BEYOND CONSERVATISM: RECOVERABILITY-CONDITIONED EXPLORATION FOR MODEL-BASED IMITATION LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Model-based imitation learning (MBIL) improves real-environment interaction efficiency by optimizing policies on imagined rollouts from a learned world model. However, the gap between model-induced and real-environment occupancies makes policy learning sensitive to model error. Conservative MBIL mitigates model exploitation during policy optimization, but when real-environment interactions are collected by the same conservative policy, uncertain regions around the expert distribution remain insufficiently sampled. Generic uncertainty-driven exploration, on the other hand, may allocate interaction to novel but task-irrelevant dynamics. We propose ***RE***coverability-***CON***ditioned Exploration for Model-Based Imitation Learning (**RECON**). RECON separates conservative policy learning from active data collection by maintaining a main policy for task execution and an explorer for real-environment interaction. The explorer is optimized based on epistemic uncertainty conditioned on recoverability estimated from multi-step main-policy imagination, focusing data collection on unknown states from which the main policy can still return toward expert behavior. Experiments on locomotion, navigation and manipulation show consistent gains in interaction efficiency, imitation performance, and robustness, indicating that RECON directs real-environment interaction toward recovery regions around the expert distribution that are underexplored by prior methods, and thereby learns a world model better suited for imitation.

1 INTRODUCTION

Imitation learning (IL) provides a natural way to learn complex behaviors from expert demonstrations without specifying a task reward (Hussein et al., 2017). The simplest approach, behavior cloning (BC), directly fits expert state-action pairs with supervised learning. However, BC is trained on states induced by the expert while being deployed on states induced by the learned policy. Small action errors can therefore move the policy outside the demonstration distribution, where subsequent errors compound (Ross et al., 2011; Seo et al., 2024).

Interactive imitation methods address this distribution shift by incorporating learner-induced data. DAgger (Ross et al., 2011) repeatedly queries the expert on states visited by the current policy. A different line of work builds on inverse reinforcement learning (IRL), where expert behavior is used to infer a reward that can subsequently be optimized with reinforcement learning (Abbeel & Ng, 2004; Finn et al., 2016). Adversarial imitation learning (AIL), such as GAIL (Ho & Ermon, 2016), avoids explicitly recovering a reward function by training a discriminator to distinguish expert and learner transitions and using its output as a surrogate reward. Because policy optimization is performed on learner-induced rollouts, these methods can correct deviations from the expert distribution, but typically require substantial real-environment interaction, which can be costly, time-consuming, and potentially unsafe in real-world systems (Dulac-Arnold et al., 2019).

Model-based imitation learning (MBIL) improves this interaction efficiency by learning predictive dynamics and replacing many real rollouts with model-based optimization or imagined rollouts (Englert et al., 2013; Rafailov et al., 2021). V-MAIL (Rafailov et al., 2021) learns a latent world model from demonstrations and environment interaction, and performs

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

adversarial policy optimization on imagined on-policy rollouts, while real interaction is used primarily to improve the dynamics model. The learned model, however, introduces another source of error: prediction errors can accumulate over imagined rollouts and be exploited by the policy, and the occupancy induced by the learned dynamics can differ from that induced by the real environment (Janner et al., 2019; Ma et al., 2023).

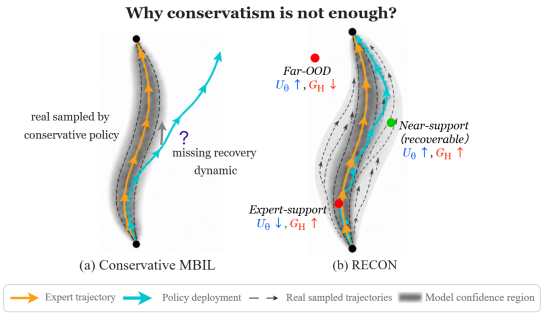

Figure 1: **Conservative model use is not sufficient for model acquisition.** A conservative main policy avoids uncertain dynamics, leaving recovery-relevant regions underexplored. RECON uses a separate explorer to collect uncertain but recoverable transitions.

This problem is closely related to model exploitation in model-based RL. Conservative model-based methods such as MOPO (Yu et al., 2020) and MOReL (Kidambi et al., 2020) account for model error when optimizing policies from limited data, typically by penalizing or avoiding uncertain model predictions. CMIL (Kolev et al., 2024) brings this principle to adversarial imitation learning by augmenting the discriminator-derived imitation reward with an ensemble-based epistemic uncertainty penalty during policy optimization. This naturally raises the *model usage* question:

> ***Usage question:*** *Given the current world model, which predictions can be safely used for policy optimization?*

Conservative optimization reduces the risk of exploiting an inaccurate model, but does not actively improve the model outside its current reliable region. When real-environment data are collected by the same conservative policy, interaction is concentrated in regions that are already well modeled, while uncertain dynamics around the expert distribution may remain insufficiently sampled (Mark et al., 2023; Chen et al., 2025). This matters because the model-induced and real-environment occupancies need not coincide: action noise, observation error, or unmodeled disturbances can move the deployed policy into states for which the world model has little data support. Conservative MBIL therefore leaves a complementary *model acquisition* question:

> ***Acquisition question:*** *Which unknown dynamics should be prioritized under a limited budget of real-environment interaction?*

Model-based exploration provides a natural starting point. MAX (Shyam et al., 2019) and Plan2Explore (Sekar et al., 2020) use information gain or model disagreement to actively collect informative transitions. Their objective is intentionally task-agnostic, aiming to learn broadly useful dynamics models that can support unknown downstream tasks. In imitation learning, however, expert demonstrations already provide a strong task prior (Ren et al., 2024). Under a limited interaction budget, highly uncertain states far from expert behavior may improve global model accuracy while providing little benefit to the imitation policy (Ma et al., 2023; Aoyama et al., 2025).

We therefore focus exploration on states that are both uncertain and *recoverable*. A state is recoverable when the current main policy can still return from it toward expert-compatible behavior (Park & Wong, 2022; Ankile et al., 2024). Epistemic uncertainty measures the information value of acquiring its dynamics, while recoverability measures its relevance to the imitation task.

Based on this principle, we propose **Recoverability-Conditioned Exploration for Model-Based Imitation Learning (RECON)**. RECON uses a conservative main policy for task execution and trains a separate explorer solely for real-environment data collection. Recoverability is estimated through multi-step rollouts of the main policy in the Dreamer world model, using the imitation discriminator to evaluate whether the imagined trajectory returns toward expert-compatible behavior. The explorer prioritizes states with both high epistemic uncertainty and high recoverability. The two policies share the world model, representation, discriminator, and replay buffer, allowing exploratory interactions to improve the model used by the main policy without introducing the exploration objective at deployment.

Our contributions are threefold. (a) we distinguish *model usage* from *model acquisition* in conservative model-based imitation and formalize the resulting blind spot in recovery-relevant dynamics. (b) we introduce recoverability-conditioned exploration, using multi-step Dreamer imagination to direct an independent explorer toward uncertain but task-relevant states. (c) under matched interaction budgets, we show on DMC and MetaWorld that RECON improves imitation performance, interaction efficiency, and robustness, and learns a world model with better coverage of recovery-relevant dynamics.

2 Related Work

2.1 Reinforcement and imitation learning

Reinforcement learning learns control policies by maximizing a task reward through environment interaction, whereas inverse reinforcement learning infers a reward function from expert behavior (Abbeel & Ng, 2004; Finn et al., 2016; Fu et al., 2017). Imitation learning instead seeks to reproduce expert behavior directly. BC is simple and interaction-free but suffers from covariate shift. Interactive methods such as DAgger (Ross et al., 2011) obtain corrective supervision on learner-induced states, while DART (Laskey et al., 2017) perturbs demonstrations to expose the learner to nearby deviations. Adversarial methods such as GAIL (Ho & Ermon, 2016) formulate imitation as occupancy matching and optimize a discriminator-derived reward with RL; off-policy variants such as DAC (Kostrikov et al., 2018) improve data reuse but still depend on real-environment rollouts. Our work follows this occupancy-matching view but focuses on improving the dynamics model used for interaction-efficient policy learning.

2.2 World models and conservative model-based RL

Model-based RL learns environment dynamics and uses the model for planning or policy optimization (Hafner et al., 2019b). Modern latent world-model methods, including Dreamer (Hafner et al., 2019a; 2020; 2023) and TD-MPC (Hansen et al., 2022; 2024), learn compact recurrent state representations and train policies on imagined trajectories. Such model-based optimization is sample efficient but is vulnerable to model exploitation when the policy visits regions poorly supported by data (Janner et al., 2019; Yu et al., 2020).

This issue has been studied extensively in offline and data-limited model-based RL. MOReL (Kidambi et al., 2020) discourages policies from entering uncertain regions, while MOPO (Yu et al., 2020) optimizes an uncertainty-penalized model MDP. COMBO (Yu et al., 2021) similarly introduces conservatism into model-based offline policy optimization. These methods share the principle that policy optimization should account for epistemic model error rather than treating all model predictions as equally reliable. RECON adopts the same principle for the deployed policy, but addresses a complementary problem: how online interaction should be allocated to improve the model itself.

2.3 Model-based imitation learning and active model acquisition

Model-based imitation combines the sample efficiency of world models with learning from demonstrations. Earlier approaches combine learned dynamics with adversarial or trajectory-based imitation (Englert et al., 2013; Baram et al., 2016), while V-MAIL (Rafailov et al., 2021) uses a variational latent model to generate approximate on-policy rollouts for adversarial imitation from high-dimensional observations. CMIL (Kolev et al., 2024) extends this framework with conservative policy optimization, using ensemble uncertainty to reduce model exploitation. Other recent methods explore complementary aspects of MBIL (Kidambi et al., 2021; Hu et al., 2022; Zhang et al., 2023). DITTO (DeMoss et al., 2023) performs fully offline imitation through latent trajectory matching, SeMAIL (Wan et al., 2023) separates task-relevant dynamics from visual distractors, and IQ-MPC (Li et al., 2025) combines reward-free world-model learning with inverse soft-Q learning and latent model predictive control. These methods primarily modify the imitation objective, representation, or controller. RECON instead studies the data distribution used to learn the world model under a limited real-interaction budget.

Model-based exploration provides a natural acquisition mechanism: MAX (Shyam et al., 2019) connects ensemble disagreement to information gain, while Plan2Explore (Sekar et al., 2020) seeks

expected future novelty through latent imagination. Whereas these methods pursue broad, task-agnostic coverage, RECON conditions epistemic uncertainty on the task information in expert demonstrations and prioritizes dynamics relevant to imitation. Related recovery methods use corrective demonstrations or backward and reverse-model augmentation to improve behavior outside nominal expert trajectories (Laskey et al., 2017; Park & Wong, 2022; Shao et al., 2024). RECON instead uses recoverability predicted in imagination, requiring neither a reverse dynamics model nor synthetic policy labels.

3 Problem Formulation and Analysis

Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, T, \mu_0, \gamma)$ be the true MDP and $\widehat{\mathcal{M}}_{\mathcal{D}}$ the model learned from real transitions $\mathcal{D}$. For analysis only, assume a bounded latent task reward $|r(s,a)| \le R_{\max}$ under which the expert π_E is optimal; this reward is never observed by the learner. For initial distribution q, define the normalized occupancy $\rho^{\pi}_{\mathcal{M},q}$ and one-step error

$$\rho^{\pi}_{\mathcal{M},q}(s,a) = (1-\gamma)\sum_{t\ge 0}\gamma^t \Pr(s_t = s, a_t = a \mid s_0 \sim q), \qquad e_{\mathcal{D}}(s,a) = D_{\mathrm{TV}}(T, \widehat{T}_{\mathcal{D}})(s,a). \quad (1)$$

3.1 The acquisition blind spot

The standard simulation argument gives a bound whose model term is evaluated under *real*, rather than imagined, visitation.

Theorem 1 (Real-environment model error)**.** *Let* $\varepsilon_q(\pi, \mathcal{D}) = \mathbb{E}_{\rho^{\pi}_{\mathcal{M},q}}[e_{\mathcal{D}}]$. *Then*

$$|V^{\pi}_{\mathcal{M},q} - V^{\pi}_{\widehat{\mathcal{M}}_{\mathcal{D}},q}| \le \frac{2\gamma R_{\max}}{(1-\gamma)^2}\varepsilon_q(\pi, \mathcal{D}), \quad (2)$$

and

$$V^{\pi_E}_{\mathcal{M}} - V^{\pi}_{\mathcal{M}} \le \frac{2R_{\max}}{1-\gamma} D_{\mathrm{TV}}(\rho_E, \rho^{\pi}_{\widehat{\mathcal{M}}_{\mathcal{D}}}) + \frac{2\gamma R_{\max}}{(1-\gamma)^2}\varepsilon_{\mu_0}(\pi, \mathcal{D}). \quad (3)$$

For $d_0 = \rho^{\pi_m}_{\mathcal{M},\mu_0}$ *and* $d_q = \rho^{\pi_m}_{\mathcal{M},q}$, *if* $d_q \ll d_0$ *with* $\kappa_q := \|\mathrm{d}d_q/\mathrm{d}d_0\|_\infty < \infty$, *then* $\varepsilon_q \le \kappa_q \varepsilon_{\mu_0}$; *without this bounded density-ratio condition, no finite distribution-free factor is guaranteed.*

Thus conservatism controls which model predictions are used, but nominal collection controls recovery error only where the recovery occupancy is covered. Proofs of all results are in Appendix A.

3.2 Recovery-weighted acquisition

At acquisition round k, freeze the current main policy and an expert-compatibility score $c_k \in [0,1]$. The H-step recoverability of a candidate state and its transition-level gate are

$$G^{\pi_m}_{k,H}(s) = \frac{1}{C_H}\mathbb{E}_{\widehat{\mathcal{M}}_{\mathcal{D}_k},\pi_m}\left[\sum_{h=1}^{H}\gamma^{h-1}c_k(s_h,a_h) \mid s_0 = s\right], \quad g_k(s,a) = \mathbb{E}_{\widehat{T}_{\mathcal{D}_k}}[G^{\pi_m}_{k,H}(s')], \quad (4)$$

where $C_H = \sum_{h=1}^{H}\gamma^{h-1}$. Let T_ω index posterior-plausible dynamics and define posterior predictive KL $\ell_{\mathcal{D}}(x) = \mathbb{E}_{\omega|\mathcal{D}} D_{\mathrm{KL}}(T_\omega(\cdot \mid x) \| \widehat{T}_{\mathcal{D}}(\cdot \mid x))$. For a reference distribution ν over candidate acquisition inputs, keep g_k fixed within round k and set

$$\mathcal{R}_k(\mathcal{D}) = \mathbb{E}_{x\sim\nu}[g_k(x)\ell_{\mathcal{D}}(x)]. \quad (5)$$

Here ν serves as a reference measure over transitions. In the practical algorithm, its empirical analogue is the replay-seeded imagined transition distribution visited by the explorer.

Theorem 2 (Recovery-weighted control)**.** *If* $\rho^{\pi_m}_{T_\omega,q}(x) \le C_q\nu(x)$ *almost surely, then, with* $Z_k = \mathbb{E}_\nu[g_k]$ *and* $\bar{e}_k(x) = \mathbb{E}_{\omega|\mathcal{D}_k} D_{\mathrm{TV}}(T_\omega, \widehat{T}_{\mathcal{D}_k})(x)$,

$$\mathbb{E}_{\omega|\mathcal{D}_k}[\varepsilon_q(\omega, \mathcal{D}_k)] \le C_q\left[\sqrt{Z_k\mathcal{R}_k(\mathcal{D}_k)/2} + \mathbb{E}_\nu[(1-g_k)\bar{e}_k]\right]. \quad (6)$$

The first term is precisely the part of predictive risk that can affect states judged recoverable by the current policy; the residual makes explicit that the criterion does not seek an everywhere-accurate model.

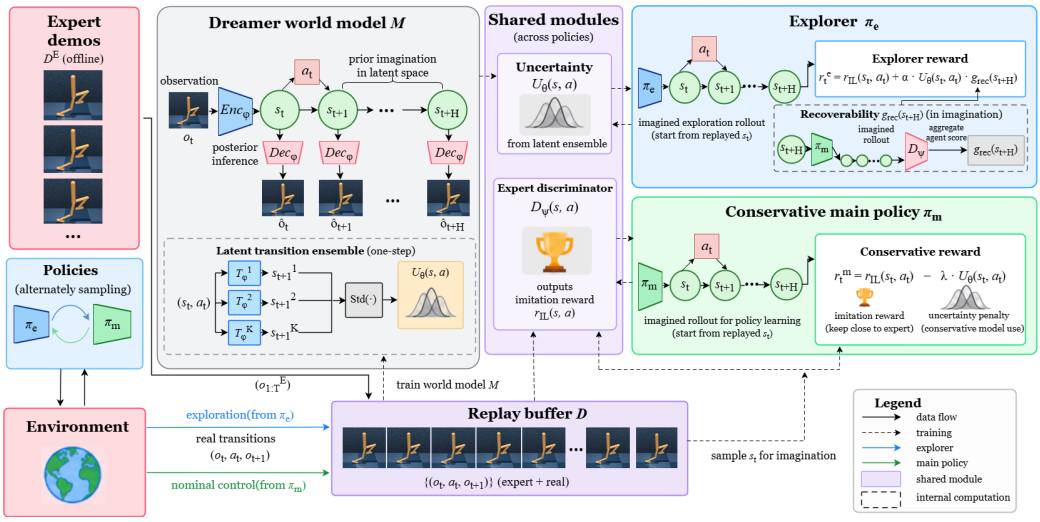

Figure 2: **RECON architecture on the DreamerV2 backbone**. The conservative main policy uses uncertainty pessimistically, while a separate explorer uses recoverability-gated uncertainty for acquisition. Both update one world model; only the main policy is deployed.

For a real query x with successor Y_x, let $\mathcal{D}'_{k,x} = \mathcal{D}_k \cup \{(x, Y_x)\}$ and let Y_z be an independent successor conditional on ω. Bayesian risk reduction yields:

Theorem 3 (Recovery-weighted information gain)**.**

$$\mathcal{R}_k(\mathcal{D}_k) - \mathbb{E}_{Y_x|\mathcal{D}_k}[\mathcal{R}_k(\mathcal{D}'_{k,x})] = \mathbb{E}_{z\sim\nu}[g_k(z)I(Y_z; Y_x \mid z, x, \mathcal{D}_k)]. \tag{7}$$

For a discrete, posterior-independent cell model whose successors are deterministic given ω, this reduces to $\Gamma_k(x) = \nu(x)g_k(x)I(\omega; Y_x \mid x, \mathcal{D}_k)$.

Approximating information gain by ensemble disagreement U_θ gives the complementary objectives

$$J_m = J_{\mathrm{IL}}(\pi_m) - \alpha\mathbb{E}_{\rho^{\pi_m}_{\widehat{\mathcal{M}}}}[U_\theta(s,a)], \qquad J_e = J_{\mathrm{IL}}(\pi_e) + \beta\mathbb{E}_{\rho^{\pi_e}_{\widehat{\mathcal{M}}}}[U_\theta(s,a)G^{\pi_m}_{k,H}(s')]. \tag{8}$$

The negative sign protects policy optimization; the positive, gated term decides which reachable uncertainty is worth resolving with real interaction.

4 RECON: Recoverability-Conditioned Exploration

RECON extends CMIL's DreamerV2-based latent adversarial-learning loop (Kolev et al., 2024; Hafner et al., 2020) with a separate recoverability-conditioned explorer, while sharing the representation, world model, discriminator, and replay. (Figure 2).

Latent world model. We use DreamerV2 with the continuous ensemble RSSM configuration of CMIL. Let $s_t = (h_t, z_t)$ denote its deterministic and Gaussian stochastic state,

$$h_t = f_\theta(h_{t-1}, z_{t-1}, a_{t-1}), \qquad z_t \sim q_\theta(z_t \mid h_t, o_t), \quad \widehat{z}_t \sim p_\theta(z_t \mid h_t), \quad o_t \sim p_\theta(o_t \mid s_t). \tag{9}$$

Posterior trajectories from expert and online replay train the model with a reconstruction–KL objective (using KL balancing and free nats in practice),

$$\mathcal{L}_{\mathrm{wm}}(\theta) = \mathbb{E}_{\tau\sim\mathcal{D}_E\cup\mathcal{D}}\sum_t \left[-\log p_\theta(o_t \mid s_t) + c_{\mathrm{KL}} D_{\mathrm{KL}}(q_\theta(z_t \mid h_t, o_t)\|p_\theta(z_t \mid h_t))\right]. \tag{10}$$

Policy rollouts instead use the recurrent prior. A bootstrap transition ensemble (Chua et al., 2018; Lakshminarayanan et al., 2017) predicts means $\mu_k(s_t, a_t)$, giving the epistemic score

$$U_\theta(s_t, a_t) = \frac{1}{d_z}\sum_{j=1}^{d_z} \mathrm{Std}_{k=1:K}[\mu_{k,j}(s_t, a_t)]. \tag{11}$$

Adversarial imitation in imagination. Starting from replay posteriors, policy $i \in \{main, explore\}$ generates $\widehat{a}_t^i \sim \pi_i(\cdot \mid \widehat{s}_t^i)$ and $\widehat{s}_{t+1}^i \sim p_\theta(\cdot \mid \widehat{s}_t^i, \widehat{a}_t^i)$. As in CMIL, the discriminator contrasts expert posterior transitions with on-policy imagination:

$$\mathcal{L}_D(\psi) = -\mathbb{E}_{(s,a)\sim\rho_E} \log D_\psi(s,a) - \mathbb{E}_{(s,a)\sim\widehat{\rho}_{\pi_m}} \log(1 - D_\psi(s,a)). \tag{12}$$

The discriminator therefore remains the common imitation signal; uncertainty changes how each policy uses the learned model.

Dual policy objectives. Both policies use the same DreamerV2 latent actor–critic and expert-action regularizer with the shared optimization specified in Appendix B.1, but have separate actors and critic ensembles. The acquisition distinction is entirely expressed by their imagined rewards,

$$r_t^m = D_\psi(\widehat{s}_t^m, \widehat{a}_t^m) - \alpha U_\theta(\widehat{s}_t^m, \widehat{a}_t^m), \qquad r_t^e = D_\psi(\widehat{s}_t^e, \widehat{a}_t^e) + \beta U_\theta(\widehat{s}_t^e, \widehat{a}_t^e) G_T. \tag{13}$$

With $\alpha = \beta = 10$, π_m retains CMIL's pessimistic model-use objective, whereas π_e assigns optimistic value only to uncertainty conditioned on recoverability.

Recoverability-conditioned acquisition. To compute recoverability G_T, rather than learning a separate inverse or backward dynamics model, RECON exploits Dreamer's multi-step latent imagination to directly evaluate whether the current main policy can return from the explorer endpoint toward expert-compatible behavior. RECON takes the terminal state $\widehat{s}_T^e$ of an explorer imagination and rolls the frozen main policy forward deterministically for H_r steps,

$$\widetilde{a}_h = \operatorname{mode} \pi_m(\cdot \mid \widetilde{s}_h), \quad \widetilde{s}_{h+1} = p_\theta(\cdot \mid \widetilde{s}_h, \widetilde{a}_h), \quad G_T = \max_{h=H_r-L+1:H_r} \sigma\left(\frac{D_\psi(\widetilde{s}_h, \widetilde{a}_h) - \delta_k}{\tau}\right). \tag{14}$$

Here δ_k is the exponential moving average of the main-policy discriminator score, $\sigma(x)$ is the logistic sigmoid. The tail maximum asks whether the current main policy can regain main-compatible behavior, rather than whether the explorer endpoint merely looks expert-like. A single terminal-derived G_T is shared by all transitions of an imagined explorer rollout. During the explorer update, model, discriminator, and main-policy parameters are frozen, but the pathwise gradient through the recovery rollout is retained (detailed in Appendix A.5). We use $H_r = 5$, $L = 3$, and $\tau = 0.05$.

Real interaction. The two policies alternate *within* each real episode rather than using separate rollouts. Each episode is partitioned into 15 temporal strata, and one 10-step explorer window is sampled uniformly inside every stratum; the main policy acts at all remaining steps. Thus, a full 500-step task episode contains 150 explorer and 350 main-policy steps. All transitions enter the same replay, which updates the shared DreamerV2 model, discriminator, and both actor–critics. Evaluation removes the explorer and executes only the deterministic main policy. The complete collection and training loop is given in Appendix B.2, with shared hyperparameters in Appendix B.3.

5 Experiments

We test the acquisition claim through five questions: **(a)** Does RECON improve imitation at a fixed interaction budget? **(b)** Does targeted acquisition improve robustness beyond nominal training? **(c)** Where does G redirect interaction? **(d)** Is recoverability necessary, and which design choices matter? **(e)** Do the resulting data improve recovery-relevant dynamics?

Protocol and baselines. We conduct experiments across continuous control, navigation, and manipulation tasks, including DMControl Hopper Stand and Walker Run, U-Maze, PointMaze, Meta-World Drawer Open, Faucet Close, Handle Press and Hammer. Rewards and success labels are used only for evaluation. All methods share the same expert demonstrations, interaction budgets, and 3 seeds. We compare with BC, which directly fits expert state–action pairs; data-augmented replay-based DA-DAC (Kostrikov et al., 2018), which performs model-free adversarial imitation with off-policy data reuse; V-MAIL (Rafailov et al., 2021), which performs adversarial imitation on latent world-model rollouts; CMIL (Kolev et al., 2024), which regularizes imagined policy optimization with epistemic uncertainty; and IQ-MPC (Li et al., 2025), which combines reward-free world-model learning with inverse soft-Q learning and latent model-predictive control. RECON and CMIL use the same deployed policy objective and differ in online data acquisition. Curves show aggregate performance with seed dispersion; complete task details are in Appendix B.4 and baseline implementations in Appendix B.7.

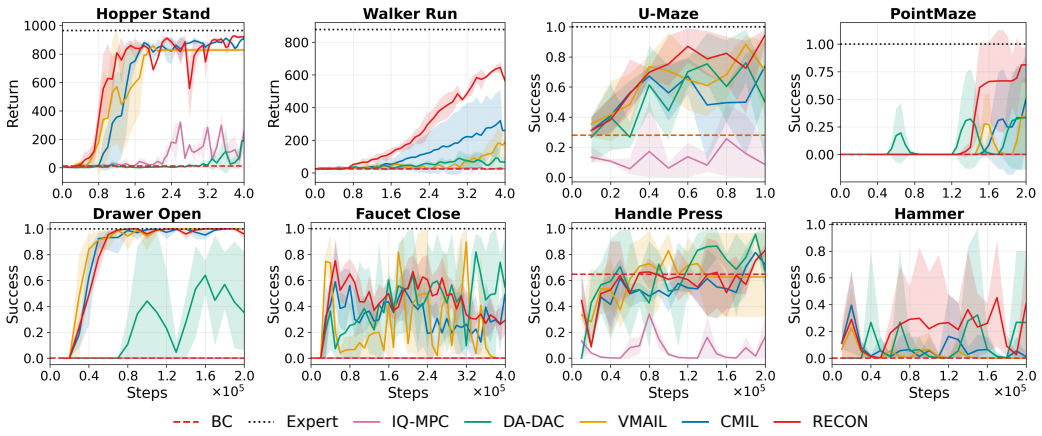

Figure 3: **Main experiments.** Imitation learning across eight environments under matched demonstrations and real-interaction budgets. RECON changes training-time acquisition; only its conservative main policy is evaluated.

(a) Does targeted acquisition improve imitation? Across all eight environments in Figure 3, RECON achieves either the best or competitive performance. The gains are particularly pronounced on Walker Run, PointMaze, and Hammer, where effective imitation requires recovering from deviations beyond the nominal expert trajectories. RECON also outperforms the most closely related CMIL baseline on Faucet Close and Handle Press, since RECON and CMIL use the same deployed actor and conservative policy objective, their comparison isolates the effect of model acquisition from conservative model use itself.

(b) Does targeted acquisition improve robustness beyond nominal training? Table 1 evaluates the main policy under action noise, delay, impulses, and dynamics shifts unseen during training (definitions in Appendix B.5). RECON improves perturbation performance across the evaluated tasks, with especially large gains under delay and altered dynamics. The results indicate that targeted acquisition improves the world model beyond the nominal demonstration distribution by collecting transitions around recoverable deviations, expanding the trust region around expert trajectories.

Table 1: Robust deployment of the main policy, reported as CMIL/RECON means over three seeds (absolute Δ). DMControl entries and deltas are returns; Maze and Meta-World entries are success percentages and deltas are percentage points (pp). Higher is better.

Environment	Clean	Noise	Delay	Impulse/scale
Hopper	905.8/**909.1**(+3.3)	862.9/**882.2**(+19.3)	408.5/**596.2**(+187.7)	864.0/**916.0**(+52.0)
Walker	421.2/**593.9**(+172.7)	401.6/**572.8**(+171.2)	133.0/**184.9**(+51.9)	378.6/**514.6**(+136.0)
Drawer	**100.0**/**100.0**(0.0 pp)	**100.0**/96.7(-3.3 pp)	95.0/**96.7**(+1.7 pp)	**100.0**/92.6(-7.4 pp)
Faucet	30.0/**70.0**(+40.0 pp)	15.0/**75.0**(+60.0 pp)	5.0/**85.0**(+80.0 pp)	15.6/**66.1**(+50.5 pp)
Handle	**75.0**/70.0(-5.0 pp)	67.5/**77.5**(+10.0 pp)	**80.0**/**80.0**(0.0 pp)	77.8/**86.1**(+8.3 pp)
Hammer	0.0/**2.5**(+2.5 pp)	**8.3**/6.3(-2.1 pp)	8.3/**26.3**(+17.9 pp)	**10.0**/3.9(-6.1 pp)
U-Maze	40.0/**45.0**(+5.0 pp)	31.7/**38.3**(+6.7 pp)	36.7/**44.2**(+7.5 pp)	32.5/**38.3**(+5.8 pp)
PointMaze	41.3/**100.0**(+58.7 pp)	42.0/**99.0**(+57.0 pp)	30.3/**100.0**(+69.7 pp)	16.0/**47.0**(+31.0 pp)

(c) Where does G redirect interaction? Figure 4 shows that around the lower turn, CMIL's conservative acquisition provides limited coverage of recovery dynamics near the expert trajectory, so inertia-induced deviations can drive the policy into the wall, resulting in noticeably higher visitation density along the boundary than RECON. To further characterize acquisition allocation, we partition explorer states into expert-support, near-support, and far-OOD regions by distance to the expert state cloud. RECON shifts interaction toward near-support states and away from far-OOD novelty. Consistent with the acquisition blind spot in Theorem 1, this indicates that recoverability conditioning improves coverage of recovery-relevant dynamics that nominal conservative collection can leave underexplored.

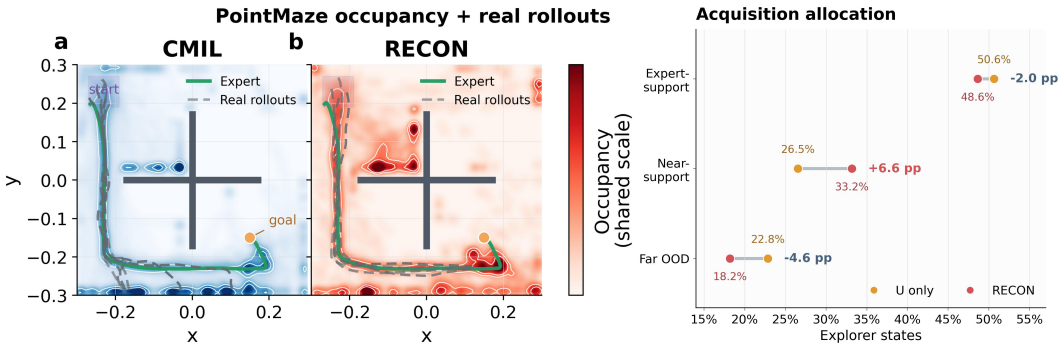

Figure 4: **PointMaze visitation and acquisition. Left:** State-visitation frequency of CMIL and RECON under the same training budget, with expert trajectories and representative real rollouts overlaid. **Right:** Acquisition allocation across expert-support, near-support, and far-OOD regions, defined by distance to the expert state cloud.

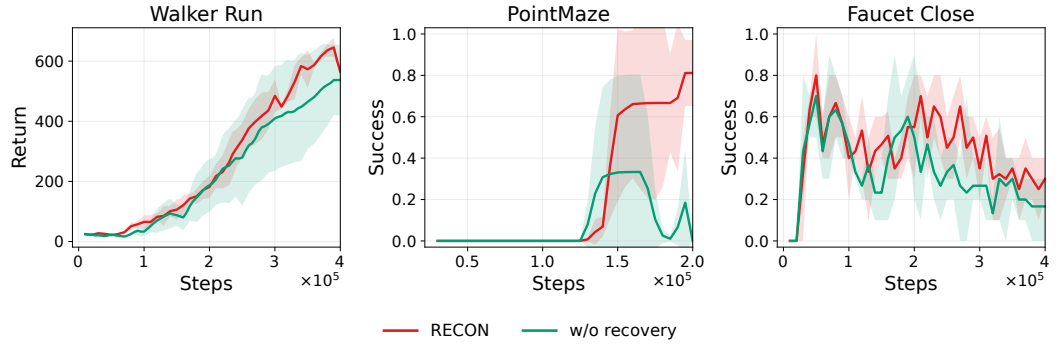

Figure 5: **Recoverability ablation.** "w/o recovery" removes the recoverability term and uses $r_t^e = D_\psi(s_t, a_t) + \beta U_\theta(s_t, a_t)$. Because recoverability gating rescales the effective uncertainty bonus in RECON, we use $\beta = 5$ for the ungated variant for a fair comparison.

(d) Is recoverability necessary, and which choices matter? Figure 5 isolates the role of recoverability by removing G while retaining the explorer's uncertainty bonus. The resulting uncertainty-only collector can preserve early learning, but degrades later performance, most clearly on Walker Run and PointMaze. Consistent with Theorem 2 3, uncertainty alone identifies under-specified dynamics but not which uncertainty is useful for imitation; Appendix C.1 Figure 9 (b) further shows that recoverability-gated acquisition yields the largest uncertainty reduction.

Figure 6 further separates the acquisition signal from the remaining design choices. Random collection improves generic model coverage but spends too little interaction on task-relevant deviations, while a terminal-only signal captures expert compatibility without modeling whether the main policy can actually recover through the dynamics. RECON instead evaluates recoverability through multi-step main-policy imagination. With 10-step collection bursts, the default 15-window schedule (totally 50-window) balances exploration coverage and main-policy control. Retaining pathwise gradients through the recovery rollout clearly outperforms stop-gradient, while the choice of tail aggregation has comparatively little effect.

(e) Does targeted acquisition improve the relevant model? As summarized in Table 2, RECON generally reduces world-model prediction error and ensemble uncertainty across the evaluated tasks. Replay error may increase because RECON's replay contains more exploratory trajectories, as in PointMaze. Figure 7 (left) shows lower held-out error across distance-based coverage regions. Figure 7 (right) further validates the acquisition signal itself: discriminator-based recoverability is strongly correlated with empirical recoverability measured from real rollouts, and this correspondence is markedly stronger for the policy trained with RECON than for CMIL.

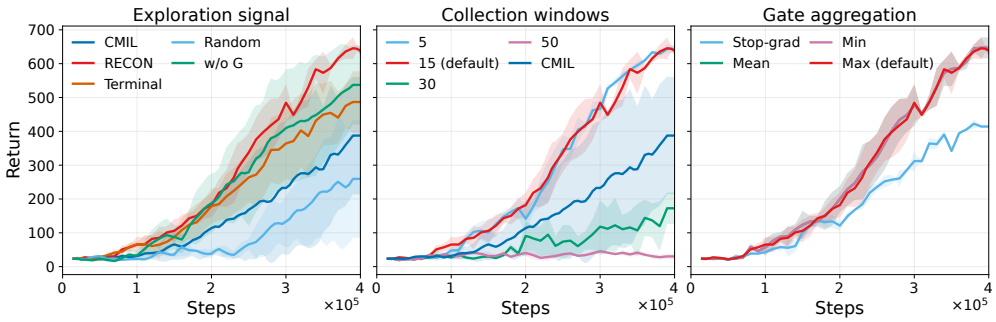

Figure 6: **Design ablation. Left:** alternative exploration signals. **Middle:** temporal placement of fixed-length collection bursts. **Right:** gate aggregation and the stop-gradient control.

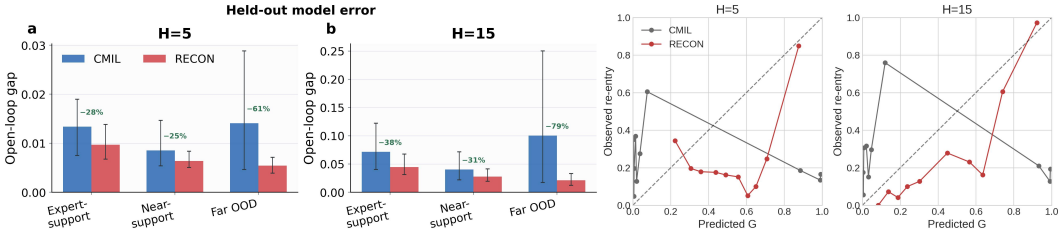

Figure 7: **Diagnostics of recoverability conditioning. Left:** Held-out discounted latent open-loop prediction gap across different regions. **Right:** Predicted recoverability G_H (rollout horizon $H \in \{5, 15\}$) versus empirical expert-tube re-entry.

Table 2: World-model diagnostics, reported as CMIL/RECON (relative change). Gaps are latent open-loop prediction errors defined in Appendix B.6. Lower is better; uncertainty is in $\times 10^{-3}$.

Environment	Expert H15 gap	Replay H5 gap	Uncertainty
Hopper	1.229/**0.979**(-20.3%)	0.397/**0.272**(-31.5%)	2.8/2.8(0%)
Walker	0.608/**0.507**(-16.6%)	0.250/**0.214**(-14.4%)	8.3/**8.0**(-3.6%)
Drawer	1.537/**1.219**(-20.7%)	0.499/**0.394**(-21.0%)	4.0/**3.3**(-17.5%)
Faucet	0.774/**0.720**(-7.0%)	0.278/**0.272**(-2.2%)	**3.1**/3.2(+3.2%)
Handle	1.072/**0.817**(-23.8%)	0.401/**0.330**(-17.7%)	3.1/**2.8**(-9.7%)
Hammer	**0.933**/0.934(+0.2%)	0.349/**0.343**(-1.6%)	3.2/**2.9**(-8.3%)
U-Maze	**7.049**/7.060(+0.2%)	**2.450**/2.471(+0.9%)	8.1/**7.7**(-5.3%)
PointMaze	0.0535/**0.0331**(-38.1%)	**0.0056**/0.0070(+25.0%)	11.9/**6.5**(-45.4%)

6 Discussion and Limitations

RECON improves the world model selectively rather than uniformly: targeted acquisition need only improve dynamics relevant to recovery from deviations around expert behavior. A current limitation is the additional model parameters and training computation introduced by the separate explorer. Future work may improve the two signals underlying acquisition by developing better calibrated epistemic uncertainty estimates and more reliable recoverability estimates beyond finite-horizon world-model imagination and discriminator scores, especially farther from current data support.

7 Conclusion

Conservative model use determines how an imperfect world model should guide policy optimization, but not where limited interaction should be spent to improve that model. RECON addresses this complementary acquisition problem by combining a conservative main policy with recoverability-conditioned exploration. Across control, navigation, and manipulation, this improves imitation, robustness and model accuracy, opening a complementary direction to conservative policy optimization: learning not only how to use an imperfect world model, but also where to improve it.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish writing, including improving wording, grammar, and clarity, and to assist with literature retrieval and discovery, such as identifying potentially relevant related work. All AI-assisted content was reviewed by the authors, and literature identified with AI assistance was checked against the corresponding original sources before inclusion. We take responsibility for the final content of this work, including all text, claims, citations, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Complete proofs of the theoretical results are provided in Appendix A. Appendix B specifies the training algorithm, model and policy hyperparameters, environments and observations, robustness perturbation protocol, world-model evaluation, and baseline implementations. All baselines are evaluated using their official implementations and the corresponding recommended hyperparameter settings. Additional diagnostic experiments are reported in Appendix C. The complete implementation of RECON, together with the code required to reproduce the reported experiments, will be released publicly upon acceptance.

REFERENCES

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 1, 2004.

Lars Ankile, Anthony Simeonov, Idan Shenfeld, and Pulkit Agrawal. Juicer: Data-efficient imitation learning for robotic assembly. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5096–5103. IEEE, 2024.

Marina Y Aoyama, Joao Moura, Juan Del Aguila Ferrandis, and Sethu Vijayakumar. Poke and strike: Learning task-informed exploration policies. *arXiv preprint arXiv:2509.00178*, 2025.

Nir Baram, Oron Anschel, and Shie Mannor. Model-based adversarial imitation learning. *arXiv preprint arXiv:1612.02179*, 2016.

Jiaqi Chen, Ji Shi, Cansu Sancaktar, Jonas Frey, and Georg Martius. Offline vs. online learning in model-based rl: Lessons for data collection strategies. *arXiv preprint arXiv:2509.05735*, 2025.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *Advances in neural information processing systems*, 31, 2018.

Branton DeMoss, Paul Duckworth, Jakob Foerster, Nick Hawes, and Ingmar Posner. Ditto: Offline imitation learning with world models. *arXiv preprint arXiv:2302.03086*, 2023.

Gabriel Dulac-Arnold, Daniel Mankowitz, and Todd Hester. Challenges of real-world reinforcement learning. *arXiv preprint arXiv:1904.12901*, 2019.

Peter Englert, Alexandros Paraschos, Jan Peters, and Marc Peter Deisenroth. Model-based imitation learning by probabilistic trajectory matching. In *2013 IEEE international conference on robotics and automation*, pp. 1922–1927. IEEE, 2013.

Chelsea Finn, Sergey Levine, and Pieter Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. In *International conference on machine learning*, pp. 49–58. PMLR, 2016.

Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*, 2017.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019b.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Nick Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, volume 2024, pp. 47376–47405, 2024.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *arXiv preprint arXiv:2203.04955*, 2022.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29, 2016.

Anthony Hu, Gianluca Corrado, Nicolas Griffiths, Zachary Murez, Corina Gurau, Hudson Yeo, Alex Kendall, Roberto Cipolla, and Jamie Shotton. Model-based imitation learning for urban driving. *Advances in Neural Information Processing Systems*, 35:20703–20716, 2022.

Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019.

Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. Morel: Model-based offline reinforcement learning. *Advances in neural information processing systems*, 33: 21810–21823, 2020.

Rahul Kidambi, Jonathan Chang, and Wen Sun. Mobile: Model-based imitation learning from observation alone. *Advances in Neural Information Processing Systems*, 34:28598–28611, 2021.

Victor Kolev, Rafael Rafailov, Kyle Hatch, Jiajun Wu, and Chelsea Finn. Efficient imitation learning with conservative world models. In *6th Annual Learning for Dynamics & Control Conference*, pp. 1777–1790. PMLR, 2024.

Ilya Kostrikov, Kumar Krishna Agrawal, Debidatta Dwibedi, Sergey Levine, and Jonathan Tompson. Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning. *arXiv preprint arXiv:1809.02925*, 2018.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Michael Laskey, Jonathan Lee, Roy Fox, Anca Dragan, and Ken Goldberg. Dart: Noise injection for robust imitation learning. In *Conference on robot learning*, pp. 143–156. PMLR, 2017.

Shangzhe Li, Zhiao Huang, and Hao Su. Reward-free world models for online imitation learning. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 34702–34724, 2025.

Yecheng Jason Ma, Kausik Sivakumar, Jason Yan, Osbert Bastani, and Dinesh Jayaraman. Learning policy-aware models for model-based reinforcement learning via transition occupancy matching. In *Learning for Dynamics and Control Conference*, pp. 259–271. PMLR, 2023.

Max Sobol Mark, Archit Sharma, Fahim Tajwar, Rafael Rafailov, Sergey Levine, and Chelsea Finn. Offline retraining for online rl: Decoupled policy learning to mitigate exploration bias. *arXiv preprint arXiv:2310.08558*, 2023.

Jung Yeon Park and Lawson Wong. Robust imitation of a few demonstrations with a backwards model. *Advances in Neural Information Processing Systems*, 35:19759–19772, 2022.

Rafael Rafailov, Tianhe Yu, Aravind Rajeswaran, and Chelsea Finn. Visual adversarial imitation learning using variational models. *Advances in Neural Information Processing Systems*, 34:3016–3028, 2021.

Juntao Ren, Gokul Swamy, Zhiwei Steven Wu, J Andrew Bagnell, and Sanjiban Choudhury. Hybrid inverse reinforcement learning. *arXiv preprint arXiv:2402.08848*, 2024.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *International conference on machine learning*, pp. 8583–8592. PMLR, 2020.

Seokin Seo, Byung-Jun Lee, Jongmin Lee, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. Mitigating covariate shift in behavioral cloning via robust stationary distribution correction. *Advances in Neural Information Processing Systems*, 37:109177–109201, 2024.

Jie-Jing Shao, Hao-Sen Shi, Lan-Zhe Guo, and Yu-Feng Li. Offline imitation learning with model-based reverse augmentation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2608–2617, 2024.

Pranav Shyam, Wojciech Jaśkowski, and Faustino Gomez. Model-based active exploration. In *International conference on machine learning*, pp. 5779–5788. PMLR, 2019.

Shenghua Wan, Yucen Wang, Minghao Shao, Ruying Chen, and De-Chuan Zhan. Semail: eliminating distractors in visual imitation via separated models. In *International Conference on Machine Learning*, pp. 35426–35443. PMLR, 2023.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in neural information processing systems*, 33:14129–14142, 2020.

Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. Combo: Conservative offline model-based policy optimization. *Advances in neural information processing systems*, 34:28954–28967, 2021.

Xingyuan Zhang, Philip Becker-Ehmck, Patrick van der Smagt, and Maximilian Karl. Action inference by maximising evidence: Zero-shot imitation from observation with world models. *Advances in Neural Information Processing Systems*, 36:46284–46303, 2023.

A Detailed Proofs

A.1 Notation and regularity conditions

Write $x = (s, a)$ and let $d^{\pi}_{M,q}$ denote the normalized discounted state–action occupancy of policy π under transition kernel M and initial distribution q. Values are unnormalized discounted returns. We assume $0 < \gamma < 1$, a common measurable reward with $|r| \le R_{\max}$, and Markov kernels on standard Borel spaces, so the conditional distributions below exist. All equalities and inequalities between densities are understood almost everywhere with respect to the indicated reference measure.

For the Bayesian statements, $p_k(\mathrm{d}\omega) = p(\mathrm{d}\omega \mid \mathcal{D}_k)$ is the current posterior and

$$\widehat{T}_k(\mathrm{d}y \mid x) = \int T_\omega(\mathrm{d}y \mid x) p_k(\mathrm{d}\omega) \tag{15}$$

is its posterior predictive kernel. A prospective Y_x is drawn by first drawing $\omega \sim p_k$ and then $Y_x \sim T_\omega(\cdot \mid x)$. Distinct prospective observations are conditionally independent given ω and their inputs. The round-k gate g_k and reference distribution ν are fixed while evaluating a query; this is essential because otherwise an additional term accounts for changing the acquisition target itself.

A.2 Proof of Theorem 1

We first record the expectation–TV inequality used throughout. Under the convention $D_{\mathrm{TV}}(P, Q) = \sup_A |P(A) - Q(A)|$, every bounded measurable f satisfies

$$|\mathbb{E}_P f - \mathbb{E}_Q f| \le 2\|f\|_\infty D_{\mathrm{TV}}(P, Q). \tag{16}$$

This follows from the dual representation $2D_{\mathrm{TV}}(P, Q) = \sup_{\|h\|_\infty \le 1} |\mathbb{E}_P h - \mathbb{E}_Q h|$.

For a fixed policy, define the state kernels

$$P^\pi(\mathrm{d}s' \mid s) = \int T(\mathrm{d}s' \mid s, a)\pi(\mathrm{d}a \mid s), \quad \widehat{P}^\pi(\mathrm{d}s' \mid s) = \int \widehat{T}_{\mathcal{D}}(\mathrm{d}s' \mid s, a)\pi(\mathrm{d}a \mid s). \tag{17}$$

Let $r^\pi(s) = \int r(s, a)\pi(\mathrm{d}a \mid s)$. The Bellman equations and the resolvent identity imply

$$V^\pi_T - V^\pi_{\widehat{T}} = (I - \gamma P^\pi)^{-1} r^\pi - (I - \gamma \widehat{P}^\pi)^{-1} r^\pi \tag{18}$$

$$= \gamma (I - \gamma P^\pi)^{-1} (P^\pi - \widehat{P}^\pi) V^\pi_{\widehat{T}}. \tag{19}$$

Because $\|V^\pi_{\widehat{T}}\|_\infty \le R_{\max}/(1 - \gamma)$, Equation 16 gives

$$\left|(P^\pi - \widehat{P}^\pi) V^\pi_{\widehat{T}}(s)\right| \le \int \pi(\mathrm{d}a \mid s) \left| \int V^\pi_{\widehat{T}}(s')[T - \widehat{T}_{\mathcal{D}}](\mathrm{d}s' \mid s, a) \right| \tag{20}$$

$$\le \frac{2R_{\max}}{1 - \gamma} \int e_{\mathcal{D}}(s, a)\pi(\mathrm{d}a \mid s). \tag{21}$$

Moreover, $q(I - \gamma P^\pi)^{-1} = \sum_{t \ge 0} \gamma^t q (P^\pi)^t$. Integrating Equation 19 against q, substituting Equation 21, and using the definition of normalized occupancy therefore yields

$$|V^\pi_{T,q} - V^\pi_{\widehat{T},q}| \le \frac{2\gamma R_{\max}}{1 - \gamma} \sum_{t \ge 0} \gamma^t \mathbb{E}[e_{\mathcal{D}}(s_t, a_t)] \tag{22}$$

$$= \frac{2\gamma R_{\max}}{(1 - \gamma)^2} \mathbb{E}_{d^\pi_{T,q}}[e_{\mathcal{D}}], \tag{23}$$

which proves Equation 2. Notice that the resolvent follows P^π, so the error is averaged under real visitation; using the learned-model occupancy here would require a separate change-of-measure argument.

For the imitation statement, insert the learned-model value of the learner:

$$V^{\pi_E}_T - V^\pi_T = V^{\pi_E}_T - V^\pi_{\widehat{T}} + V^\pi_{\widehat{T}} - V^\pi_T. \tag{24}$$

The second term is bounded by the result above. The normalized occupancy identity $V = (1-\gamma)^{-1}\mathbb{E}_d[r]$ and Equation 16 give

$$V_T^{\pi_E} - V_{\widehat{T}}^{\pi} \le \frac{2R_{\max}}{1-\gamma} D_{\mathrm{TV}}(d^{\pi_E}_{T,\mu_0}, d^{\pi}_{\widehat{T},\mu_0}). \tag{25}$$

Combining this inequality with Equation 24 establishes Equation 3.

It remains to justify the coverage claim. Let $d_0 = d^{\pi_m}_{T,\mu_0}$ and $d_q = d^{\pi_m}_{T,q}$. If $d_q \ll d_0$, then for the nonnegative error $e_{\mathcal{D}}$,

$$\varepsilon_q(\pi_m, \mathcal{D}) = \int e_{\mathcal{D}}(x) \frac{\mathrm{d}d_q}{\mathrm{d}d_0}(x)\, d_0(\mathrm{d}x) \tag{26}$$

$$\le \left\| \frac{\mathrm{d}d_q}{\mathrm{d}d_0} \right\|_\infty \varepsilon_{\mu_0}(\pi_m, \mathcal{D}). \tag{27}$$

Thus the comparison yields a finite uniform factor whenever $\|\mathrm{d}d_q/\mathrm{d}d_0\|_\infty < \infty$. If $d_q \ll d_0$ but this essential supremum is unbounded, the change-of-measure identity remains valid but does not provide a finite uniform multiplicative bound.

Conversely, if $d_q \not\ll d_0$, there is a measurable A with $d_0(A) = 0$ and $d_q(A) > 0$. The bounded error function $e = 1_A$ then has zero nominal expectation and positive recovery expectation. Equivalently, on any nondegenerate successor space one may choose kernels that agree on A^c and have disjoint successor laws on A. Hence no finite constant can control recovery error from nominal error uniformly over transition kernels.

A.3 Proof of Theorem 2

For compactness define

$$e_{\omega,k}(x) = D_{\mathrm{TV}}(T_\omega(\cdot \mid x), \widehat{T}_k(\cdot \mid x)), \qquad \bar{e}_k(x) = \int e_{\omega,k}(x) p_k(\mathrm{d}\omega). \tag{28}$$

Let $d_{\omega,q} = d^{\pi_m}_{T_\omega,q}$. By the assumed occupancy domination $d_{\omega,q}(x) \le C_q \nu(x)$ for posterior-almost every ω, nonnegativity and Tonelli's theorem give

$$\mathbb{E}_{p_k}[\varepsilon_q(\omega, \mathcal{D}_k)] = \int p_k(\mathrm{d}\omega) \int e_{\omega,k}(x) d_{\omega,q}(\mathrm{d}x) \tag{29}$$

$$\le C_q \int \nu(\mathrm{d}x) \int e_{\omega,k}(x) p_k(\mathrm{d}\omega) = C_q \mathbb{E}_\nu[\bar{e}_k]. \tag{30}$$

The identity $1 = g_k + (1-g_k)$ separates the last expectation without dropping either part:

$$\mathbb{E}_\nu[\bar{e}_k] = \mathbb{E}_\nu[g_k \bar{e}_k] + \mathbb{E}_\nu[(1-g_k)\bar{e}_k]. \tag{31}$$

Pinsker's inequality applied for each (x, ω), followed by Jensen's inequality for the concave square root, yields

$$\bar{e}_k(x) \le \int \sqrt{\tfrac{1}{2} D_{\mathrm{KL}}(T_\omega(\cdot \mid x) \| \widehat{T}_k(\cdot \mid x))}\, p_k(\mathrm{d}\omega) \tag{32}$$

$$\le \sqrt{\ell_{\mathcal{D}_k}(x)/2}. \tag{33}$$

Since $0 \le g_k \le 1$, write $g_k\sqrt{\ell} = \sqrt{g_k}\sqrt{g_k \ell}$ and apply Cauchy–Schwarz under ν:

$$\mathbb{E}_\nu[g_k \bar{e}_k] \le \frac{1}{\sqrt{2}} \mathbb{E}_\nu[\sqrt{g_k}\sqrt{g_k \ell_{\mathcal{D}_k}}] \tag{34}$$

$$\le \sqrt{\frac{\mathbb{E}_\nu[g_k]\mathbb{E}_\nu[g_k \ell_{\mathcal{D}_k}]}{2}} = \sqrt{Z_k \mathcal{R}_k(\mathcal{D}_k)/2}. \tag{35}$$

Substituting Equations 31 and 35 into Equation 30 proves the theorem. The result is also valid when $Z_k = 0$, in which case the weighted-risk term vanishes. The residual term is unavoidable unless the gate upper-bounds recovery occupancy everywhere; retaining it makes explicit that the theorem motivates targeted, not global, model accuracy.

A.4 Proof of Theorem 3

Fix inputs x and z. From Equation 15, the conditional mutual-information identity for a mixture distribution is

$$I(\omega; Y_z \mid z, \mathcal{D}_k) = \int p_k(\mathrm{d}\omega) D_{\mathrm{KL}}\Big(T_\omega(\cdot \mid z) \| \widehat{T}_k(\cdot \mid z)\Big) \tag{36}$$

$$= \ell_{\mathcal{D}_k}(z). \tag{37}$$

After observing Y_x, Bayes' rule replaces p_k by $p(\omega \mid \mathcal{D}_k, x, Y_x)$. Averaging the resulting predictive KL over the as-yet-unobserved Y_x gives

$$\mathbb{E}_{Y_x \mid x, \mathcal{D}_k}[\ell_{\mathcal{D}'_{k,x}}(z)] = I(\omega; Y_z \mid z, x, Y_x, \mathcal{D}_k). \tag{38}$$

Let $C = (z, x, \mathcal{D}_k)$. Expanding the same mutual information in two orders gives

$$I(\omega, Y_x; Y_z \mid C) = I(Y_x; Y_z \mid C) + I(\omega; Y_z \mid Y_x, C), \tag{39}$$

$$I(\omega, Y_x; Y_z \mid C) = I(\omega; Y_z \mid C) + I(Y_x; Y_z \mid \omega, C). \tag{40}$$

The last term is zero by conditional independence of prospective observations. Consequently,

$$I(\omega; Y_z \mid z, \mathcal{D}_k) - I(\omega; Y_z \mid z, x, Y_x, \mathcal{D}_k) = I(Y_z; Y_x \mid z, x, \mathcal{D}_k). \tag{41}$$

Multiplying by the fixed, nonnegative $g_k(z)$, integrating with respect to $z \sim \nu$, and using Tonelli's theorem with Equations 37–41 proves Equation 7.

We finally make the cellwise specialization precise. Suppose the input space is a discrete partition into cells, the posterior factorizes over cell parameters $\omega = (\omega_x)_x$, and an observation at x depends only on ω_x. Then $I(Y_z; Y_x \mid z, x, \mathcal{D}_k) = 0$ for $z \neq x$, while for $z = x$ a fresh replicate Y'_x gives

$$\mathcal{R}_k(\mathcal{D}_k) - \mathbb{E}[\mathcal{R}_k(\mathcal{D}'_{k,x})] = \nu(x) g_k(x) I(Y'_x; Y_x \mid x, \mathcal{D}_k). \tag{42}$$

For stochastic transitions, the Markov chain $Y_x - \omega_x - Y'_x$ implies that this predictive information is at most $I(\omega_x; Y_x \mid x, \mathcal{D}_k)$. If successors are deterministic given ω_x, then $Y_x = Y'_x$ almost surely and both quantities equal $H(Y_x \mid x, \mathcal{D}_k)$, yielding the stated $\Gamma_k(x)$. Thus generic ensemble disagreement is a tractable proxy for the exact gain, not an equality claimed for arbitrary stochastic neural dynamics.

A.5 Latent-model specialization and pathwise gate gradient

The analysis requires only a gate in $[0, 1]$ held fixed during one acquisition round; it does not rely on the discounted-average aggregation in Equation 4. The implemented tail maximum in Equation 14 is therefore a valid instantiation of the same weighting principle.

Let ϕ parameterize a reparameterized explorer rollout $\widehat{s}_{0:T}(\phi), \widehat{a}_{0:T-1}(\phi)$ and set $U_t(\phi) = U_\theta(\widehat{s}_t, \widehat{a}_t)$. Starting from its endpoint $\widetilde{s}_0 = \widehat{s}_T(\phi)$, the frozen main policy and world model produce a deterministic recovery rollout and gate $G(\widehat{s}_T(\phi))$. For fixed nonnegative return weights w_t, the acquisition component is

$$A(\phi) = \mathbb{E}_\epsilon \left[\sum_{t=0}^{T-1} w_t U_t(\phi) G(\widehat{s}_T(\phi)) \right], \tag{43}$$

where ϵ collects actor and latent reparameterization noise. Under the usual dominated-differentiation condition, the pathwise gradient is

$$\nabla_\phi A = \mathbb{E}_\epsilon \sum_{t=0}^{T-1} w_t \left[G(\widehat{s}_T) \nabla_\phi U_t + U_t \nabla_{\widetilde{s}_0} G(\widetilde{s}_0) \frac{\partial \widehat{s}_T}{\partial \phi} \right]. \tag{44}$$

The first term moves the explorer toward epistemically informative transitions. The second differentiates through the subsequent main-policy rollout and moves the explorer endpoint toward states from which that fixed policy produces a large recovery score. Freezing model, discriminator, and main-policy *parameters* sets their parameter gradients to zero but preserves their input Jacobians, which is exactly what the second term requires.

For the implemented $G = \max_{h \in \mathcal{I}} p_h$, let h^* be its unique maximizer. Away from the measure-zero set of ties,

$$\nabla_{\widetilde{s}_0} G = \frac{p_{h^*}(1 - p_{h^*})}{\tau} \nabla_{\widetilde{s}_0} D_\psi(\widetilde{s}_{h^*}, \widetilde{a}_{h^*}), \tag{45}$$

where the final derivative includes all recurrent model and main-actor Jacobians along the h^*-step recovery rollout. At a tie, automatic differentiation selects a valid max subgradient. Applying stop-gradient to G removes the second term of Equation 44 while leaving the uncertainty-gradient term intact, matching the control in Figure 6. Our experiments use the continuous Gaussian RSSM configuration, so its reparameterized samples preserve this pathwise gradient.

Finally, suppose the latent belief is sufficient, and learned models use the same observation kernel $K(\mathrm{d}o' \mid s')$. Their observable one-step laws are the compositions $T_S K$ and $\widehat{T}_S K$. The data-processing inequality for every f-divergence gives

$$D_f(T_S K \| \widehat{T}_S K) \le D_f(T_S \| \widehat{T}_S). \tag{46}$$

Thus controlling predictive discrepancy in belief space controls the induced observable discrepancy under this sufficiency assumption. It does not assert that ensemble standard deviation equals KL: the implemented U_θ is a monotone epistemic surrogate, exact only under additional ensemble likelihood assumptions.

B Implementation and Experimental Details

B.1 DreamerV2 backbone and actor–critic optimization

RECON uses the DreamerV2 training architecture with CMIL's continuous ensemble RSSM option. An image encoder supplies the posterior state $s_t = (h_t, z_t)$ in Equation 9; a recurrent prior predicts imagined states, and a decoder reconstructs observations. The implementation sets `discrete=false`: z_t is a reparameterized Gaussian rather than the categorical latent used in the original Atari configuration. Ten bootstrapped prior heads share the deterministic state and expose their predicted means for U_θ in Equation 11. Expert and online sequences train the shared representation and world model, while replay posteriors seed 15-step prior rollouts for behavior learning.

For policy $i \in \{main, explore\}$, let r_t^i be the corresponding reward in Equation 13. Main and explorer each have an actor and two critics; the actor uses their pointwise minimum $V_t^i = \min_{k \in \{1,2\}} Q_{i,k}(\widehat{s}_t^i, \widehat{a}_t^i)$ and the pathwise return

$$R_t^{\lambda,i} = r_t^i + \gamma_t \Big[(1 - \lambda) V_{t+1}^i + \lambda R_{t+1}^{\lambda,i} \Big], \qquad w_t = \prod_{j<t} \gamma_j. \tag{47}$$

The DreamerV2 actor objective is regularized by expert behavior cloning,

$$\begin{aligned} J_i &= \mathbb{E}_{\widehat{\rho}_{\pi_i}} \sum_t w_t \left[(1 - \lambda) V_t^i + \lambda R_t^{\lambda,i} + \eta_i \mathcal{H}(\pi_i(\cdot \mid \widehat{s}_t^i)) \right], \\ \mathcal{L}_{\pi_i} &= -J_i + \kappa_i \mathbb{E}_{(s,a) \sim \mathcal{D}_E} [-\log \pi_i(a \mid s)], \\ \mathcal{L}_{Q_i} &= -\frac{1}{2} \sum_{k=1}^{2} \mathbb{E}_{\widehat{\rho}_{\pi_i}} \sum_t w_t \log p_{Q_{i,k}} \Big(\mathrm{sg}[\overline{R}_t^{\lambda,i}] \mid \widehat{s}_t^i, \widehat{a}_t^i \Big). \end{aligned} \tag{48}$$

Here $\overline{R}^{\lambda,i}$ replaces V^i in Equation 47 by one uniformly sampled slow-target critic head. The main critics additionally fit one-step replay targets. Thus only the rewards and collected state distribution differ; the latent optimization machinery is shared.

B.2 Complete training algorithm

Algorithm 1 summarizes the complete RECON training loop. The main policy uses uncertainty conservatively, whereas the explorer uses recoverability-gated uncertainty to acquire real transitions for the shared world model.

Algorithm 1 RECON training and data acquisition

Require: Expert replay $\mathcal{D}_E$, interaction budget B
Ensure: Deterministic main policy π_m
Initialize world model p_θ, discriminator D_ψ, main policy π_m, explorer π_e, and online replay $\mathcal{D}$
Pretrain p_θ, π_m, π_e on $\mathcal{D}_E$
Collect B_0 random transitions into $\mathcal{D}$
while interaction budget is not exhausted **do**
  Sample stratified explorer windows $\mathcal{W}$
  for each environment step t **do**
    Select

$$\pi_t = \begin{cases} \pi_e, & t \in \mathcal{W}, \\ \pi_m, & \text{otherwise}, \end{cases}$$

    execute $a_t \sim \pi_t(\cdot \mid s_t)$, and store the transition in $\mathcal{D}$
    if an update is scheduled **then**
      Sample $\xi_E \sim \mathcal{D}_E$ and $\xi \sim \mathcal{D}$
      Update the shared world model using $\mathcal{L}_{\mathrm{wm}}(\xi_E \cup \xi)$
      Imagine $\widehat{\tau}_m$ with π_m and update D_ψ
      Update π_m using

$$r_t^m = D_\psi(\widehat{s}_t^m, \widehat{a}_t^m) - \alpha U_\theta(\widehat{s}_t^m, \widehat{a}_t^m)$$

      Imagine $\widehat{\tau}_e$ with π_e and set $\widetilde{s}_0 = \widehat{s}_H^e$
      Roll out the frozen main policy for H_r steps:

$$\widetilde{a}_h = \operatorname{mode} \pi_m(\cdot \mid \widetilde{s}_h), \qquad \widetilde{s}_{h+1} = p_\theta(\cdot \mid \widetilde{s}_h, \widetilde{a}_h)$$

      Compute recoverability

$$G_T = \max_{h=H_r-L+1:H_r} \sigma\left(\frac{D_\psi(\widetilde{s}_h, \widetilde{a}_h) - \delta_k}{\tau}\right)$$

      Update π_e using

$$r_t^e = D_\psi(\widehat{s}_t^e, \widehat{a}_t^e) + \beta U_\theta(\widehat{s}_t^e, \widehat{a}_t^e) G_T$$

    while freezing θ, ψ, and π_m
    end if
  end for
end while
return $\operatorname{mode} \pi_m$

B.3 Hyperparameters

Tables 3 reports the full configuration of RECON method; these values are shared across environments and seeds. Environment rewards and success labels were not used for training.

B.4 Environments and expert observations

Figure 8 shows representative frames used in our experiments. It records the actual camera, rendering, and preprocessing visible to every pixel-based method. DMC Hopper and Walker test locomotion and contact dynamics; the two mazes test long-horizon correction around bottlenecks; and the four Meta-World tasks test visually similar manipulation scenes with distinct contact objectives.

For PointMaze diagnostics, let $d_E(s) = \min_{s_E \in \mathcal{S}_E} \|p(s) - p(s_E)\|_2$, where $p(s)$ is the planar environment position and $\mathcal{S}_E$ denotes the expert state cloud. We define expert-support by $d_E \leq 0.025$, near-support by $0.025 < d_E \leq 0.10$, and far-OOD by $d_E > 0.10$. These labels are used only for diagnostics and are unavailable to training.

B.5 Robustness perturbation protocol

For every method–environment pair, we evaluate three random seeds under the same deployment configurations. These comprise **Clean:** Default condition; **Noise**: two Gaussian action-noise levels, $\sigma \in \{0.05, 0.10\}$; **Delay:** two action delays, $d \in \{1, 2\}$ environment steps; and **Impulse/scale:** nine dynamics shifts including signed single-step action impulses of magnitude 0.15 or 0.30 at $t = 50$,

Table 3: Shared RECON hyperparameters.

Component	Hyperparameter	Value
Input/replay	observation; replay capacity	64×64 RGB; 5×10^6 transitions
	sequence batch; sequence length	32; 50
	random seed transitions; update ratio	5,000; 2 updates per 5 actions
World model	RSSM deterministic/stochastic size	400 / 360 (Gaussian)
	transition ensemble; shared hidden size	10 heads; 400
	encoder/decoder CNN depth	48 / 48
	encoder kernels; decoder kernels	$(4, 4, 4, 4)$; $(5, 5, 6, 6)$
	KL scale; free nats; balance	1.0; 1.0; 0.8
	model optimizer	Adam, lr 3×10^{-4}, $\epsilon = 10^{-5}$
	weight decay; gradient clip	10^{-6}; 100
Behavior	actor/critic architecture	4 layers $\times$ 400 ELU; two critics
	actor/critic optimizer	Adam, lr 3×10^{-4}
	imagination horizon	15
	discount γ; lambda λ	0.99; 0.95
	entropy coefficient	10^{-4}
	target update period/fraction	100 / 1.0
	BC pretraining updates; main BC coefficient	100 / 10
Discriminator	architecture	2 layers $\times$ 200 ELU, binary output
	optimizer; input noise std.	Adam, lr 3×10^{-5}; 2.5
Main policy	uncertainty penalty α	10
Explorer	uncertainty coefficient β	10
	imitation coefficient; explorer BC coefficient	1; 10
	collection schedule	15 strata, one 10-step window per stratum
Recovery gate	rollout horizon H_r; tail length L	5; 3
	aggregation; temperature τ	maximum; 0.05
Optimization	behavior/model pretraining updates	1,000
	numerical precision; deterministic mode	FP32; enabled

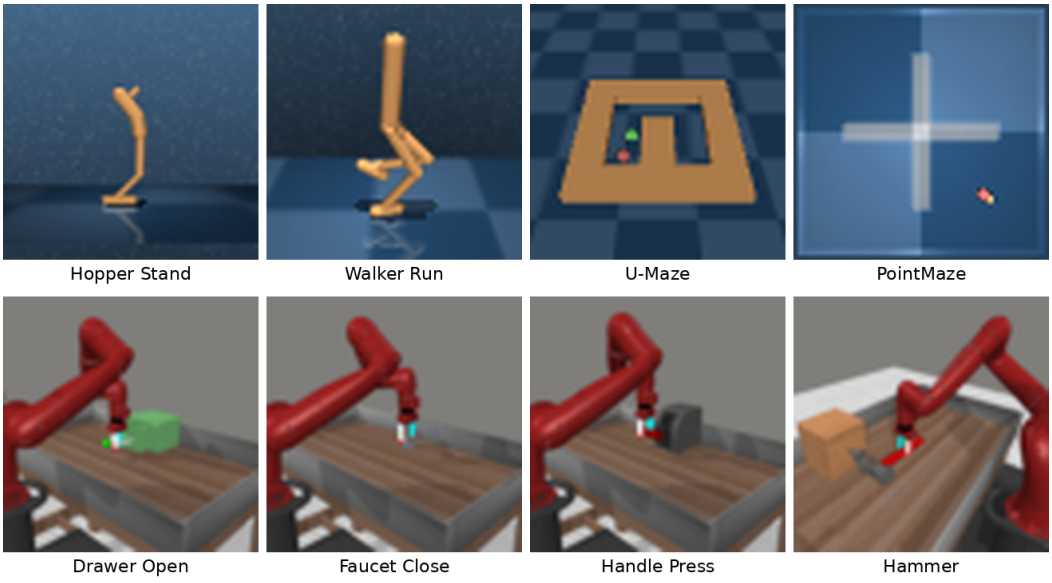

Figure 8: The eight evaluation environments. Policies receive the underlying 64×64 RGB observations.

component-wise action dropout with probability 0.10, action scaling by 0.85, mass scaling by 1.25 or 0.80, friction scaling by 1.50 or 0.70, and the joint mass/friction scale $(1.25, 1.50)$. All perturbed actions are clipped to the original action bounds.

Each configuration is evaluated independently. We first average episodes within a configuration, then configurations within each column of Table 1, and finally the three seeds. DMC tasks use episode return, while navigation and Meta-World tasks use success percentage. The single-step impulses test recovery from a localized deviation, action dropout and scaling test actuator mismatch, and mass/friction changes test persistent dynamics shift.

B.6 Open-loop model prediction gap.

We evaluate world-model accuracy in latent space rather than by pixel reconstruction error. For a held-out trajectory, let f_{t+h}^{post} denote the posterior latent feature inferred using the real observation at time $t+h$. Starting from the posterior state at time t, we roll the world model open-loop for H steps using only the recorded actions $a_t, \ldots, a_{t+H-1}$, without conditioning on subsequent observations, and denote the predicted latent feature at step $t+h$ by $\widehat{f}_{t+h}$. We define the discounted H-step open-loop prediction gap at starting point t as

$$\mathcal{E}_H(t) = \sum_{h=1}^{H} \gamma^{h-1} \frac{1}{d_f} \left\| \widehat{f}_{t+h} - f_{t+h}^{\text{post}} \right\|_2^2, \qquad \gamma = 0.99, \tag{49}$$

where d_f is the latent feature dimension. Reported errors average $\mathcal{E}_H(t)$ over the batch and all valid rollout starting points.

B.7 Baseline implementation details

All baselines use the same expert replay, image resolution, action bounds, environment wrappers, evaluation seeds, and per-task real-interaction budget. No baseline receives privileged state. All baselines are faithfully reproduced from their official implementations.

Behavior cloning (BC). BC uses the same image encoder, continuous actor, and expert batches as the model-based methods. The actor is optimized for 100 updates by maximum likelihood on expert actions and then frozen. The world model may continue to fit incoming images for matched logging, but neither those images nor environment rewards update the BC actor; thus online data do not alter its deployed behavior.

V-MAIL. V-MAIL learns a variational latent dynamics model and performs adversarial imitation using on-policy rollouts generated inside the learned model. The original implementation uses an earlier DreamerV1-style continuous RSSM; for a stronger and fair comparison, we replace it with the continuous DreamerV2 RSSM used by CMIL/RECON while preserving the V-MAIL imitation objective. We otherwise match the discriminator, replay batches, actor–critic, BC regularizer, and update ratio, set the uncertainty penalty to zero, and use a single actor for both collection and evaluation. This avoids confounding the comparison with differences in the world-model backbone.

CMIL. CMIL is the single-policy version of our backbone. Its imagined reward is $D_\psi(s,a) - 10U_\theta(s,a)$, and that same conservative actor collects all online transitions and is evaluated. It shares every entry in Table 3 except for the absent explorer and recovery gate. This is the closest comparison because RECON leaves CMIL's deployed objective unchanged.

DA-DAC. DA-DAC is implemented as pixel DrQ-SAC with a four-layer, 32-channel CNN, a 50-dimensional feature projection, three stacked frames, two 1024-unit actor and critic layers, twin critics, and automatic entropy tuning. It applies a four-pixel replication-pad random crop twice per image and uses the V-MAIL Appendix-C reward $\log D(s,a)$. Actor and critic learning rates are 10^{-3}, the temperature learning rate is 10^{-4}, batch size is 128, target $\tau = .01$, and actor and target networks update every two critic updates. Expert transitions are kept in a separate positive replay and are never inserted into the policy replay as discriminator negatives. DA-DAC has no world model, uncertainty term, or BC regularizer.

IQ-MPC. IQ-MPC is run with the inverse soft-Q objective and latent MPC. For a fair comparison with pixel inputs, it uses three stacked RGB frames and the TD-MPC2 encoder; we set model size 5 and otherwise retain the released optimizer and planner: batch size 256, learning rate 3×10^{-4},

latent size 512, five Q heads, planning horizon 3, six CEM iterations, 512 samples, 64 elites, and 24 policy trajectories. Only the step budget, evaluation frequency, and demonstration path are changed to match our protocol.

B.8 Compute, runtime, and memory overhead

RECON shares the encoder, RSSM ensemble, decoder, discriminator, and replay with CMIL. Its only persistent networks are one additional actor, two critics, and their targets; the five-step recovery rollout reuses the shared model and main actor. Table 4 reports an artifact-level audit of archived Walker checkpoints and the final 50 logged throughput records.

Table 4: Compute overhead relative to CMIL.

Quantity	CMIL	RECON	Difference
World-model parameters	17.38M	17.38M	0
Total stored parameters	21.52M	25.46M	+3.94M (+18.3%)
Explorer share of RECON parameters	—	15.5%	—
FP32 storage for added parameters	—	15.8 MB	+15.8 MB
Serialized checkpoint	83 MiB	98 MiB	+15 MiB
Logged FPS, last 50 records	2.541 ± 0.060	2.430 ± 0.148	-4.4%
Real interaction at evaluation	main actor	main actor	identical
Deployed policy class	main actor	main actor	identical

C Additional Experiments

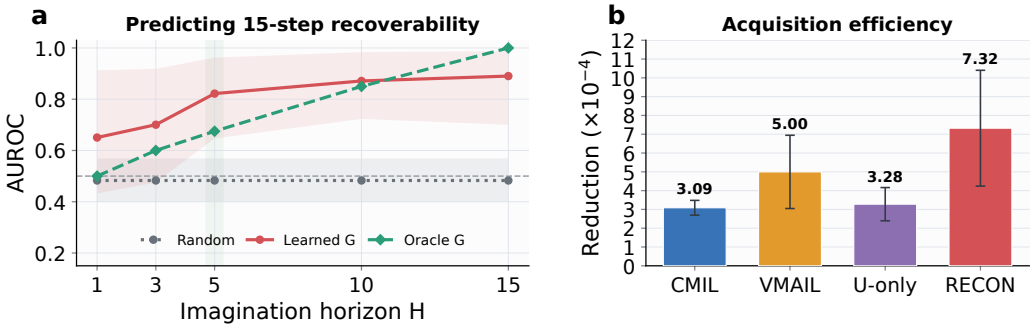

Figure 9: **Recoverability prediction and acquisition efficiency. (a)** AUROC between the predicted recoverability score and empirical recovery within 15 real-environment steps as the imagination horizon varies. **(b)** Reduction in ensemble uncertainty on fixed main-policy anchor transitions. The acquisition rewards are CMIL: $D_\psi - 10U_\theta$, V-MAIL: D_ψ, U-only: $D_\psi + 5U_\theta$, and RECON: $D_\psi + 10U_\theta G$.

C.1 Recoverability prediction across imagination horizons.

Figure 9(a) evaluates recoverability prediction on PointMaze. Empirical 15-step recoverability and the oracle predictor are defined using true environment-space distance to the expert trajectory, whereas the learned gate uses only discriminator scores along imagined main-policy rollouts. We vary the imagination horizon H and report AUROC against the empirical recovery labels. Longer lookahead improves the learned predictor and brings it closer to the oracle, supporting multi-step discriminator-based imagination as a practical estimate of recoverability.

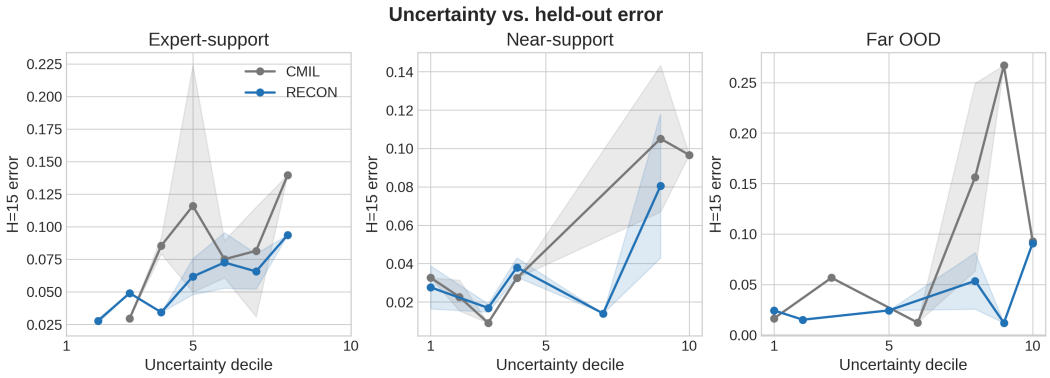

Figure 10: **Disagreement ranking and held-out model error.** $H = 15$ open-loop prediction error across within-run uncertainty deciles on common held-out PointMaze spatial bins.

C.2 Acquisition efficiency.

Figure 9(b) measures how efficiently online data reduce ensemble uncertainty on a fixed set of main-policy anchor transitions. For an anchor set A, we define the average ensemble uncertainty as

$$U_A(\theta) = \frac{1}{|A|} \sum_{(s,a)\in A} U_\theta(s,a), \qquad U_\theta(s,a) = \mathrm{MeanDim}\left[\mathrm{Std}_m\left(\mu_{\theta,m}(s,a)\right)\right]. \tag{50}$$

Keeping the anchor set fixed, the acquisition gain from model updates is

$$\Delta U_A = U_A(\theta_{\text{before}}) - U_A(\theta_{\text{after}}), \qquad R = \frac{\Delta U_A}{\Delta N_{\text{online}}}, \tag{51}$$

where ΔN_{online} denotes the effective number of online environment steps associated with the update interval.

RECON achieves the largest uncertainty reduction on the same anchor transitions. In contrast, the ungated uncertainty-only collector is less efficient than the reward-only V-MAIL collector, indicating that generic uncertainty seeking can spend interaction on novel dynamics that contribute little to reducing uncertainty around the main policy. Recoverability conditioning directs exploration toward uncertain transitions that more effectively improve the shared world model in regions relevant to deployment.

C.3 Disagreement and held-out model error.

In Figure 10, within expert-support regions, larger ensemble disagreement generally corresponds to larger held-out prediction error for RECON. The relationship is weaker and non-monotone in near-support and far-OOD regions, where long-horizon error additionally reflects compounding rollout error and sparse data support. These results support using U_θ as an acquisition ranking signal, without requiring it to provide a calibrated pointwise estimate of model error.