# ISI-WM: INTERVENTIONAL SELECTIVE INVARIANCE FOR WORLD MODELS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Visual world models enable prediction and control directly from pixels, but reliable deployment requires representations that remain stable under irrelevant appearance changes without losing action-dependent dynamics. Existing robustness objectives typically align augmented views without identifying which variation is a nuisance or when distinct actions produce outcomes that must remain separable. We address this gap with *Interventional Selective Invariance for World Models* (ISI-WM), a training framework that suppresses background-induced variation while preserving distinctions associated with divergent action outcomes. A background intervention pairs observations of the same physical state rendered against different backgrounds and aligns their latent encodings. An action intervention forks alternative actions from a shared state, trains the latent rollout toward its executed branch, and activates separation and ranking only when the realized simulator outcomes diverge. Together, these signals optimize the existing RGB encoder and latent dynamics without extending the underlying architecture; simulator access is confined to training, and deployment remains RGB-only. Under a unified dynamic-background protocol, ISI-WM achieves stronger overall generalization to unseen backgrounds than the evaluated baselines, while ablations and latent-response analyses support the complementary roles of the background and action interventions. Code is available at https://anonymous.4open.science/r/isi-wm-code-E0AF.

## 1 INTRODUCTION

A visual world model must compress pixel observations into a latent state that supports prediction and control. Dynamic backgrounds complicate this task: the same physical configuration can produce markedly different images, while different actions can lead to distinct physical futures. Preserving every pixel-level change spends capacity on nuisance motion, but imposing invariance indiscriminately can erase the action-dependent structure required for control. The appropriate objective is therefore selective invariance: stability to background variation together with sensitivity to action-induced dynamics.

Existing approaches to visual robustness use data augmentation, contrastive objectives, or domain randomization to broaden the training distribution or enforce agreement across views (Laskin et al., 2020a;b; Yarats et al., 2021b; Hansen & Wang, 2021; Tobin et al., 2017). These strategies can reduce dependence on appearance, but their supervision does not directly identify which visual changes are irrelevant to the task or determine when different actions produce distinguishable outcomes. The learner therefore lacks a unified criterion for deciding what its representation should ignore and what it must preserve.

We propose *Interventional Selective Invariance for World Models* (ISI-WM), which learns from two controlled training-time signals. Cross-background state pairs align representations of the same physical state, whereas outcome-gated action forks preserve latent differences only when alternative actions lead to distinguishable physical outcomes. We instantiate ISI-WM on TD-MPC2 (Hansen et al., 2024) because its explicit RGB encoder and action-conditioned latent dynamics naturally accommodate both signals. ISI-WM does not expand the underlying model architecture; privileged simulator information is confined to training, and deployment uses RGB observations alone. To-

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.

gether, these signals yield representations that are stable to background variation while remaining sensitive to action consequences.

Our contributions are:

- We introduce an intervention-based selective-invariance principle that uses controlled interventions to distinguish background changes that should be ignored from action consequences that must be preserved, thereby reconciling visual robustness with sensitivity to task dynamics.
- Building on this principle, we propose ISI-WM, which unifies cross-background state pairing and outcome-gated action forks in world-model training to learn latents that are stable to background changes yet sensitive to action outcomes. It does so without expanding the underlying architecture, and deployment remains RGB-only.
- We systematically evaluate ISI-WM under a unified protocol on six DMC tasks with dynamic video backgrounds. ISI-WM achieves stronger overall generalization to unseen backgrounds than the evaluated baselines, while controlled ablations show that the two intervention signals provide complementary supervision.

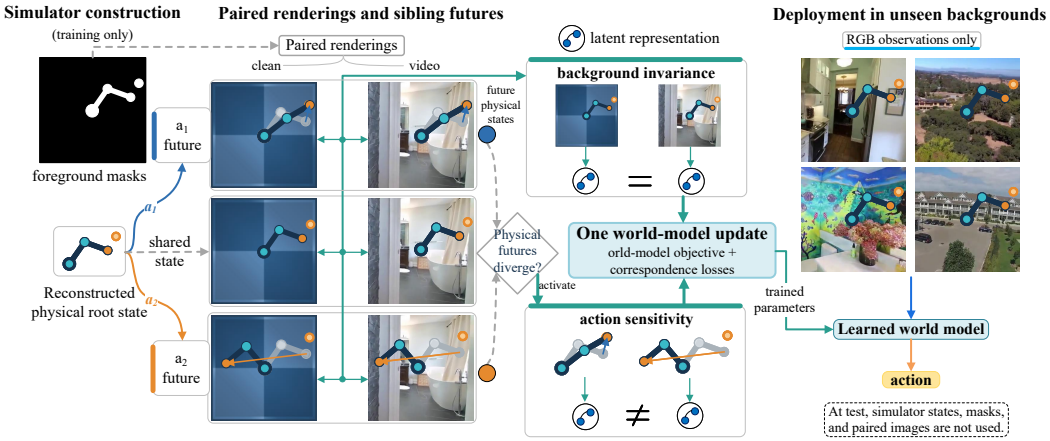


Figure 1: Overview of ISI-WM. Training begins from a shared simulator physical state, constructs clean/hard paired renderings, and executes sibling futures with a shared continuation. Background invariance is enforced on matched renders; action sensitivity is enabled by executed physical divergence. The auxiliary terms join the base world-model loss in one world-model update. Deployment receives RGB only.

## 2 Related Work

**Visual world models for control.** Learned dynamics models support trajectory optimization, policy learning, and search (Chua et al., 2018; Janner et al., 2019; Schrittwieser et al., 2020). Visual world models such as World Models, PlaNet, and the Dreamer family learn compact predictive states from pixels (Ha & Schmidhuber, 2018; Hafner et al., 2019; 2020; 2021; 2025). Representation-oriented variants change what these states retain or how they are structured (Deng et al., 2022; Chen et al., 2022; Kipf et al., 2020; Veerapaneni et al., 2020), while DayDreamer extends world-model learning to physical robots (Wu et al., 2023). TD-MPC and TD-MPC2 combine latent prediction with sampling-based online planning (Williams et al., 2016; Hansen et al., 2022; 2024). ISI-WM does not replace the base world-model architecture or planner; it changes the supervision applied to the visual encoder and latent dynamics.

**Visual robustness under distribution shift.** Pixel-based control has used reconstruction, contrastive learning, data augmentation, prototypical representations, domain randomization, and multi-view bottlenecks to shape visual representations (Yarats et al., 2021c; Laskin et al., 2020a;b; Yarats et al., 2021b; 2022; 2021a; Hansen & Wang, 2021; Tobin et al., 2017; Fan & Li, 2022). Augmentation and domain randomization broaden the observed appearance distribution (Laskin et al., 2020b; Yarats et al., 2021b; 2022; Hansen & Wang, 2021; Tobin et al., 2017), whereas CURL and DRIBO

suppress view-specific information (Laskin et al., 2020a; Fan & Li, 2022). Deployment-time adaptation provides a separate route to handling visual shifts (Hansen et al., 2021). The Distracting Control Suite standardizes evaluation under appearance shifts (Stone et al., 2021). World-model methods also isolate task-relevant information using reward, predictability, or action structure (Fu et al., 2021; Wang et al., 2022; Zhu et al., 2023; Ha et al., 2023; Wang et al., 2024). Classical MDP abstraction and bisimulation formalize state equivalence through reward and action-conditioned transition behavior (Givan et al., 2003; Ferns et al., 2004; Li et al., 2006; Ferns et al., 2011; Gelada et al., 2019; Zhang et al., 2021; Castro et al., 2021). They share ISI-WM's emphasis on control-relevant equivalence rather than pixel identity, but target a global abstraction; ISI-WM imposes local constraints from exact same-state renderings and realized branches, without estimating a bisimulation metric or claiming an abstraction guarantee. Invariant Causal Prediction and invariant risk minimization seek predictors stable across environments (Peters et al., 2016; Arjovsky et al., 2019), without specifying which action-induced changes a control representation must retain. HRSSM targets robust latent dynamics under visual changes, whereas R2-Dreamer uses redundancy reduction to learn without decoders or augmentation (Zbontar et al., 2021; Sun et al., 2024; Morihira et al., 2026).

**Privileged supervision and interventions.** Learning with privileged information and generalized distillation formalize side information available only during training (Vapnik & Vashist, 2009; Lopez-Paz et al., 2016); asymmetric learning and learning by cheating instantiate this pattern in control (Pinto et al., 2018; Chen et al., 2020). Causal inference and causal representation learning motivate intervention-based analysis (Pearl, 2009; Peters et al., 2017; Schölkopf et al., 2021; Ahuja et al., 2023), while theoretical and weakly supervised work studies how augmentations or paired interventions can isolate stable factors (von Kügelgen et al., 2021; Brehmer et al., 2022). Unsupervised disentanglement is itself underdetermined without inductive biases (Locatello et al., 2019). ISI-WM likewise confines privileged information to training, using it to construct exact cross-background correspondences and measure separation between executed action outcomes; evaluation still requires only RGB observations.

## 3 Interventional Selective Invariance for World Models

### 3.1 Problem formulation and intervention construction

Let $s$ denote the simulator-provided physical task state, $b$ a visual background, and $x = R(s, b)$ the resulting RGB observation. An encoder $f_\theta$ maps $x$ to $z \in \mathbb{R}^{D_z}$, and an action-conditioned latent dynamics model $d_\theta$ predicts its evolution. Our objective is selective invariance: the representation should be stable when $b$ changes at fixed $s$, while the learned dynamics should preserve distinctions supported by different action outcomes.

Here, an intervention denotes the controlled construction of paired and branched training examples, not the identification of causal effects.

We instantiate ISI-WM on RGB TD-MPC2 (Hansen et al., 2024). Its explicit RGB encoder and action-conditioned latent dynamics provide a direct interface for the two forms of supervision below. We leave the base architecture, planner, and original learning objectives in place; ISI-WM adds only the intervention objectives below.

The offline construction recreates the same physical root state $s_0$ in independent simulator runs by a fresh reset and deterministic replay of a common action prefix; it does not restore simulator snapshots. The root is rendered with the native background $b_c$ and a training-video background $b_v$:

$$x_0^c = R(s_0, b_c), \qquad x_0^v = R(s_0, b_v). \tag{1}$$

In the normalized action space $[-1, 1]^A$, we execute the alternative first actions

$$\mathcal{K} = \{0\} \cup \{+me_i, -me_i\}_{i=1}^{A}, \qquad K = 1 + 2A, \tag{2}$$

where $m$ is the intervention magnitude and $e_i$ is the $i$th coordinate basis vector. Each fork spans $H$ rollout steps, with every branch receiving the same open-loop continuation after its first action. This shared continuation isolates the first action as the source of differences between branches. At each executed rollout step, we retain the task-state vector and render its RGB stack with both backgrounds. The learner samples an ordered pair of distinct branches per root: $p$ is the branch to be

predicted and $n$ is its sibling comparison, not an action sign. Simulator-provided task states are used to derive the physical gap and gate, then stripped when the auxiliary capsule is materialized. The model forward path receives only RGB stacks and action sequences; derived labels are read by the auxiliary objectives. The resulting interventions therefore provide two relations: matched physical states across backgrounds, and alternative actions whose realized outcomes can be compared.

### 3.2 Cross-background state alignment

Define the per-coordinate squared latent error

$$\mathrm{rms}^2(u, v) = \frac{1}{D_z}\|u - v\|_2^2. \tag{3}$$

For an auxiliary minibatch of $B$ roots, let $z^u_{r,0} = f_\theta(x^u_{r,0})$ and $y^{u,a}_{r,h} = f_\theta(x^{u,a}_{r,h})$, where $u \in \{c, v\}$ denotes the clean or video rendering and $a \in \{p, n\}$ the sampled branch role. The background objective aligns the root and both realized branch futures:

$$\mathcal{L}_{\mathrm{bg}} = \frac{1}{B(2H+1)} \sum_{r=1}^{B} \Big[ \mathrm{rms}^2(z^c_{r,0}, z^v_{r,0}) + \sum_{h=1}^{H} \sum_{a\in\{p,n\}} \mathrm{rms}^2(y^{c,a}_{r,h}, y^{v,a}_{r,h}) \Big]. \tag{4}$$

Every term compares observations with identical physical content and different backgrounds. The loss therefore identifies background variation to suppress without aligning unrelated states, and gradients reach both encodings in each pair.

### 3.3 Outcome-gated action-fork learning

Background alignment alone does not ensure that the latent dynamics preserve control-relevant differences. We therefore use the executed action forks to identify when two actions actually produce distinguishable outcomes. For root $r$, let $s^p_{r,h}$ and $s^n_{r,h}$ be the task-state vectors reached by the sampled branches at rollout step $h$. Let $D_s$ be the task-state dimension and $\sigma_j = \max\{\mathrm{Std}_{\mathrm{train}}[s_j], 10^{-6}\}$ the guarded population standard deviation of coordinate $j$ over executed training futures. Because $p$ and $n$ are sampled as distinct branches, the standardized outcome gap and eligibility gate are

$$g_{r,h} = \left[ \frac{1}{D_s} \sum_{j=1}^{D_s} \left( \frac{s^p_{r,h,j} - s^n_{r,h,j}}{\sigma_j} \right)^2 \right]^{1/2}, \tag{5}$$

$$q_{r,h} = \mathbf{1}[g_{r,h} > \tau]. \tag{6}$$

Here $\tau$ is the outcome-separation threshold. Both quantities are detached from optimization. A different commanded action is not sufficient to activate the gate: separation is required only when the realized physical outcomes exceed $\tau$. This avoids forcing apart actions that are locally ineffective or otherwise lead to indistinguishable states.

For each visual condition $u \in \{c, v\}$, the model rolls out the predicted branch's action sequence:

$$\hat{z}^u_{r,h} = d^{(h)}_\theta(z^u_{r,0}, a^p_{r,1:h}). \tag{7}$$

Let $i = (r, u)$ index the $B^\star = 2B$ visual instances. Because the two renderings share physics, $g_{i,h} = g_{r,h}$ and $q_{i,h} = q_{r,h}$. With $w_h = \rho^{h-1}$, define the prediction error, sibling-comparison error, and distance between the two executed futures as

$$e^p_{i,h} = \mathrm{rms}^2(\hat{z}_{i,h}, \mathrm{sg}(y^p_{i,h})), \qquad e^n_{i,h} = \mathrm{rms}^2(\hat{z}_{i,h}, \mathrm{sg}(y^n_{i,h})), \tag{8}$$

$$d_{i,h} = \sqrt{\max\{\mathrm{rms}^2(y^p_{i,h}, y^n_{i,h}), 10^{-12}\}}. \tag{9}$$

Here $\mathrm{sg}(\cdot)$ stops gradients and $[t]_+ = \max\{t, 0\}$. Using the gap-dependent margin $\mu(g)$ specified in Section A.4, the three fork losses are

$$\mathcal{L}_{\mathrm{pred}} = \frac{1}{HB^\star} \sum_{i=1}^{B^\star} \sum_{h=1}^{H} w_h e^p_{i,h}, \tag{10}$$

$$Z = \max\left\{1, \sum_{i=1}^{B^\star} \sum_{h=1}^{H} w_h q_{i,h}\right\}, \tag{11}$$

$$\mathcal{L}_{\mathrm{sep}} = \frac{1}{Z} \sum_{i=1}^{B^\star} \sum_{h=1}^{H} w_h q_{i,h} \left[\mu(g_{i,h}) - d_{i,h}\right]^2_+, \tag{12}$$

$$\mathcal{L}_{\mathrm{rank}} = \frac{1}{Z} \sum_{i=1}^{B^\star} \sum_{h=1}^{H} w_h q_{i,h} \left[e^p_{i,h} - e^n_{i,h} + \mu(g_{i,h})^2\right]_+. \tag{13}$$

Prediction trains the action-conditioned rollout toward its executed future. Separation preserves a margin between futures whose physical outcomes pass the gate, while ranking makes the rollout closer to its own executed target than to the sibling target. The sibling is an encoded executed future, not a second learned rollout. When $q = 0$, both separation and ranking contribute exactly zero; detailed gradient paths are recorded in the appendix.

### 3.4 Joint training and deployment

The complete objective augments the base world-model loss:

$$\mathcal{L}_{\mathrm{wm}} = \mathcal{L}_{\mathrm{base}} + \lambda_{\mathrm{bg}}\mathcal{L}_{\mathrm{bg}} + \lambda_{\mathrm{pred}}\mathcal{L}_{\mathrm{pred}} + \lambda_{\mathrm{sep}}\mathcal{L}_{\mathrm{sep}} + \lambda_{\mathrm{rank}}\mathcal{L}_{\mathrm{rank}}. \tag{14}$$

Here, $\mathcal{L}_{\mathrm{base}}$ is the original objective of the base world model. The coefficients and ablation settings are reported in Section A.4. All terms are summed within the existing world-model update, which retains one backward pass, one gradient-clipping operation, and one optimizer step; TD-MPC2's separate policy update is unchanged. The auxiliary objectives train the existing encoder and latent dynamics under the gradient boundaries described in Section A. They introduce neither a new network nor additional trainable parameters.

Privileged task-state readout, exact prefix replay, paired rendering, construction masks, and action forks are confined to offline data construction. During optimization, the auxiliary objectives use only the capsule's RGB stacks, action sequences, and derived supervision, not raw simulator states. At deployment, the auxiliary capsule is also absent: the trained model receives only RGB stacks and follows TD-MPC2's original inference and planning path. Table 1 summarizes this boundary for the baselines and ablations.

## 4 Experiments

We evaluate ISI-WM on six DMC tasks under a common dynamic-background protocol. The central question is whether the two simulator-derived relations used during training—same-state agreement across backgrounds and outcome-gated sensitivity across action branches—translate into stronger generalization to backgrounds not observed during training. Beyond endpoint returns, we isolate the contribution of each relation and measure the representation shifts induced by background changes and action outcomes. We then examine episode-level behavior and report computational cost.

### 4.1 Experimental setup

**Tasks and training budget.** We evaluate Acrobot Swingup, Cartpole Swingup, Cup Catch, Finger Spin, Reacher Easy, and Walker Walk from the DeepMind Control Suite (Tassa et al., 2018). All agents receive $64 \times 64$ RGB observations; TD-MPC2 and ISI-WM use the carrier's three-frame stack. An episode lasts 500 control steps with action repeat two. Each method is trained for 300k control steps with seeds 6, 7, and 8.

Table 1: Training supervision and evaluation input. Privileged simulator-state access is used to construct ISI-WM labels only; every evaluated policy receives RGB.

| Method | RGB | Sim. state | BG pair | Fork | Test |
|---|---|---|---|---|---|
| TD-MPC2 | ✓ | – | – | – | RGB |
| R2 | ✓ | – | – | – | RGB |
| HRSSM | ✓ | – | – | – | RGB |
| BG only | ✓ | ✓ | ✓ | – | RGB |
| Fork only | ✓ | ✓ | – | ✓ | RGB |
| ISI-WM | ✓ | ✓ | ✓ | ✓ | RGB |

**Dynamic-background protocol.** The source pool contains 100 natural-video clips, partitioned into train (0–79), validation (80–84), support (85–89), and held-out test (90–99) splits. Training backgrounds are drawn only from the train split. At each reset, the environment samples a clip and a random starting frame; the video advances once per control step and loops when necessary. At render time, a mask identifies the native blue background and replaces only those pixels with the current video frame, leaving the task foreground and physics unchanged. *Clean* uses the native DMC rendering, *seen* draws new episodes from the train video split, and *unseen* draws only from the test split. Figure 2 visualizes the three conditions.

**Comparisons.** We compare ISI-WM with RGB TD-MPC2 (Hansen et al., 2024), R2-Dreamer (Morihira et al., 2026), and HRSSM (Sun et al., 2024). The TD-MPC2 baseline uses the same carrier architecture and optimization code as ISI-WM, with the intervention objectives disabled. R2-Dreamer and HRSSM checkpoints are trained under the same task, background, and control-step protocol and evaluated under the three conditions above. Background-only and action-fork-only variants use the same carrier and training budget. Every policy receives only RGB input at evaluation.

**Evaluation and reporting.** For the main comparison, we evaluate 20 episodes per training seed and condition (50 for Cup Catch, whose returns are bimodal); every ablation arm uses 20 episodes for every task and condition. Conditions share the same environment-seed schedule, and the dynamic conditions additionally share a fixed background-seed schedule. Episode returns are averaged within each seed before aggregation across seeds. The reported raw macro return is the arithmetic mean of the six task means, without task normalization.

### 4.2 Generalization to unseen backgrounds

ISI-WM achieves the highest six-task macro-average return under clean, seen, and unseen backgrounds; the advantage persists when every test video is held out from training.

Table 2: Evaluation return at 300k control steps under clean, seen, and unseen backgrounds. Mean ± std over three training seeds; 20 episodes per condition (50 for Cup Catch, whose returns are bimodal). The mean column is the raw macro-average over six tasks.

| Method | Cond. | Acrobot | Cartpole | Cup | Finger | Reacher | Walker | Mean raw |
|---|---|---|---|---|---|---|---|---|
| TD-MPC2 | clean | $3 \pm 2$ | $157 \pm 63$ | $138 \pm 67$ | $850 \pm 143$ | $256 \pm 66$ | $145 \pm 10$ | 258 |
| | seen | $3 \pm 1$ | $192 \pm 5$ | $75 \pm 21$ | $752 \pm 190$ | $203 \pm 28$ | $136 \pm 11$ | 227 |
| | unseen | $3 \pm 1$ | $179 \pm 18$ | $67 \pm 22$ | $678 \pm 146$ | $231 \pm 23$ | $138 \pm 9$ | 216 |
| R2 | clean | $10 \pm 5$ | $109 \pm 28$ | $73 \pm 12$ | $46 \pm 76$ | $58 \pm 12$ | $99 \pm 82$ | 66 |
| | seen | $9 \pm 4$ | $154 \pm 10$ | $79 \pm 2$ | $21 \pm 8$ | $69 \pm 4$ | $80 \pm 59$ | 69 |
| | unseen | $\mathbf{9 \pm 2}$ | $162 \pm 8$ | $79 \pm 1$ | $25 \pm 14$ | $69 \pm 6$ | $98 \pm 61$ | 74 |
| HRSSM | clean | $6 \pm 4$ | $483 \pm 32$ | $32 \pm 46$ | $407 \pm 265$ | $175 \pm 57$ | $491 \pm 103$ | 266 |
| | seen | $8 \pm 2$ | $493 \pm 27$ | $20 \pm 15$ | $401 \pm 265$ | $219 \pm 67$ | $491 \pm 122$ | 272 |
| | unseen | $6 \pm 1$ | $\mathbf{495 \pm 34}$ | $38 \pm 24$ | $399 \pm 267$ | $234 \pm 96$ | $\mathbf{499 \pm 119}$ | 278 |
| ISI-WM | clean | $3 \pm 2$ | $302 \pm 17$ | $518 \pm 232$ | $663 \pm 285$ | $792 \pm 10$ | $467 \pm 9$ | **458** |
| | seen | $4 \pm 1$ | $229 \pm 12$ | $588 \pm 21$ | $885 \pm 5$ | $704 \pm 121$ | $417 \pm 64$ | **471** |
| | unseen | $4 \pm 1$ | $228 \pm 27$ | $\mathbf{537 \pm 36}$ | $\mathbf{843 \pm 20}$ | $\mathbf{630 \pm 43}$ | $365 \pm 65$ | **434** |

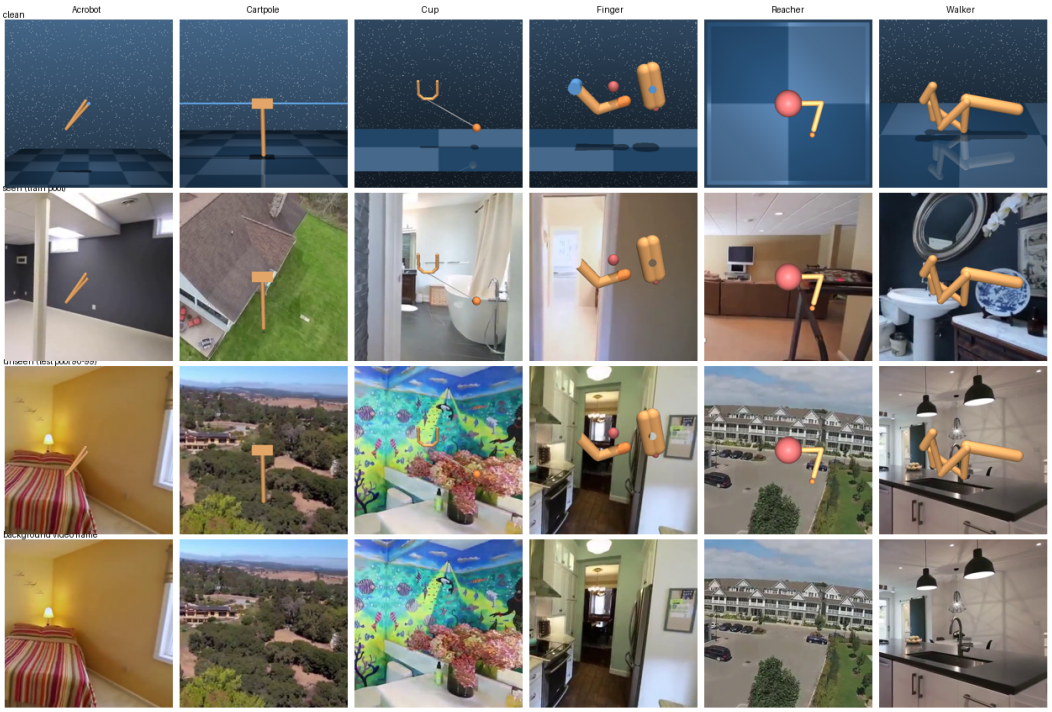


Figure 2: Tasks and background conditions. Rows show the clean rendering, a seen composite using a train-split video, an unseen composite using a held-out test video, and the corresponding raw test-video frame. Within each column, the task state is fixed; compositing changes the visual layer rather than the physics.

On unseen backgrounds, ISI-WM reaches a macro-average return of 434, compared with 278 for HRSSM, 216 for TD-MPC2, and 74 for R2. Its largest per-task gains over TD-MPC2 are on Cup Catch (67 to 537) and Reacher Easy (231 to 630). Performance remains task dependent: HRSSM obtains the highest returns on Cartpole (495) and Walker (499), whereas ISI-WM is strongest on Cup Catch, Finger, and Reacher Easy. Acrobot was not learned by any method at this budget under dynamic backgrounds.

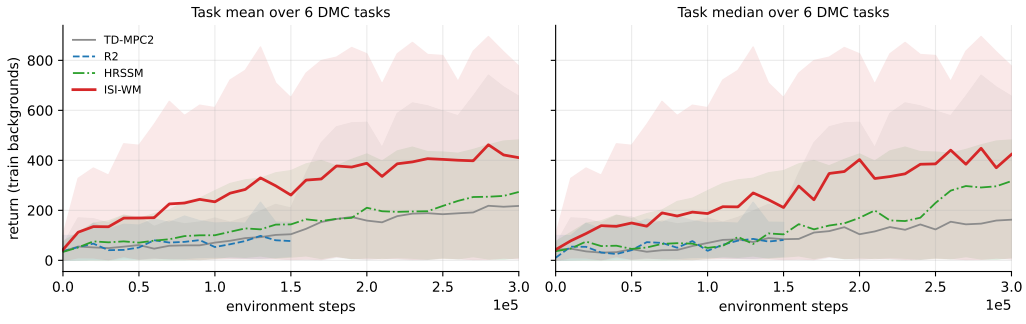


Figure 3: Aggregated return over six tasks under training backgrounds. Left/right: task mean/median; shading: range across tasks. TD-MPC2, ISI-WM, and R2 use periodic evaluation scores; HRSSM uses training-episode returns. Final unseen-background returns in Table 2.

In the recorded training-background trajectories in Figure 3, the ISI-WM curve rises above the other logged curves in both task mean and task median. The corresponding median trend shows that the aggregate pattern is not due to a single high-return task. Unseen-background generalization is

measured by the fixed endpoint evaluation in Table 2; we next separate the contributions of the two intervention signals.

### 4.3 Complementarity of the interventions

Combining the two interventions gives the highest unseen-background macro-average return, while their isolated contributions differ across tasks.

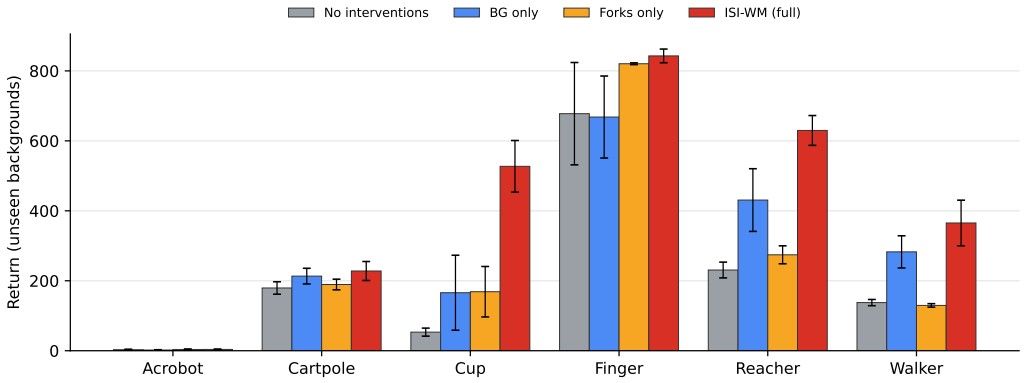


Figure 4: Per-task unseen-background return for the four ablation variants. Bars show means over three training seeds and error bars show sample std; each seed is evaluated for 20 episodes.

The macro-average is 214 with neither intervention, 294 with background-only, 264 with action-fork-only, and 433 with both. The per-task pattern in Figure 4 reveals distinct roles for the two signals: background pairing drives the gain on Reacher, whereas action forks primarily benefit Finger. On Cup, neither intervention alone is sufficient, while their combination produces the largest gain. On Walker, fork-only is slightly below no intervention, whereas combining it with background pairing substantially improves return. This pattern suggests that action-branch supervision alone is insufficient for this task and becomes effective when paired with an explicit background-invariance constraint. Complete results for all three evaluation conditions appear in Table 5. These task-dependent patterns motivate a direct measurement of the learned latent geometry.

### 4.4 Representation analysis

ISI-WM reduces the latent response to background changes relative to the response induced by realized action outcomes.

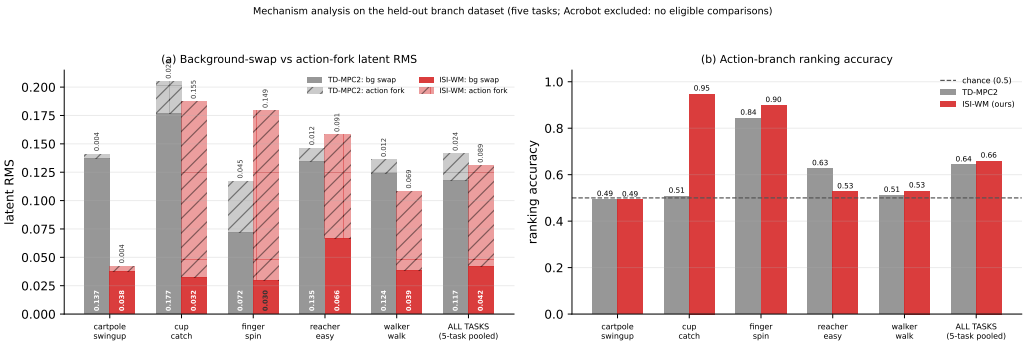


Figure 5: Latent response to held-out interventions over five tasks. Left: background-swap RMS (solid) and eligible action-fork RMS (hatched); the hatched segment height is the action-fork value. Right: action-branch ranking accuracy. The pooled group aggregates all five tasks; Acrobot is excluded because it has no eligible action comparisons.

As shown in Figure 5, ISI-WM compresses background-induced latent shifts to roughly a third of the baseline (0.042 vs 0.117), while expanding action-induced distances nearly four-fold (0.089 vs 0.024). Ranking accuracy stays above chance for both methods (0.64 vs 0.66). Corresponding rollout visualizations and episode-level return distributions appear in Appendix D.

### 4.5 Efficiency

Table 3 summarizes model capacity, action-time inference cost, and wall-clock training time.

Table 3: Model and runtime cost on RTX 4090 GPUs. Inference is measured at batch size one; peak is the maximum allocated CUDA memory. Training is the mean wall time per 300k-control-step job over the matched five-task subset.

| Method | Params (M) | P50 (ms) | P95 (ms) | Peak (MB) | Train (h/job) |
|---|---|---|---|---|---|
| TD-MPC2 | 4.86 | 25.85 | 28.79 | 66.9 | 11.9 |
| R2 | 11.84 | 1.34 | 1.37 | 114.6 | 0.9 |
| HRSSM | 18.68 | 1.16 | 1.20 | 85.1 | 14.7 |
| ISI-WM | 4.86 | 25.78 | 26.48 | 97.4 | 14.8 |

ISI-WM adds no trainable parameters and retains essentially unchanged action latency relative to TD-MPC2 (4.86M parameters; 25.78 vs. 25.85 ms P50), while its measured peak inference allocation is 97.4 versus 66.9 MB. Under the matched five-task, 300k-step protocol, mean training time increases from 11.9 to 14.8 h/job (+24%), consistent with the additional training-only intervention computation. R2 and HRSSM act substantially faster through direct policy heads rather than online MPC; R2 also trains faster, whereas HRSSM has a training time comparable to ISI-WM. Runtime and unseen-background return therefore do not follow the same ordering.

## 5 Conclusion

ISI-WM frames background-robust world-model learning as selective invariance and realizes it through cross-background state pairing and outcome-gated action forks. The resulting objectives update the existing encoder and latent dynamics without adding learned parameters or changing RGB-only deployment. Under a unified dynamic-background protocol, ISI-WM generalizes more strongly to unseen backgrounds than the evaluated baselines, while ablations and latent-response measurements show complementary, task-dependent roles for the two signals.

Training currently uses simulator access to construct matched-state background pairs and action branches, while deployment requires only RGB observations. Applying the same principle to uninstrumented real-world data would require approximate state matching or learned intervention generation. Our experiments isolate dynamic-background shifts; extending the intervention construction to lighting and object-appearance changes remains future work.

### AI use statement

Generative AI tools assisted with language editing and document maintenance. The authors remain responsible for the method, experiments, citations, equations, and reported values.

### Reproducibility statement

The appendix records the implementation constants, evaluation splits, per-seed endpoints, and provenance qualifications used in this draft. The paired-data builder, frozen training implementation, aggregate JSON files, and final table/figure assets are bound to the accompanying artifact.

## REFERENCES


Kartik Ahuja, Divyat Mahajan, Yixin Wang, and Yoshua Bengio. Interventional causal representation learning. In *International Conference on Machine Learning (ICML)*, pp. 372–407, 2023.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Johann Brehmer, Pim de Haan, Phillip Lippe, and Taco S. Cohen. Weakly supervised causal representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 38319–38331, 2022. doi: 10.52202/068431-2776.

Pablo Samuel Castro, Tyler Kastner, Prakash Panangaden, and Mark Rowland. MICo: Improved representations via sampling-based state similarity for Markov decision processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 30113–30126, 2021.

Chang Chen, Yi-Fu Wu, Jaesik Yoon, and Sungjin Ahn. TransDreamer: Reinforcement learning with transformer world models. *arXiv preprint arXiv:2202.09481*, 2022.

Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *Conference on Robot Learning (CoRL)*, pp. 66–75, 2020.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 4754–4765, 2018.

Fei Deng, Ingook Jang, and Sungjin Ahn. DreamerPro: Reconstruction-free model-based reinforcement learning with prototypical representations. In *International Conference on Machine Learning (ICML)*, pp. 4956–4975, 2022.

Jiameng Fan and Wenchao Li. DRIBO: Robust deep reinforcement learning via multi-view information bottleneck. In *International Conference on Machine Learning (ICML)*, pp. 6074–6102, 2022.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite Markov decision processes. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 162–169, 2004.

Norm Ferns, Prakash Panangaden, and Doina Precup. Bisimulation metrics for continuous Markov decision processes. *SIAM Journal on Computing*, 40(6):1662–1714, 2011. doi: 10.1137/10080484X.

Xiang Fu, Ge Yang, Pulkit Agrawal, and Tommi Jaakkola. Learning task informed abstractions. In *International Conference on Machine Learning (ICML)*, pp. 3480–3491, 2021.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G. Bellemare. DeepMDP: Learning continuous latent space models for representation learning. In *International Conference on Machine Learning (ICML)*, pp. 2170–2179, 2019.

Robert Givan, Thomas L. Dean, and Matthew Greig. Equivalence notions and model minimization in Markov decision processes. *Artificial Intelligence*, 147(1–2):163–223, 2003. doi: 10.1016/S0004-3702(02)00376-4.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Jeongsoo Ha, Kyungsoo Kim, and Yusung Kim. Dream to generalize: Zero-shot model-based reinforcement learning for unseen visual distractions. In *AAAI Conference on Artificial Intelligence (AAAI)*, pp. 7802–7810, 2023. doi: 10.1609/aaai.v37i6.25945.

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 (ICML)*, pp. 2555–2565, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations (ICLR)*, 2020.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering Atari with discrete world models. In *International Conference on Learning Representations (ICLR)*, 2021.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025. doi: 10.1038/s41586-025-08744-2.

Nicklas Hansen and Xiaolong Wang. Generalization in reinforcement learning by soft data augmentation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13611–13617, 2021. doi: 10.1109/ICRA48506.2021.9561103.

Nicklas Hansen, Rishabh Jangir, Yu Sun, Guillem Alenyà, Pieter Abbeel, Alexei A. Efros, Lerrel Pinto, and Xiaolong Wang. Self-supervised policy adaptation during deployment. In *International Conference on Learning Representations (ICLR)*, 2021.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. In *International Conference on Machine Learning (ICML)*, pp. 8387–8406, 2022.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations (ICLR)*, 2024.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. In *International Conference on Learning Representations (ICLR)*, 2020.

Michael Laskin, Aravind Srinivas, and Pieter Abbeel. CURL: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning (ICML)*, pp. 5639–5650, 2020a.

Misha Laskin, Kimin Lee, Adam Stooke, Lerrel Pinto, Pieter Abbeel, and Aravind Srinivas. Reinforcement learning with augmented data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 19884–19895, 2020b.

Lihong Li, Thomas J. Walsh, and Michael L. Littman. Towards a unified theory of state abstraction for MDPs. In *International Symposium on Artificial Intelligence and Mathematics (ISAIM)*, 2006.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning (ICML)*, pp. 4114–4124, 2019.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. In *International Conference on Learning Representations (ICLR)*, 2016.

Naoki Morihira, Amal Nahar, Kartik Bharadwaj, Yasuhiro Kato, Akinobu Hayashi, and Tatsuya Harada. R2-Dreamer: Redundancy-reduced world models without decoders or augmentation. In *International Conference on Learning Representations (ICLR)*, 2026.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009. doi: 10.1017/CBO9780511803161.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: Identification and confidence intervals. *Journal of the Royal Statistical Society: Series B*, 78(5):947–1012, 2016. doi: 10.1111/rssb.12167.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, 2017.

Lerrel Pinto, Marcin Andrychowicz, Peter Welinder, Wojciech Zaremba, and Pieter Abbeel. Asymmetric actor critic for image-based robot learning. In *Proceedings of Robotics: Science and Systems*, 2018. doi: 10.15607/RSS.2018.XIV.008.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021. doi: 10.1109/JPROC.2021.3058954.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering Atari, Go, Chess and Shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020. doi: 10.1038/s41586-020-03051-4.

Austin Stone, Oscar Ramirez, Kurt Konolige, and Rico Jonschkowski. The distracting control suite – a challenging benchmark for reinforcement learning from pixels. *arXiv preprint arXiv:2101.02722*, 2021.

Ruixiang Sun, Hongyu Zang, Xin Li, and Riashat Islam. Learning latent dynamic robust representations for world models. In *International Conference on Machine Learning (ICML)*, pp. 47234–47260, 2024.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. DeepMind Control Suite. *arXiv preprint arXiv:1801.00690*, 2018.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, 2017. doi: 10.1109/IROS.2017.8202133.

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5–6):544–557, 2009. doi: 10.1016/j.neunet.2009.06.042.

Rishi Veerapaneni, John D. Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua B. Tenenbaum, and Sergey Levine. Entity abstraction in visual model-based reinforcement learning. In *Conference on Robot Learning (CoRL)*, pp. 1439–1456, 2020.

Julius von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 16451–16467, 2021.

Tongzhou Wang, Simon Du, Antonio Torralba, Phillip Isola, Amy Zhang, and Yuandong Tian. Denoised MDPs: Learning world models better than the world itself. In *International Conference on Machine Learning (ICML)*, pp. 22591–22612, 2022.

Yucen Wang, Shenghua Wan, Le Gan, Shuai Feng, and De-Chuan Zhan. AD3: Implicit action is the key for world models to distinguish the diverse visual distractors. In *International Conference on Machine Learning (ICML)*, pp. 51546–51568, 2024.

Grady Williams, Paul Drews, Brian Goldfain, James M. Rehg, and Evangelos A. Theodorou. Aggressive driving with model predictive path integral control. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1433–1440, 2016. doi: 10.1109/ICRA.2016.7487277.

Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. DayDreamer: World models for physical robot learning. In *Conference on Robot Learning (CoRL)*, pp. 2226–2240, 2023.

Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Reinforcement learning with prototypical representations. In *International Conference on Machine Learning (ICML)*, pp. 11920–11931, 2021a.

Denis Yarats, Ilya Kostrikov, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International Conference on Learning Representations (ICLR)*, 2021b.

Denis Yarats, Amy Zhang, Ilya Kostrikov, Brandon Amos, Joelle Pineau, and Rob Fergus. Improving sample efficiency in model-free reinforcement learning from images. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pp. 10674–10681, 2021c. doi: 10.1609/aaai.v35i12.17276.

Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Mastering visual continuous control: Improved data-augmented reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2022.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning (ICML)*, pp. 12310–12320, 2021.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. In *International Conference on Learning Representations (ICLR)*, 2021.

Chuning Zhu, Max Simchowitz, Siri Gadipudi, and Abhishek Gupta. RePo: Resilient model-based reinforcement learning by regularizing posterior predictability. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 32445–32467, 2023. doi: 10.52202/075280-1409.

# A Implementation and protocol details

## A.1 Base setup

The base model uses three stacked RGB frames ($9 \times 64 \times 64$), a 512-dimensional latent, 500-control-step episodes, action repeat two, and training seeds 6, 7, and 8. Background clips 0–79 form the training split, 80–84 the validation split, 85–89 the support split, and 90–99 the test split.

## A.2 Intervention data construction

The offline source set for each task contains 80 intervention starting points (roots), split into 50 training, 10 validation, and 20 test roots; auxiliary batches use only the training roots. Root times cycle through control steps 80, 160, 240, 320, and 400. A frozen clean-evaluation policy generates a common action prefix, which each trajectory exactly replays from a fresh reset to reconstruct the same physical root state; simulator snapshots are not restored.

At each reconstructed root, the source record contains matched clean/video views and futures generated by the zero-action branch and the positive and negative branch $\pm me_i$ along every action axis. All branches share the same $H - 1$ continuation actions, sampled once per root from $\mathrm{Uniform}([-1, 1]^A)$, after this first-step intervention. The learner samples one ordered predicted/sibling pair with balanced coverage of the zero, positive, and negative branches; these role names describe their use in the objective rather than the sign of an action.

Matched clean/video roots share one spatial crop. At each future horizon, the two branch roles and two backgrounds also share one crop, preventing crop noise from entering the comparison. For Equation (5), each task-state coordinate is normalized by its population standard deviation over executed future states from all training roots and branches. The common root state is excluded, and a $10^{-6}$ floor is applied.

Algorithm 1 summarizes the offline construction and its corresponding joint world-model update.

## A.3 Optimization and gradient paths

The background loss $\mathcal{L}_{\mathrm{bg}}$ updates the encoder paths for paired roots and futures. The prediction and ranking losses $\mathcal{L}_{\mathrm{pred}}$ and $\mathcal{L}_{\mathrm{rank}}$ update the root encoder and latent dynamics while treating executed future encodings as stop-gradient targets; the separation loss $\mathcal{L}_{\mathrm{sep}}$ updates the two executed future encodings. The base TD-MPC2 objectives retain their original gradient paths.

**Algorithm 1** Offline intervention construction and joint ISI-WM update.

**Offline construction**

**for** each training source-root prefix $r$ **do**
  Fresh-reset and replay $r$ to reconstruct $s_{r,0}$; record the paired roots $x^{c}_{r,0}$ and $x^{v}_{r,0}$.
  Sample once a continuation $a_{r,2:H}$ shared by all sibling branches.
  **for** each first action $k \in \mathcal{K}$ in Equation (2) **do**
    Fresh-reset and replay $r$; execute $k$ followed by $a_{r,2:H}$.
    Record $s^{k}_{r,1:H}$ and the paired RGB futures $x^{c,k}_{r,1:H}, x^{v,k}_{r,1:H}$.
  **end for**
**end for**
Estimate $\{\sigma_j\}$ from executed training futures, excluding roots; derive $g$ and $q$ by Equations (5) and (6) for every ordered branch pair.
Materialize a capsule $\mathcal{C}$ containing RGB stacks, actions, branch identities, $g$, and $q$; discard raw task states.

**Each world-model update**

Sample a base replay minibatch and compute $\mathcal{L}_{\text{base}}$.
Sample $B$ roots and balance $(p, n)$ over $(0, +me_i)$, $(0, -me_i)$, $(+me_i, -me_i)$, and their reverses; share one crop per correspondence.
Encode paired root observations and executed $p/n$ futures, and compute $\mathcal{L}_{\text{bg}}$ by Equation (4).
Roll out $d_\theta$ from each root using only $p$'s action sequence.
Compute $\mathcal{L}_{\text{pred}}$, $\mathcal{L}_{\text{sep}}$, and $\mathcal{L}_{\text{rank}}$ by Equations (10), (12) and (13); $q$ gates only $\mathcal{L}_{\text{sep}}$ and $\mathcal{L}_{\text{rank}}$.
Form $\mathcal{L}_{\text{wm}}$ by Equation (14); apply one backward pass, gradient clipping, and optimizer step to the existing world model.
Run the base agent's separate policy update unchanged.

### A.4 Intervention hyperparameters

All six tasks use the same intervention setting, without task-specific tuning. The fixed values are summarized below.

| Setting | Value |
|---|---|
| Branch magnitude | $m = 0.8$ in the normalized action domain $[-1, 1]^A$ |
| Eligibility gate | $q = \mathbf{1}[g > 0.05]$ |
| Latent margin | $\mu(g) = 0.05 + 0.15\,\text{clamp}(g, 0, 1) \in [0.05, 0.20]$ |
| Fork horizon | $H = 3$ training steps; five steps stored in the source records |
| Full-model coefficients | $(\lambda_{\text{bg}}, \lambda_{\text{pred}}, \lambda_{\text{sep}}, \lambda_{\text{rank}}) = (1.0, 1.0, 1.0, 1.0)$ |
| Sampling | $B = 8$ ordered branch pairs per update; update frequency 1 |
| Temporal discount | $\rho = 0.5$ (the TD-MPC2 default) |
| Correspondence mode | `correct` (background/fork correspondences retained) |

The branch magnitude balances a discernible action intervention against excessive departure from the behavior-action distribution. The eligibility gate filters near-coincident futures, while the gap-dependent margin scales the separation supervision with the realized outcome gap. The three-step horizon and temporal discount focus the auxiliary objective on more reliable short-term outcomes; all other loss and sampling settings are shared across tasks. Background-only and fork-only use $(1, 0, 0, 0)$ and $(0, 1, 1, 1)$, respectively, in the coefficient order shown above.

### A.5 Background compositor

The DMC compositor identifies native background pixels by the renderer's blue key ($B > G$ and $B > R$) and substitutes the current video frame there. This mask is a construction device, not a learned-model input. At each reset, the environment chooses a clip and a random valid start; one source frame is advanced per control step and clips loop.

## B PER-SEED PRIMARY ENDPOINTS

Table 4: Per-seed return triplets (seed 6 / 7 / 8). Evaluations use 20 episodes per seed and condition, except Cup Catch, which uses 50.

| Task | Method | Clean | Seen | Unseen |
|---|---|---|---|---|
| Finger | TD-MPC2 | 922.4/942.1/684.9 | 843.5/878.5/533.9 | 744.5/778.6/509.9 |
| | ISI-WM | 910.9/352.1/725.9 | 880.8/890.4/884.9 | 822.8/843.8/861.9 |
| Cartpole | TD-MPC2 | 88.8/213.4/170.2 | 194.8/186.6/195.0 | 159.3/193.2/185.7 |
| | ISI-WM | 299.0/321.1/286.7 | 242.3/223.8/219.8 | 258.3/205.9/219.6 |
| Reacher | TD-MPC2 | 185.3/315.8/266.6 | 184.9/234.9/189.9 | 205.2/239.3/247.8 |
| | ISI-WM | 793.5/781.5/801.7 | 809.8/572.6/728.4 | 677.9/596.9/614.4 |
| Cup | TD-MPC2 | 174.5/60.7/178.8 | 99.0/61.7/63.5 | 66.5/88.5/45.2 |
| | ISI-WM | 254.6/688.9/610.7 | 585.4/569.5/610.5 | 496.8/545.4/567.7 |
| Walker | TD-MPC2 | 137.8/156.1/142.4 | 148.5/131.8/126.8 | 140.5/127.7/144.7 |
| | ISI-WM | 457.5/468.2/475.3 | 364.2/399.3/487.8 | 301.4/362.5/431.7 |
| Acrobot | TD-MPC2 | 1.6/1.3/4.8 | 1.4/3.7/3.2 | 1.3/4.1/3.4 |
| | ISI-WM | 2.5/4.9/2.0 | 3.4/3.8/6.1 | 4.7/4.0/2.5 |

## C FULL-CONDITION ABLATION

Table 5: Clean/seen/unseen ablation returns reproduced from the archived per-seed evaluation JSONs. Each cell is the three-seed summary from 20 episodes per seed and condition. The final block reports the macro-average over all six tasks for each condition.

| Task | Variant | Clean | Seen | Unseen |
|---|---|---|---|---|
| Acrobot | None | $3 \pm 2$ | $3 \pm 1$ | $3 \pm 1$ |
| | BG only | $6 \pm 2$ | $4 \pm 3$ | $2 \pm 1$ |
| | Fork only | $3 \pm 3$ | $6 \pm 3$ | $4 \pm 2$ |
| | Full | $3 \pm 2$ | $4 \pm 1$ | $4 \pm 1$ |
| Cartpole | None | $157 \pm 63$ | $192 \pm 5$ | $179 \pm 18$ |
| | BG only | $172 \pm 96$ | $201 \pm 10$ | $213 \pm 22$ |
| | Fork only | $157 \pm 74$ | $187 \pm 15$ | $189 \pm 15$ |
| | Full | $302 \pm 17$ | $229 \pm 12$ | $228 \pm 27$ |
| Cup | None | $147 \pm 62$ | $68 \pm 67$ | $53 \pm 11$ |
| | BG only | $175 \pm 27$ | $198 \pm 55$ | $166 \pm 107$ |
| | Fork only | $141 \pm 5$ | $196 \pm 89$ | $169 \pm 72$ |
| | Full | $555 \pm 204$ | $583 \pm 84$ | $527 \pm 74$ |
| Finger | None | $850 \pm 143$ | $752 \pm 190$ | $678 \pm 146$ |
| | BG only | $752 \pm 152$ | $702 \pm 137$ | $668 \pm 117$ |
| | Fork only | $784 \pm 91$ | $891 \pm 5$ | $820 \pm 3$ |
| | Full | $663 \pm 285$ | $885 \pm 5$ | $843 \pm 20$ |
| Reacher | None | $256 \pm 66$ | $203 \pm 28$ | $231 \pm 23$ |
| | BG only | $670 \pm 146$ | $434 \pm 123$ | $431 \pm 90$ |
| | Fork only | $132 \pm 41$ | $234 \pm 22$ | $274 \pm 26$ |
| | Full | $792 \pm 10$ | $704 \pm 121$ | $630 \pm 43$ |
| Walker | None | $145 \pm 10$ | $136 \pm 11$ | $138 \pm 9$ |
| | BG only | $381 \pm 47$ | $287 \pm 30$ | $283 \pm 46$ |
| | Fork only | $79 \pm 24$ | $123 \pm 2$ | $130 \pm 5$ |
| | Full | $467 \pm 9$ | $417 \pm 64$ | $365 \pm 65$ |
| Macro avg. | None | **260** | **226** | **214** |
| | BG only | **359** | **304** | **294** |
| | Fork only | **216** | **273** | **264** |
| | Full | **464** | **470** | **433** |

## D QUALITATIVE AND EPISODE-LEVEL ANALYSIS

Episode traces and return distributions reveal the near-binary behavior behind the Cup Catch averages. The rollout comparison in Figure 6 illustrates how ISI-WM maintains task-relevant tracking as the background changes. The Cup Catch distributions in Figure 7 remain near-binary across all three conditions, with mass near zero and near the maximum return. The Acrobot distributions remain concentrated near zero for every method, matching the task-level result in Table 2.




Figure 6: All rows show the $64 \times 64$ policy observation upscaled for display; all methods receive identical $64 \times 64$ inputs. TD-MPC2, R2, and HRSSM obtain zero return; ISI-WM tracks and secures the ball (return 872).

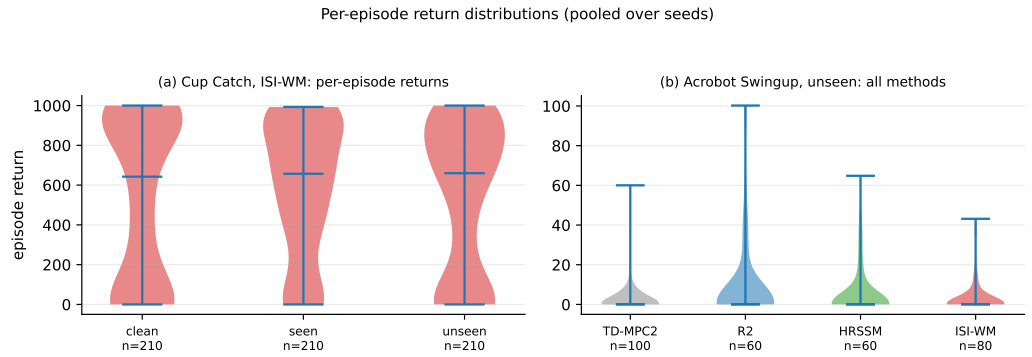


Figure 7: Per-episode return distributions pooled over the archived evaluations. Left: ISI-WM on Cup Catch under clean, seen, and unseen backgrounds. Right: all methods on Acrobot Swingup under unseen backgrounds. Sample counts are shown below each distribution.