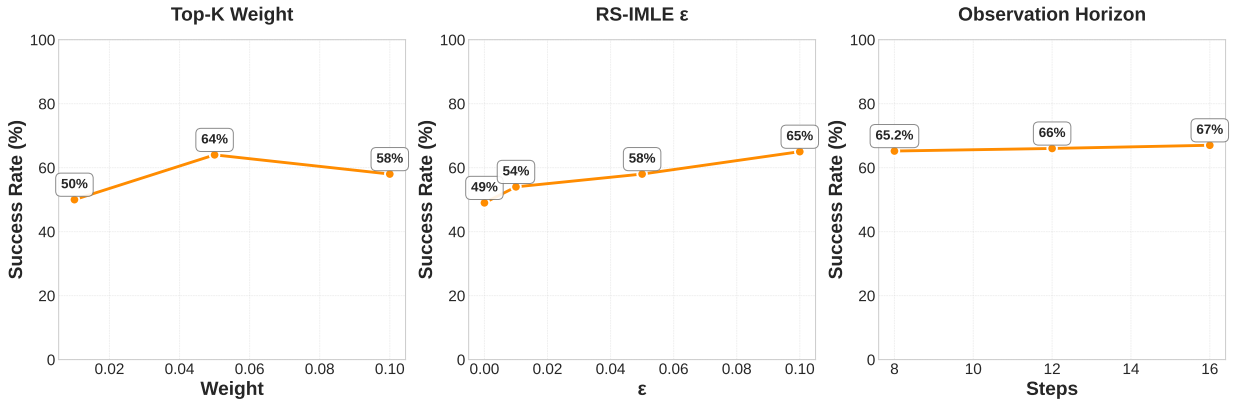

Figure 5. Training hyperparameter ablations on CALVIN: soft-coverage weight, batch-global RS–IMLE threshold calibration, and observation horizon.

Table 5. Model size and memory footprint.

Model	Params (M)	MB	Succ. (%)
MIRROR	**44.4**	**169.4**	**66.0**
UNet RS–IMLE	91.8	350.4	58.8
Diffusion Policy	79.9	305.0	62.0
Flow Matching	79.9	305.0	61.5

Table 6. Inference latency (ms) on NVIDIA A100. A strict 30 Hz loop requires p99 latency $\leq$ 33 ms; MIRROR meets the mean budget at all settings reported and meets the strict p99 budget at T_o=8, T_p=16.

Method	T_o=8, T_p=16		T_o=16, T_p=32		Strict 30 Hz?
	Mean	p99	Mean	p99	
Diffusion Policy (NFE=10)	142.3	168.5	285.7	342.1	No
Flow Matching (NFE=5)	73.8	89.2	147.6	181.3	No
MIRROR (K=8)	**15.0**	**16.0**	**31.2**	35.8	Mean only at T_o=16

D.3. Robomimic (PH): Extended Results

D.4. Push-T Multimodality Visualization

We follow the Push-T setup from IMLE Policy (Rana et al., 2025). 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 7 compares MIRROR against IMLE, Diffusion Policy, and Flow Matching: MIRROR 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 (Liu et al., 2023a) and SAM (Kirillov et al., 2023), then re-use these masks as soft attention priors during deployment. Figure 8 illustrates the pipeline.

E.2. Language-Conditioned Manipulation

For language-conditioned tasks, we encode commands with a frozen CLIP text encoder (Radford et al., 2021) and inject the embedding as an additional context token. Table 15 reports per-command success; Figure **??** shows representative rollouts.

Table 15. Language-conditioned manipulation success rates (50 trials per task).

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

F. Theoretical Notes

F.1. IMLE and Implicit Likelihood

MIRROR builds on IMLE (Li & Malik, 2018), which trains a latent-variable generator by matching each data point to a nearby generated sample. The following standard result motivates the hard IMLE loss in Algorithm 2.

Lemma F.1 (Implicit-likelihood interpretation of IMLE, Li & Malik, 2018)**.** *Let G_θ be a generator with prior $p(z)$ and let D_ρ be a distance such that $\exp(-D_\rho(\mathbf{a}, \mathbf{a}')/h)$ defines a valid kernel. Minimizing $\mathbb{E}_{\mathbf{a}\sim p_{\text{data}}}[\min_k D_\rho(\mathbf{a}, G_\theta(z_k))]$ with $z_k \sim p(z)$ is a consistent surrogate for an implicit maximum-likelihood estimator of the data distribution as $K \to \infty$.*

We use this result as motivation for the batch-global RS–IMLE objective; our empirical claims rely on the ablations

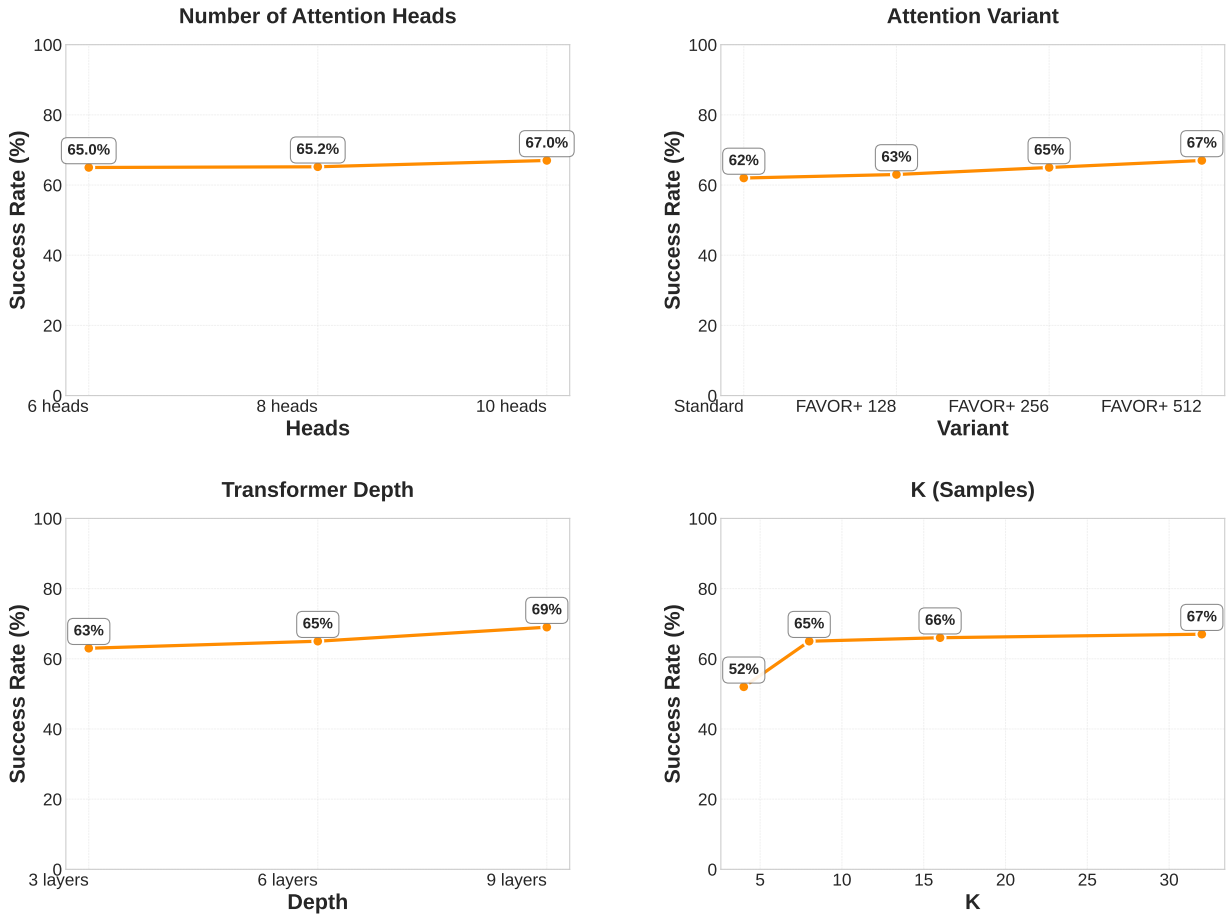

Figure 6. Architecture ablations on CALVIN: attention heads, FAVOR$^+$ random features, transformer depth, and number of candidates K.

in Sec. C rather than on asymptotic guarantees alone.

F.2. Batch-Global Quantile Calibration

Lemma F.2 (Consistency of the batch-global quantile estimator)**.** *Let $\{D_{i,k\to j}\}$ denote the candidate-to-target distances in a minibatch of size N, and 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 (Bahadur, 1966; Van der Vaart, 1998). The result motivates aggregating distances across the minibatch when calibrating $\varepsilon_{\mathrm{RS}}$. □

F.3. Soft Coverage

The top-K' soft-coverage term is a smooth approximation to nearest-candidate assignment. It encourages gradients to flow through multiple high-quality candidates rather than only the current nearest one.

We do not claim the soft term guarantees coverage of every true mode; empirically, its effect is evaluated in Sec. C.2.

Lemma F.3 (Soft coverage as a smoothed nearest-neighbor surrogate)**.** *For temperature $\tau > 0$ and any positive distance set $\{D_k\}_{k=1}^K$, $-\tau \log \sum_{k\in \mathrm{Top}K'} \exp(-D_k/\tau) \xrightarrow{\tau\to 0^+} \min_{k\in \mathrm{Top}K'} D_k$. For $\tau > 0$, the surrogate is differentiable in all top-K' candidates and assigns each a positive gradient weight proportional to $\exp(-D_k/\tau)$.*

Table 7. 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 (Chi et al., 2025)	10	83.6	31.1	9.0	26.6
DP3* (Ze et al., 2024)	10	89.0	72.7	38.0	75.8
ManiCM (Lu et al., 2024)	1	83.6	55.6	33.3	67.0
SDM (Jia et al., 2024)	1	86.5	65.8	35.8	71.6
FlowPolicy* (Zhang et al., 2025)	1	92.1	73.6	46.2	80.0
AdaFlow* (Hu et al., 2024)	1	50.6	19.1	12.6	32.3
Flow Matching* (Black et al., 2026)	1	61.4	20.6	13.4	36.0
IMLE Policy (Rana et al., 2025)	1	75.6	60.4	43.5	76.0
MIRROR	**1**	**96.4**	**85.5**	**58.0**	**85.8**

Table 8. 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
MIRROR	**100 ± 0**	**100 ± 0**	**100 ± 0**	**90 ± 7**	**100 ± 0**	**100 ± 0**

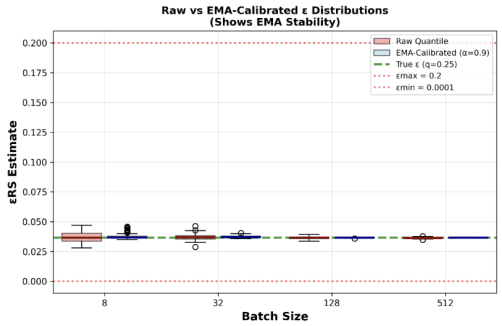

Figure 10. Batch-global rejection-threshold stability as a function of batch size.

F.4. 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 (Choromanski et al., 2020). This gives MIRROR 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%.

Table 9. 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
MIRROR	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{38 \pm 2}$

Table 10. 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
MIRROR	$\mathbf{100 \pm 0}$	$\mathbf{53 \pm 4}$	$\mathbf{76 \pm 2}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{93 \pm 2}$	$\mathbf{55 \pm 7}$	$\mathbf{83 \pm 6}$

Table 11. 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
MIRROR	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{100 \pm 0}$	$\mathbf{98 \pm 2}$	$\mathbf{63 \pm 20}$	$\mathbf{70 \pm 4}$	$\mathbf{93 \pm 2}$	$\mathbf{97 \pm 2}$

Table 12. Per-task success rates on MetaWorld Medium (part 2) and Hard (part 1) tasks.

	Medium					Hard		
Method	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
MIRROR	$\mathbf{100 \pm 0}$	$\mathbf{88 \pm 7}$	$\mathbf{75 \pm 6}$	$\mathbf{49 \pm 6}$	$\mathbf{98 \pm 2}$	$\mathbf{39 \pm 14}$	$\mathbf{100 \pm 0}$	$\mathbf{30 \pm 0}$

Table 13. Per-task success rates on MetaWorld Hard (part 2) and Very Hard tasks, with overall average across all 50 tasks.

	Hard					Very Hard			
Method	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	55.5 ± 3.58
3D Diffusion Policy	44 ± 3	0 ± 0	56 ± 5	0 ± 0	47 ± 2	91 ± 4	67 ± 0	100 ± 0	76.1 ± 2.32
FlowPolicy*	36 ± 6	66 ± 2	61 ± 16	–	46 ± 8	80 ± 4	78 ± 6	100 ± 0	81.5 ± 3.84
ManiCM	30 ± 16	0 ± 0	55 ± 2	0 ± 0	48 ± 3	87 ± 3	63 ± 2	100 ± 0	69.0 ± 4.60
SDM Policy	34 ± 24	0 ± 0	57 ± 0	100 ± 0	51 ± 4	86 ± 10	68 ± 10	0 ± 0	74.8 ± 4.51
MIRROR	$\mathbf{48 \pm 8}$	$\mathbf{63 \pm 6}$	$\mathbf{75 \pm 4}$	**–**	$\mathbf{68 \pm 10}$	$\mathbf{83 \pm 8}$	$\mathbf{78 \pm 4}$	$\mathbf{100 \pm 0}$	**84.2**

Table 14. Robomimic (PH) extended results. Multi-step and one-step baselines compared on the same image-based protocol (50 initializations per task). MIRROR matches strong multi-step baselines while remaining single-pass.

Method	NFE↓	Lift↑	Can↑	Square↑	Transport↑	Tool Hang↑	Avg.↑
Diffusion Policy (Chi et al., 2025)	15	1.00	0.99 ± 0.01	0.92 ± 0.03	0.79 ± 0.04	0.55 ± 0.05	0.85
RectifiedFlow* (Liu et al., 2023b)	15	1.00	0.96 ± 0.02	0.90 ± 0.02	0.84 ± 0.04	0.90 ± 0.02	0.92
Falcon-DDPM (Chen et al., 2025)	12–52	1.00	0.97 ± 0.02	0.95 ± 0.02	0.85 ± 0.04	0.55 ± 0.05	0.86
AdaFlow* (Hu et al., 2024)	∼1.2	1.00	1.00	0.98	0.92	0.88	**0.956**
Consistency Policy (Prasad et al., 2024)	1	1.00	0.98 ± 0.01	0.92 ± 0.02	0.78 ± 0.03	0.70 ± 0.03	0.876
Flow Matching (Black et al., 2026)	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 (Rana et al., 2025)	1	1.00	0.94 ± 0.01	0.81 ± 0.01	0.85 ± 0.05	0.75 ± 0.06	0.87
MIRROR	**1**	**1.00**	**1.00**	0.84 ± 0.02	**0.91 ± 0.02**	**0.86 ± 0.03**	0.922

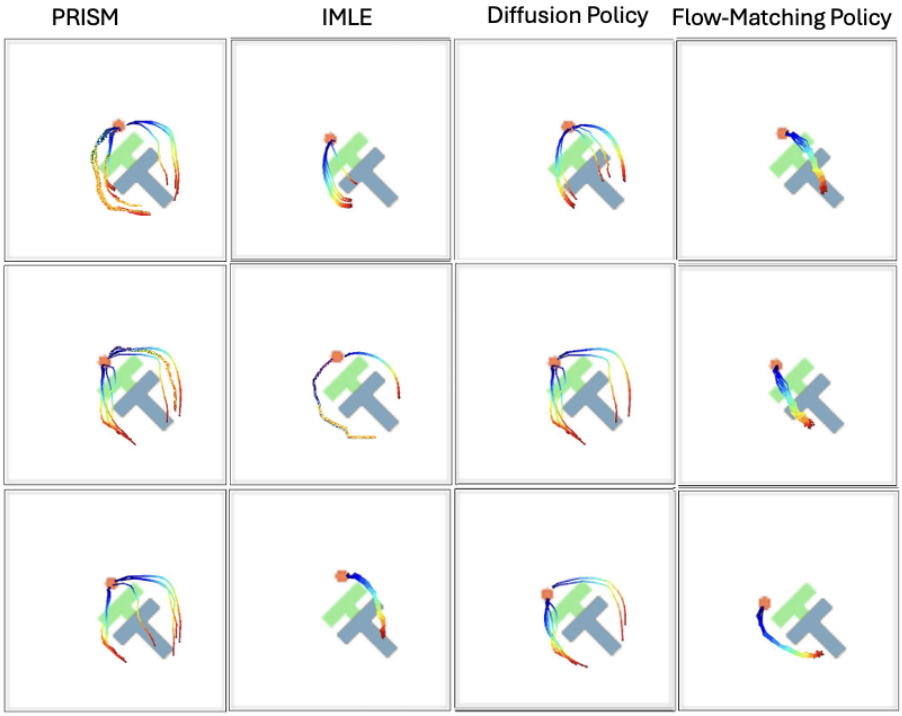

Figure 7. Push-T multimodality visualization. Columns sweep the end-effector start position. MIRROR maintains diverse but smooth trajectory families in ambiguous regions, while Flow Matching is effectively unimodal.

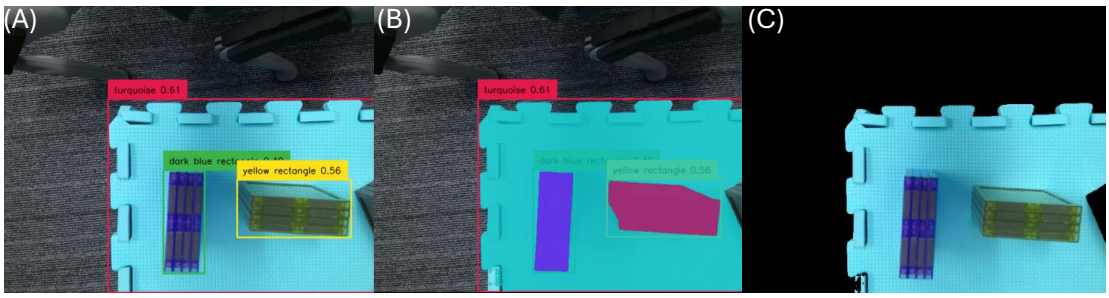

Figure 8. Visual preprocessing pipeline used in tabletop manipulation experiments.