FOCUSDREAMER: TASK-DIRECTED MODEL-BASED REINFORCEMENT LEARNING GUIDED BY RETURN RELEVANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Model-based reinforcement learning improves interaction efficiency through learned world models and imagination, but model bias can distort return differences among local behaviors and misguide policy optimization. When local alternatives are weakly supported by experience, their predictions may be biased toward familiar latent dynamics, causing real return advantages to shrink or even reverse in imagination. Therefore, under limited interaction budgets, it is more sensible to prioritize states where local behavioral distinctions matter more for return. We propose **FocusDreamer**, which learns state-level return relevance from completed real trajectories and uses it to focus exploration, model learning, and imagination without altering the original Dreamer objectives. Return relevance is characterized by criticality, which measures its relative strength, and polarity, which reflects the direction of return-predictive progression. Criticality determines where learning resources should be concentrated, while polarity modulates policy-side intervention. Across visual DMControl and Meta-World, FocusDreamer improves performance and sample efficiency, achieving a 53.1% average return improvement over DreamerV3 on DMControl.

1 INTRODUCTION

World-model reinforcement learning improves interaction efficiency by learning predictive dynamics from real experience and optimizing policies through model-generated trajectories (Hafner et al., 2025; Hansen et al., 2024; Wang et al., 2024). Modern latent agents such as DreamerV3 perform substantial actor–value learning in imagined rollouts, making prediction accuracy directly relevant to policy optimization (Hafner et al., 2025). Consequently, inaccurate future predictions can distort behavioral differences used for value estimation and policy improvement. Yet reliably modeling all potentially reachable futures from limited real interaction remains difficult, especially in high-dimensional continuous-control problems (Rigter et al., 2024; Duan et al., 2024; Liu et al., 2026). Reliable world-model learning under limited interaction therefore remains a fundamental challenge.

Existing approaches improve world-model reliability from two complementary perspectives. One learns more useful dynamics from collected experience through improved representations and predictive objectives (Deng et al., 2022; Burchi & Timofte, 2025; Zhou et al., 2025; Morihira et al., 2026), including value-aware model learning (Voelcker et al., 2022; 2025). Yet such improvements cannot eliminate errors on weakly covered or unseen behaviors (Duan et al., 2024; Liu et al., 2026). The other expands behavioral coverage through exploration, information gain, or uncertainty estimation (Sekar et al., 2020; Rigter et al., 2024; Sancaktar et al., 2025; Liu et al., 2026). However, limited real-world interaction makes it impractical to cover all uncertain or poorly modeled regions, while indiscriminately improving model accuracy under fixed interaction and update budgets may divert resources from policy exploration and optimization. Thus, effective world-model learning requires identifying where limited learning resources are most valuable for policy optimization.

The key question is therefore where limited interaction and modeling effort should be concentrated. Learned latent models can exhibit attractor-like behavior toward well-supported or high-density regions, with similar effects observed in latent world-model rollouts (Alain & Bengio, 2014; Fumero et al., 2026; Berger et al., 2026). We find that this tendency can obscure useful behavioral alter-

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.

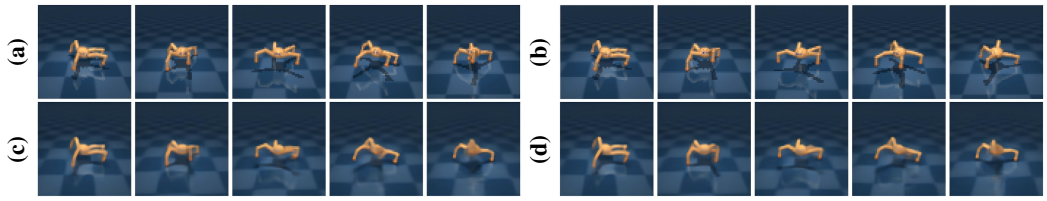

Figure 1: Motivating example. (a,b) Real reference and alternative trajectories; (c,d) their imagined counterparts. The alternative clearly deviates in the real environment but remains close to the reference in imagination, with its cumulative normalized return advantage changing from $+0.0739$ in the environment to -0.0259 in the world model.

natives: as illustrated in Fig. 1, an alternative that clearly deviates from the reference in the real environment may remain close to it in imagination, and its return advantage can even become negative. Because the world model is trained on policy-generated data while the policy is optimized through the learned model (Hafner et al., 2025; Liu et al., 2026), this reduces the incentive to revisit the alternative and limits corrective evidence. Therefore, exploration should focus on states where local behavioral distinctions matter more for task return, with the acquired experience further emphasized in world-model learning and imagination.

Motivated by this analysis, we propose **FocusDreamer**, a task-directed model-based RL framework that allocates limited learning resources according to trajectory returns. FocusDreamer learns two complementary signals from an episodic-return predictor: *criticality*, which measures a state's relevance to final return, and *polarity*, which captures return-predictive progression. These signals guide three stages of learning: focused exploration acquires alternative experience, focused model learning improves the prediction of return-relevant dynamics, and focused imagination reuses the recovered behavioral distinctions for policy optimization. FocusDreamer preserves the original Dreamer world-model and actor–value objectives.

We evaluate FocusDreamer on visual continuous-control tasks from DMControl (Tunyasuvunakool et al., 2020) and robotic manipulation tasks from Meta-World (Yu et al., 2020a). FocusDreamer consistently improves performance and sample efficiency over strong model-free and world-model baselines. Beyond aggregate benchmark performance, counterfactual analyses provide direct evidence for the proposed mechanism: learned world models can underestimate better local alternatives, return criticality identifies states with larger improvement opportunities, and FocusDreamer better preserves these behavioral value differences in imagination. Ablations further demonstrate the complementary contribution of each component.

Our main contributions are:

- We propose **FocusDreamer**, a task-directed model-based RL framework that uses trajectory-derived return relevance, characterized by *criticality* and *polarity*, to guide focused exploration, world-model learning, and imagination.
- We identify *alternative value-gap contraction*, where learned imagination can distort the real return advantage of better local alternatives, and theoretically connect criticality and polarity to local improvement opportunity and return progression under explicit regularity assumptions.
- Experiments on DMControl and Meta-World demonstrate improved performance and sample efficiency, while counterfactual analyses show that return criticality localizes larger improvement opportunities and FocusDreamer better preserves behavioral value differences.

2 Related Work

World-model RL and task-relevant model learning. World-model reinforcement learning has evolved from latent dynamics and planning (Ha & Schmidhuber, 2018; Hafner et al., 2019) to imagination-based policy learning (Hafner et al., 2020; 2025), scalable model-based control (Hansen et al., 2024; Wang et al., 2024; Pu et al., 2025), and Transformer world models (Zhang et al., 2023; Burchi & Timofte, 2025). Recent variants further improve world-model learning through

reconstruction-free prediction and task harmonization (Burchi & Timofte, 2024; Ma et al., 2024). Model errors in poorly supported regions can misguide policy optimization, motivating conservative model usage (Janner et al., 2019; Yu et al., 2020b; Kidambi et al., 2020), targeted data acquisition (Rigter et al., 2024; Duan et al., 2024), and uncertainty-guided exploration or imagination (Sekar et al., 2020; Sancaktar et al., 2025; Liu et al., 2026); Knowledge Boundary (Liu et al., 2026), for example, uses ensemble disagreement to guide exploration and filter unreliable imagination. Related studies also report attractor-like latent dynamics (Fumero et al., 2026; Berger et al., 2026). Meanwhile, DreamerPro, TWISTER, DINO-WM, and R2-Dreamer improve task-relevant representations (Deng et al., 2022; Burchi & Timofte, 2025; Zhou et al., 2025; Morihira et al., 2026), value-aware methods shape model learning with downstream value (Voelcker et al., 2022; 2025). Unlike reliability-oriented methods that locate potentially unreliable regions and representation-oriented methods that improve what is learned from existing data, FocusDreamer uses trajectory-derived return relevance to identify where local behavioral distinctions matter more for return, guiding focused exploration, model learning, and imagination.

Trajectory-return guidance and prioritized allocation. A complementary line uses completed trajectory outcomes for long-horizon credit assignment. RUDDER and Randomized Return Decomposition redistribute episodic returns over time (Arjona-Medina et al., 2019; Ren et al., 2022), while Diaster estimates sub-trajectory reward differences and Skill or Luck decomposes return through advantage-based contributions (Lin et al., 2024; Pan & Schölkopf, 2024). FocusDreamer's non-uniform replay is also related in spirit to prioritized experience replay (Schaul et al., 2016). However, FocusDreamer neither constructs proxy rewards nor prioritizes transitions according to TD error. Instead, completed episodic returns provide a state-level relevance signal that coordinates where to acquire new evidence, where to fit the world model more frequently, and where to reuse the learned model for imagination.

3 FocusDreamer

3.1 Problem Formulation and Framework

Following latent world-model agents based on recurrent state-space models (Hafner et al., 2019; 2020; 2025), an RSSM infers posterior latent states z_t from visual observations, the world model is learned from replay $\mathcal{D}$, and actor–value optimization is performed in latent imagination. For the current policy action a_π and a nearby alternative a, let Q_P^H and $\widehat{Q}^H$ denote the H-step returns under the real environment dynamics P and the learned world model, respectively, using the same continuation after the first action. We define the corresponding relative returns as

$$\Delta_P^H(z,a) = Q_P^H(z,a) - Q_P^H(z,a_\pi), \qquad \widehat{\Delta}^H(z,a) = \widehat{Q}^H(z,a) - \widehat{Q}^H(z,a_\pi). \tag{1}$$

When $\Delta_P^H > 0$ but $\widehat{\Delta}^H < \Delta_P^H$, imagination contracts a real improvement; $\widehat{\Delta}^H \leq 0$ further hides the better alternative. FocusDreamer asks where such behavioral distinctions are worth additional support and uses trajectory-derived return relevance to coordinate exploration, model learning, and imagination.

Figure 2 summarizes this design. FocusDreamer first learns return relevance only from completed real trajectories, so the allocation signal is grounded in realized task outcomes rather than in the world model's own uncertainty. Criticality provides a nonnegative importance score for locating consequential states, while polarity separates favorable from unfavorable return-predictive progression. These retrospective annotations are then reused at different stages according to their roles: exploration needs an online approximation before the episode terminates, world-model learning uses completed criticality annotations directly, and imagination uses both criticality and polarity after replay states are available. The three stages share the same state-selection principle but do not share a new optimization objective, which keeps FocusDreamer compatible with the original Dreamer losses.

3.2 State-Level Return Relevance

For a completed real episode $\tau = \{(o_t, a_t, r_t)\}_{t=1}^T$ of length T, we use its empirical undiscounted return $J_\tau = \sum_{t=1}^T r_t$ as trajectory-level supervision. Let z_t denote the RSSM posterior latent state

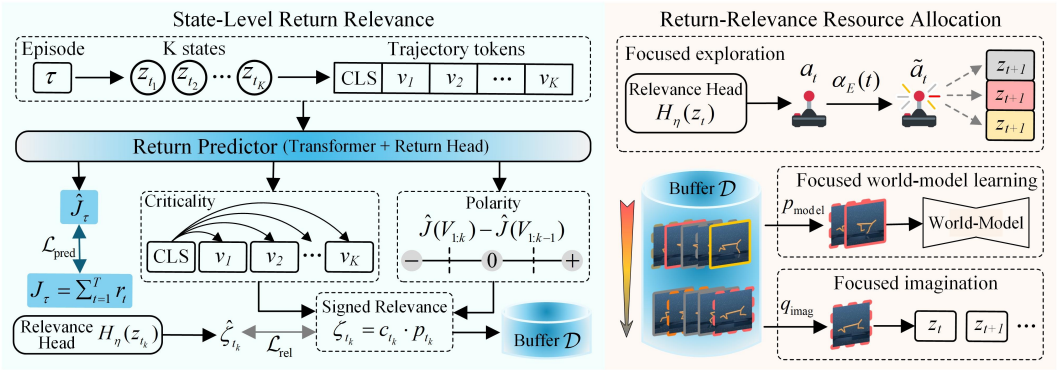

Figure 2: Overview of FocusDreamer. State-level return relevance is estimated from completed episodes and used to allocate learning resources across focused exploration, world-model learning, and imagination, while keeping the original Dreamer objectives unchanged.

at time t, and define the detached relevance feature $\bar{z}_t = \mathrm{sg}(z_t)$, which has the same forward value as z_t but is treated as an independent input to the return-relevance branch. We sparsely sample K temporally ordered states $\{z_{t_k}\}_{k=1}^{K}$ from the completed episode and construct trajectory tokens $v_k = e_{\mathrm{traj}}(\bar{z}_{t_k})$, where e_{traj} is a lightweight projection. A lightweight Transformer processes $V = [v_1, \ldots, v_K]$ together with a learnable CLS token and predicts the episodic return:

$$\widehat{J}_\tau = g_\nu(h_{\mathrm{cls}}), \qquad \mathcal{L}_{\mathrm{pred}} = \mathbb{E}_{\tau\sim\mathcal{D}}\left[(\widehat{J}_\tau - J_\tau)^2\right], \tag{2}$$

where h_{cls} is the Transformer output of the CLS token, g_ν is the return regression head, and $\mathcal{D}$ is the replay buffer. The stop-gradient operation decouples return-relevance learning from the Dreamer representation update.

Criticality. We estimate state-level return relevance through attention rollout (Abnar & Zuidema, 2020). Let A_ℓ denote the multi-head attention matrix at Transformer layer $\ell \in \{1, \ldots, L\}$, and $\bar{A}_\ell$ its head-averaged, residual-augmented, and row-normalized form. The rollout $A^{\mathrm{roll}} = \bar{A}_L \cdots \bar{A}_1$ aggregates attention propagation across layers. We define the criticality of sampled state t_k as

$$c_{t_k} = \frac{[A^{\mathrm{roll}}]_{\mathrm{cls},k}}{\sum_{j=1}^{K}[A^{\mathrm{roll}}]_{\mathrm{cls},j} + \epsilon}, \qquad c_{t_k} \geq 0, \tag{3}$$

where $\epsilon > 0$ ensures numerical stability. The resulting c_{t_k} quantifies the relative relevance of state t_k to the final episodic-return prediction and serves as the primary signal for locating consequential states. Sec. 4 further connects criticality to local return sensitivity under explicit regularity conditions.

Polarity. Criticality identifies *where* return-relevant information lies, but does not characterize the direction of trajectory progression. For the same token sequence, let $V_{1:k} = [v_1, \ldots, v_k]$ denote the prefix ending at state t_k. We compute $d_{t_k} = \widehat{J}(V_{1:k}) - \widehat{J}(V_{1:k-1})$ and define

$$p_{t_k} = \begin{cases} +1, & d_{t_k} > \delta_p, \\ -1, & d_{t_k} < -\delta_p, \\ 0, & |d_{t_k}| \leq \delta_p, \end{cases} \tag{4}$$

where $\delta_p > 0$ is a dead-zone threshold for small predictive changes. We set $p_{t_1} = 0$ when no preceding prefix is available and define signed relevance as $\zeta_{t_k} = c_{t_k} p_{t_k}$. Criticality captures the *strength* of return relevance over the completed trajectory, whereas polarity captures the *direction* of return-predictive progression. Accordingly, criticality determines state selection, while polarity modulates policy-side intervention.

Online relevance prediction. Because (c_t, p_t, ζ_t) become available only after episode completion, we train a lightweight online head H_η to predict signed relevance from the current latent state,

$\widehat{\zeta}_t = H_\eta(\bar{z}_t)$, using $\mathcal{L}_{\text{rel}} = \mathbb{E}[\text{Huber}(\widehat{\zeta}_t, \zeta_t)]$. The online prediction guides real-time exploration, while world-model learning and imagination directly use retrospective annotations stored and periodically refreshed in replay.

3.3 Return-Relevance Resource Allocation

Focused exploration. Let a_t be the action proposed by the original Dreamer collection policy. FocusDreamer adjusts local exploration according to the predicted signed relevance:

$$\alpha_E(t) = g_E(|\widehat{\zeta}_t|)\left[\lambda_E^+ \mathbb{I}(\widehat{\zeta}_t > 0) + \lambda_E^- \mathbb{I}(\widehat{\zeta}_t < 0)\right], \qquad \lambda_E^- > \lambda_E^+ \geq 0, \tag{5}$$

where $g_E : [0,1] \to [0,1]$ is a bounded monotonic relevance gate, and λ_E^+ and λ_E^- control the intervention strength for positive and negative progression, respectively. The executed action is $\widetilde{a}_t = \text{clip}(a_t + \alpha_E(t)\epsilon_t)$, with ϵ_t a bounded local action perturbation. Thus, exploration remains close to the original policy at weakly relevant states and becomes stronger around return-relevant states, with larger intervention for negative progression. The environment-interaction budget is unchanged.

Focused world-model learning. Let p_{std} denote Dreamer's standard replay-window distribution. For each annotated state, we sample an anchor according to $p_{\text{crit}}(t) \propto (c_t + \epsilon)^{\beta_M}$, where β_M controls prioritization strength, and construct a standard contiguous training window around the selected anchor. The induced criticality-focused window distribution is mixed with standard replay as

$$p_{\text{model}} = (1 - \rho_M)p_{\text{std}} + \rho_M p_{\text{crit}}, \qquad 0 \leq \rho_M \leq 1, \tag{6}$$

where ρ_M controls the fraction of focused model learning. The sampled windows are optimized using the original Dreamer world-model objective. Only criticality is used at this stage, so both favorable and unfavorable critical dynamics receive additional modeling support.

Focused imagination. Let q_{std} denote Dreamer's original distribution over posterior states used to initialize imagination. Each annotated state receives priority $I_t = c_t w_I(p_t)$, with $w_I(+1) = \lambda_I^+$, $w_I(-1) = \lambda_I^-$, and $\lambda_I^- > \lambda_I^+ > 0$; neutral states receive no additional focused priority. Normalizing I_t over eligible states yields q_{rel}, which is mixed with the standard start-state distribution:

$$q_{\text{imag}} = (1 - \rho_I)q_{\text{std}} + \rho_I q_{\text{rel}}, \qquad 0 \leq \rho_I \leq 1. \tag{7}$$

Here, ρ_I controls the fraction of relevance-focused imagination starts. Negative critical states are therefore revisited more frequently than positive ones at equal criticality. Once the start state is selected, FocusDreamer follows the original Dreamer latent rollout and actor–value optimization.

Together, focused exploration *acquires* alternative real experience, criticality-focused replay *models* its local consequences, and focused imagination *reuses* the learned behavioral distinctions for policy optimization. Criticality provides the common state-selection signal, while polarity asymmetrically modulates the two policy-side stages.

4 Theoretical Analysis

We present two main results corresponding to the motivation and design of FocusDreamer. Full derivations and additional guarantees for criticality and polarity are deferred to Appendix A.

Value-gap contraction. Let x_p and x_b denote the true future latent summaries induced by a familiar policy action and a weakly supported alternative, respectively. Under a local familiarity-attraction model $\hat{x}_b = (1 - \rho)x_b + \rho x_p + \epsilon_b$ and $\hat{x}_p = x_p + \epsilon_p$, with $0 \leq \rho < 1$, assume the local return is affine, $V(x) = w^\top x + b$.

Proposition 1 (Familiarity-induced value-gap contraction)**.** *For the true local gap $\Delta_P = V(x_b) - V(x_p)$, the imagined gap satisfies*

$$\widehat{\Delta} = (1 - \rho)\Delta_P + w^\top(\epsilon_b - \epsilon_p). \tag{8}$$

The factor $1-\rho$ directly contracts the advantage of the alternative, while the residual term can further reverse its ranking. This formalizes the failure examined empirically in Sec. 5.4 and motivates preserving local return differences rather than improving prediction uniformly.

Criticality-guided resource allocation. Let $J^*(V) = \mathbb{E}[J_\tau \mid V]$ denote the true conditional episodic return associated with the trajectory representation, and let $S_{\hat{J}}(t) = \|\nabla_{\bar{z}_t} \hat{J}(V)\|$ denote predictor sensitivity with respect to the detached relevance feature. Under locally stable attention and bounded, non-degenerate, non-canceling attention paths, attention rollout provides $S_{\hat{J}}(t) \geq \alpha c_t$ for some $\alpha > 0$. Further assume predictor-gradient fidelity

$$\|\nabla_{\bar{z}_t} J^*(V) - \nabla_{\bar{z}_t} \hat{J}(V)\| \leq \epsilon_J$$

and local action controllability

$$\|\nabla_a Q_P^H(z_t, a_\pi)\| \geq \kappa \|\nabla_{\bar{z}_t} J^*(V)\|.$$

Define the best ε-local improvement as

$$G_\varepsilon(z_t) = \max_{\|\delta a\| \leq \varepsilon} \left[Q_P^H(z_t, a_\pi + \delta a) - Q_P^H(z_t, a_\pi) \right].$$

Theorem 1 (Criticality-guided local improvement)**.** *If $Q_P^H(z_t, \cdot)$ is locally L-smooth, then*

$$G_\varepsilon(z_t) \geq \max \left\{ 0,\, \varepsilon\kappa(\alpha c_t - \epsilon_J) - \frac{L}{2}\varepsilon^2 \right\}. \tag{9}$$

The bound is non-decreasing in c_t, so higher criticality yields a larger certified lower bound on local improvement opportunity. This supports criticality as the common location signal for focused exploration, model learning, and imagination; additional selection and polarity guarantees are given in Appendix A.

5 Experiments

We evaluate FocusDreamer in terms of control performance, sample efficiency, component contribution, and mechanism validity. In addition to standard benchmark evaluation, we conduct controlled counterfactual analyses to examine three central aspects of the proposed approach: whether learned imagination contracts the return advantage of better local alternatives, whether return criticality identifies states with larger improvement opportunities, and whether FocusDreamer better preserves these behavioral distinctions in the learned world model.

5.1 Setup

We evaluate FocusDreamer on eight challenging visual continuous-control tasks from DMControl and four visual robotic-manipulation tasks from Meta-World. All agents are trained from pixels for 1M environment steps under matched interaction budgets; Dreamer-based methods additionally use matched model-update and imagination budgets when implementation permits. We compare against DrQ-v2 (Yarats et al., 2022), a strong data-augmented model-free baseline; TACO (Zheng et al., 2023), which improves visual control through task-relevant representation learning; DreamerPro, DreamerV3, and R2-Dreamer as representative world-model approaches; TD-MPC2 as a strong latent model-based control method; and Knowledge Boundary (KB), which allocates exploration and imagination using model uncertainty. Complete implementation details, per-task configurations, additional benchmarks, and extended hyperparameter analyses are provided in the appendix.

5.2 Overall Performance

Table 1 shows that FocusDreamer achieves the best or tied-best performance on six of eight tasks and ranks second on the remaining two, indicating that its improvement is not driven by a small subset of environments. Its average return reaches 778.0, substantially exceeding both KB (686.9) and R2-Dreamer (684.0), while improving the DreamerV3 backbone from 508.1 to 778.0. The largest gains over DreamerV3 appear on Quadruped Run, Quadruped Walk, Acrobot Swingup, and Reacher Hard, where standard imagination leaves considerable room for better local behavioral choices. This pattern is consistent with the design of FocusDreamer: return relevance directs local exploration toward consequential states, increases model learning on the resulting sparse experience, and reuses these states more frequently during policy optimization. The strong results across both

Table 1: Return on eight challenging visual DMControl tasks after 1M environment steps.

DMControl	DrQ-v2	TACO	DreamerPro	TD-MPC2	DreamerV3	R2-Dreamer	KB	FocusDreamer
Quadruped Run	407	541	616	362	331	490	684	**877**
Hopper Hop	189	261	253	211	369	363	306	**420**
Walker Run	517	637	750	728	765	707	718	**837**
Quadruped Walk	680	793	677	253	353	866	871	**902**
Cheetah Run	691	821	**898**	550	728	881	853	**898**
Finger Turn Hard	220	632	720	872	810	**930**	842	916
Acrobot Swingup	128	241	438	216	210	266	357	**451**
Reacher Hard	572	883	294	877	499	**969**	864	923
Average	425.5	601.1	580.8	508.6	508.1	684.0	686.9	**778.0**

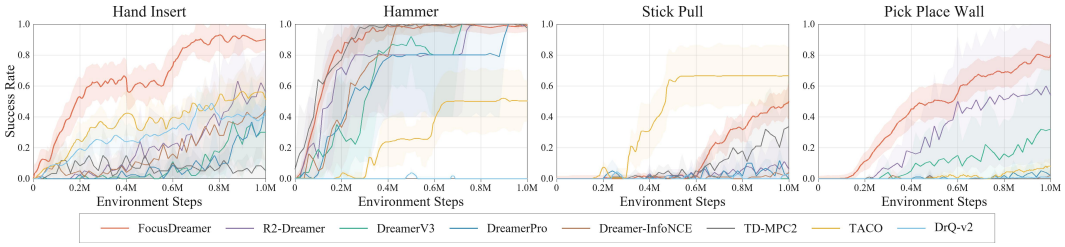

Figure 3: Success-rate learning curves on four visual Meta-World tasks.

locomotion and control tasks therefore suggest that selectively allocating a fixed learning budget is more effective than applying model learning and imagination uniformly.

Figure 3 further evaluates whether the same allocation principle transfers from continuous locomotion to robotic manipulation. FocusDreamer achieves the best or tied-best final performance among the model-based methods across all four tasks and shows faster policy improvement on several tasks. On Stick Pull, TACO attains a higher final success rate; as a model-free method based on temporal contrastive encoding, it may benefit from a different representation bias that is particularly effective on this task. Overall, the results indicate that trajectory-return relevance transfers effectively to manipulation settings, where concentrating exploration, model learning, and imagination on return-relevant contexts helps preserve useful local behavioral distinctions and supports continued policy improvement.

5.3 Ablation Study

Table 2 examines the cumulative contribution of the three allocation stages. Random allocation remains close to DreamerV3 (453.3 vs. 450.3), showing that simple budget redistribution offers little benefit. Return relevance guided exploration raises the average return to 593.3 by concentrating local search on consequential states. Focused world-model learning further improves performance to 683.3 by increasing updates on these sparse but important local dynamics, while focused imagination reaches 748.0 by exposing them more frequently to actor and value optimization. These gains support the complementary acquire, model, and reuse stages of FocusDreamer.

The second group isolates the effect of polarity. Removing polarity reduces the average return to 698.0, while the positive-only and negative-only variants reach 711.3 and 732.7, respectively. The stronger negative-polarity result is consistent with its larger local improvement opportunity, supporting stronger corrective intervention at unfavorable critical states. The full model further improves to 748.0, showing that positive critical states also help discover better policy choices beyond correcting unfavorable behavior. Overall, criticality identifies where learning resources are most useful, while polarity provides complementary directional guidance for policy optimization.

Table 2: Component and signal ablations on representative DMControl tasks. All variants use matched interaction and update budgets.

Method	Quadruped Run	Finger Hard	Acrobot	Avg.
DreamerV3	331	810	210	450.3
Random Allocation	338	806	216	453.3
Focused Exploration	606	850	324	593.3
Focused Exploration + Focused Model Learning	764	888	398	683.3
Full FocusDreamer	877	916	451	748.0
w/o Polarity	803	889	402	698.0
Positive Polarity Only	819	898	417	711.3
Negative Polarity Only	850	909	439	732.7

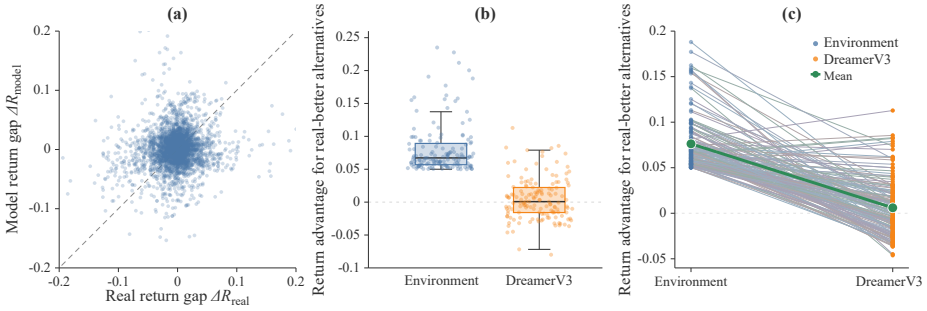

Figure 4: Evidence of value-gap contraction in learned world models. (a) Real and imagined return gaps for sampled local alternatives. (b) Distribution of return advantages for alternatives that are better in the real environment. (c) Paired comparison showing the contraction of alternative advantages from the environment to the world model.

5.4 Mechanism Validation

We further examine whether FocusDreamer preserves local behavioral value distinctions. For each DMControl state s_t, we restore the exact simulator state and compare the actor action a_π with $N = 32$ locally perturbed alternatives. Each action is evaluated by an $H = 15$ step rollout in both the real environment and the learned world model using the same subsequent action sequence. We then compute return advantages relative to a_π and aggregate results within each state and across seeds.

For candidate action a_i, define

$$\Delta R_{\text{real}} = R^H_{\text{real}}(s_t, a_i) - R^H_{\text{real}}(s_t, a_\pi), \qquad \Delta R_{\text{model}} = R^H_{\text{model}}(z_t, a_i) - R^H_{\text{model}}(z_t, a_\pi), \quad (10)$$

and define the sampled local improvement opportunity as

$$G_N(s_t) = \max_{a_i \in \mathcal{A}_N(s_t)} \Delta R_{\text{real}}(a_i), \quad (11)$$

where $\mathcal{A}_N(s_t)$ is the set of sampled local alternatives. We regard $\Delta R_{\text{real}} > m$ as a meaningful real improvement and measure the better-alternative miss rate by $\Pr(\Delta R_{\text{model}} \le 0 \mid \Delta R_{\text{real}} > m)$.

Value-gap contraction. Figure 4 examines the discrepancy between real and imagined behavioral advantages from three complementary views. Figure 4(a) compares real and imagined return gaps across locally sampled alternatives, showing that distinctions evident in the environment are substantially compressed in the learned world model. Restricting the analysis to alternatives that are genuinely better in the real environment, Fig. 4(b) shows a pronounced downward shift in their return-advantage distribution, with many imagined advantages approaching or crossing zero. The paired comparison in Fig. 4(c) further shows that this reduction occurs consistently at the level of individual alternatives rather than arising only from aggregate statistics. Consequently, locally better actions can become weakly distinguished or even misranked in imagination. Since Dreamer

Table 3: Mechanism analysis of FocusDreamer. Left: sampled local improvement opportunity under different selection signals and polarity groups. Right: value-gap error and better-alternative miss rate across FocusDreamer criticality quartiles, where c-Q1–Q4 denote increasing criticality. Lower is better for both metrics in the right panel.

(a) Opportunity Localization

Selection Signal	Q1	Q2	Q3	Q4
Random	0.059	0.062	0.060	0.063
Uncertainty	0.047	0.055	0.064	0.078
Criticality	0.028	0.043	0.065	0.108

High-Criticality States

Polarity Group	Mean G_N	Relative Gain
Positive	0.094	1.00×
Negative	0.121	1.29×

(b) Value-Gap Recovery

Method	c-Q1	c-Q2	c-Q3	c-Q4
Value-Gap Error ↓				
DreamerV3	0.063	0.071	0.076	0.093
KB	0.057	0.061	0.060	0.067
FocusDreamer	0.052	0.049	0.043	0.038
Better-Alternative Miss Rate (%) ↓				
DreamerV3	44.0	47.5	49.0	56.5
KB	41.0	42.5	41.5	44.0
FocusDreamer	37.5	34.0	29.5	24.0

optimizes its actor using imagined returns, such contraction can weaken the optimization signal associated with better but insufficiently supported alternatives. Together, these results provide direct empirical evidence for the alternative value-gap contraction analyzed in Sec. 4.

Localization of improvement opportunity. Table 3(a) evaluates how well different signals localize states with larger real improvement opportunity. We first construct a common state pool from DreamerV3 trajectories and compute G_N for each state relative to the DreamerV3 actor action. The same states are then ranked independently by a random permutation, ensemble uncertainty, and FocusDreamer criticality, and each ranking is divided into equal-size quartiles. Random partitioning yields nearly unchanged G_N across quartiles, while uncertainty provides a moderate ordering. Criticality shows a much stronger separation, with G_N increasing from 0.028 in its lowest quartile to 0.108 in its highest quartile. This indicates that criticality is more effective at locating states where nearby actions offer meaningful real policy improvement. Within the high-criticality subset, negative states exhibit 29% larger mean G_N than positive states (0.121 vs. 0.094), supporting stronger corrective intervention for negative progression while retaining positive states for discovering further policy improvements.

Recovery of behavioral value gaps. Table 3(b) then asks whether these consequential behavioral distinctions are preserved by the learned world model. Here the common evaluation states are partitioned once using FocusDreamer criticality, and the resulting c-quartiles are shared by DreamerV3, KB, and FocusDreamer. All methods are evaluated on exactly the same restored states, candidate actions, and continuations. DreamerV3 shows substantial value-gap distortion in the high-criticality subset, reaching an error of 0.093 and a better-alternative miss rate of 56.5% in Q4. KB reduces these values to 0.067 and 44.0%, while FocusDreamer further lowers them to 0.038 and 24.0%. Thus, the advantage of FocusDreamer is largest precisely in the states that criticality identifies as behaviorally consequential, consistent with its targeted allocation of exploration, model learning, and imagination. Additional sensitivity analyses of ρ_M, ρ_I, and λ^-/λ^+ are provided in Appendix F.

6 Conclusion

FocusDreamer addresses a policy-relevant failure in world-model reinforcement learning, where weakly supported local alternatives may lose their real return advantages in imagination. It learns state-level return relevance from completed trajectories to focus exploration, world-model learning, and imagination on states where behavioral distinctions matter more for return. Criticality determines where learning matters, while polarity modulates policy-side intervention. Experiments on visual DMControl and Meta-World, together with theoretical and counterfactual analyses, demonstrate improved performance, sample efficiency, and preservation of policy-relevant return differences. These results support return-directed resource allocation as an effective principle for world-model reinforcement learning under limited interaction and computation.

AI USE STATEMENT

In this work, we used generative AI tools to assist with manuscript writing and language polishing. The authors reviewed and verified all AI-assisted content and take full responsibility for the final manuscript, including its technical claims, mathematical expressions, experimental results, and citations.

ETHICS STATEMENT

This work does not involve human subjects or personally identifiable data.

REPRODUCIBILITY STATEMENT

Key implementation details, hyperparameters, theoretical derivations, and evaluation protocols are provided in the appendix. Detailed implementation configurations and source code are released to facilitate reproduction of the reported results.

REFERENCES

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197, July 2020.

Guillaume Alain and Yoshua Bengio. What regularized auto-encoders learn from the data-generating distribution. *Journal of Machine Learning Research*, 15(110):3743–3773, 2014.

Jose A. Arjona-Medina, Michael Gillhofer, Michael Widrich, Thomas Unterthiner, Johannes Brandstetter, and Sepp Hochreiter. RUDDER: Return decomposition for delayed rewards. In *Proceedings of the Advances in Neural Information Processing Systems*, volume 32, 2019.

Julia Berger, Bernd Frauenknecht, Sebastian Trimpe, and Bastian Leibe. Biased dreams: Limitations to epistemic uncertainty quantification in latent dynamics models. *Reinforcement Learning Journal*, 7, 2026.

Maxime Burchi and Radu Timofte. Mudreamer: Learning predictive world models without reconstruction, 2024. URL https://arxiv.org/abs/2405.15083.

Maxime Burchi and Radu Timofte. Learning transformer-based world models with contrastive predictive coding. In *International Conference on Learning Representations*, pp. 6949–6970, 2025.

Fei Deng, Ingook Jang, and Sungjin Ahn. DreamerPro: Reconstruction-free model-based reinforcement learning with prototypical representations. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 4956–4975, 17–23 Jul 2022.

Yuanlin Duan, Wensen Mao, and He Zhu. Learning world models for unconstrained goal navigation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 59236–59265, 2024.

Marco Fumero, Luca Moschella, Emanuele Rodolà, and Francesco Locatello. Navigating the latent space dynamics of neural models. In *International Conference on Learning Representations*, pp. 24765–24793, 2026.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 2555–2565, 09–15 Jun 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to Control: Learning Behaviors by Latent Imagination. In *International Conference on Learning Representations*, 2020.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8081):647–653, 2025.

Nick Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, pp. 47376–47405, 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*, volume 32, 2019.

Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. Morel: Model-based offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21810–21823, 2020.

Haoxin Lin, Hongqiu Wu, Jiaji Zhang, Yihao Sun, Junyin Ye, and Yang Yu. Episodic return decomposition by difference of implicitly assigned sub-trajectory reward. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 13808–13816, 2024. doi: 10.1609/aaai.v38i12.29287.

Zhenxian Liu, Peixi Peng, Yangru Huang, and Yonghong Tian. Perceiving the knowledge boundary: Uncertainty-guided exploration and imagination for world models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 23990–23998, 2026. doi: 10.1609/aaai.v40i28.39576.

Haoyu Ma, Jialong Wu, Ningya Feng, Chenjun Xiao, Dong Li, Jianye Hao, Jianmin Wang, and Mingsheng Long. HarmonyDream: Task harmonization inside world models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 33983–34007, 21–27 Jul 2024.

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*, 2026.

Hsiao-Ru Pan and Bernhard Schölkopf. Skill or luck? return decomposition via advantage functions. In *The Twelfth International Conference on Learning Representations (ICLR)*, May 2024.

Yuan Pu, Yazhe Niu, Zhenjie Yang, Jiyuan Ren, Hongsheng Li, and Yu Liu. Unizero: Generalized and efficient planning with scalable latent world models. *Transactions on Machine Learning Research*, 2025(5), May 2025. ISSN 2835-8856.

Zhizhou Ren, Ruihan Guo, Yuan Zhou, and Jian Peng. Learning long-term reward redistribution via randomized return decomposition. In *Proceedings of the International Conference on Learning Representations*, 2022.

Marc Rigter, Minqi Jiang, and Ingmar Posner. Reward-free curricula for training robust world models. In *International Conference on Learning Representations*, pp. 2369–2396, 2024.

Cansu Sancaktar, Christian Gumbsch, Andrii Zadaianchuk, Pavel Kolev, and Georg Martius. SENSEI: Semantic exploration guided by foundation models to learn versatile world models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 52745–52777, 13–19 Jul 2025.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized Experience Replay. In *International Conference on Learning Representations*, 2016.

Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 8583–8592, 13–18 Jul 2020.

Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, Nicolas Heess, and Yuval Tassa. dmcontrol: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020. ISSN 2665-9638.

Claas A Voelcker, Victor Liao, Animesh Garg, and Amir massoud Farahmand. Value gradient weighted model-based reinforcement learning. In *International Conference on Learning Representations*, 2022.

Claas A Voelcker, Anastasiia Pedan, Arash Ahmadian, Romina Abachi, Igor Gilitschenski, and Amir-Massoud Farahmand. Calibrated value-aware model learning with probabilistic environment models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 61745–61768, 13–19 Jul 2025.

Shengjie Wang, Shaohuai Liu, Weirui Ye, Jiacheng You, and Yang Gao. EfficientZero v2: Mastering discrete and continuous control with limited data. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 51041–51062, 21–27 Jul 2024.

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

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-World: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Proceedings of the Conference on Robot Learning*, volume 100, pp. 1094–1100, 30 Oct–01 Nov 2020a.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. In *Advances in Neural Information Processing Systems*, volume 33, pp. 14129–14142, 2020b.

Weipu Zhang, Gang Wang, Jian Sun, Yetian Yuan, and Gao Huang. Storm: Efficient stochastic transformer based world models for reinforcement learning. In *Neural Information Processing Systems*, 2023.

Ruijie Zheng, Xiyao Wang, Yanchao Sun, Shuang Ma, Jieyu Zhao, Huazhe Xu, Hal Daumé, III, and Furong Huang. TACO: Temporal latent action-driven contrastive loss for visual reinforcement learning. In *Proceedings of the Advances in Neural Information Processing Systems*, volume 36, pp. 48203–48225, 2023.

Gaoyue Zhou, Hengkai Pan, Yann Lecun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 79115–79135, 13–19 Jul 2025.

A Additional Theoretical Analysis and Proofs

This section provides the derivations omitted from Sec. 4. The main paper states only the assumptions and results needed for the argument; the proofs below make explicit the local regularity, approximation, and non-cancellation conditions under which the guarantees hold.

A.1 Proof of Proposition 1

For completeness, write the local familiarity-attraction model as

$$\hat{x}_b = (1-\rho)x_b + \rho x_p + \epsilon_b, \qquad \hat{x}_p = x_p + \epsilon_p, \qquad 0 \le \rho < 1. \tag{12}$$

Using $V(x) = w^\top x + b$,

$$\begin{aligned}
\widehat{\Delta} &= w^\top(\hat{x}_b - \hat{x}_p) && (13)\\
&= w^\top((1-\rho)x_b + \rho x_p + \epsilon_b - x_p - \epsilon_p) && (14)\\
&= (1-\rho)w^\top(x_b - x_p) + w^\top(\epsilon_b - \epsilon_p) && (15)\\
&= (1-\rho)\Delta_P + w^\top(\epsilon_b - \epsilon_p). && (16)
\end{aligned}$$

When the residual term vanishes, the gap is exactly multiplied by $1-\rho$. For a truly better alternative $\Delta_P > 0$, the imagined ordering reverses whenever

$$w^\top(\epsilon_b - \epsilon_p) \le -(1-\rho)\Delta_P. \tag{17}$$

Thus attraction alone contracts the gap, while residual error determines whether the contracted gap remains positive or crosses zero.

A.2 General Better-Alternative Misranking Condition

Let a_p be the current action, a^* a truly better local action, and

$$G_c = Q_P^H(z_c, a^*) - Q_P^H(z_c, a_p) > 0. \tag{18}$$

Define signed imagined-return error

$$b_n(z,a) = \widehat{Q}_n^H(z,a) - Q_P^H(z,a). \tag{19}$$

Then

$$\widehat{Q}_n^H(z_c,a_p) - \widehat{Q}_n^H(z_c,a^*) = -G_c + b_n(z_c,a_p) - b_n(z_c,a^*). \tag{20}$$

Hence the true ranking is reversed whenever

$$b_n(z_c,a_p) - b_n(z_c,a^*) > G_c. \tag{21}$$

Conversely,

$$\widehat{Q}_n^H(z_c,a^*) - \widehat{Q}_n^H(z_c,a_p) \ge G_c - |b_n(z_c,a^*)| - |b_n(z_c,a_p)|, \tag{22}$$

so the true ordering is preserved whenever the sum of absolute errors is smaller than the true action gap. This condition clarifies why a global model-error scalar is insufficient: policy improvement depends on error relative to the behavioral gap being compared.

A.3 Attention-Flow Sensitivity

Lemma 1 (Attention-flow sensitivity)**.** *Under locally stable attention, bounded non-degenerate path Jacobians, and non-canceling active path contributions, there exist constants $0 < \alpha \le \beta$ such that*

$$\alpha c_t \le \|\nabla_{\bar{z}_t} \hat{J}(V)\| \le \beta c_t. \tag{23}$$

We make the structural assumptions used by the attention-flow argument explicit. In a neighborhood of the evaluated trajectory, (i) attention matrices are locally stable, so their first-order variation can be ignored; (ii) token projections, residual paths, and the return head have bounded Jacobian operators; and (iii) active path contributions from token t to the global token do not cancel in aggregate. These assumptions are local analytical conditions and do not imply that attention is a causal attribution mechanism.

Under frozen-attention local linearization, the `[CLS]` output is a sum of contributions transmitted through active attention paths. Let $\mathcal{P}_t$ be the set of paths from token t to `[CLS]`, $w_p \ge 0$ the product of row-normalized attention coefficients along path p, and J_p the associated local value/Jacobian operator after token projection and the return head. Then

$$\nabla_{\bar{z}_t} \hat{J} = \sum_{p \in \mathcal{P}_t} w_p J_p. \tag{24}$$

Attention rollout accumulates the same path products; after normalization, $\sum_{p\in\mathcal{P}_t} w_p$ is proportional to c_t. If $\|J_p\| \le \bar{m}$ for every active path,

$$\|\nabla_{\bar{z}_t} \hat{J}(V)\| \le \bar{m} \sum_{p \in \mathcal{P}_t} w_p = \beta c_t, \tag{25}$$

where β absorbs normalization and projection constants. For the lower bound, assume there exists $\underline{m} > 0$ such that active path operators are non-degenerate and satisfy the aggregate non-cancellation condition

$$\left\| \sum_{p \in \mathcal{P}_t} w_p J_p \right\| \ge \underline{m} \sum_{p \in \mathcal{P}_t} w_p. \tag{26}$$

Then

$$\|\nabla_{\bar{z}_t} \hat{J}(V)\| \ge \alpha c_t \tag{27}$$

for an appropriate $\alpha > 0$, proving Lemma 1.

A.4 From Learned to True Return Sensitivity

Recall that $J^*(V) = \mathbb{E}[J_\tau \mid V]$ denotes the true conditional episodic return associated with the trajectory representation. Assume the return predictor satisfies the local gradient-fidelity condition

$$\|\nabla_{\bar{z}_t} J^*(V) - \nabla_{\bar{z}_t} \hat{J}(V)\| \le \epsilon_J. \tag{28}$$

The reverse triangle inequality then gives

$$\|\nabla_{\bar{z}_t} J^*(V)\| \ge \|\nabla_{\bar{z}_t} \hat{J}(V)\| - \|\nabla_{\bar{z}_t} J^*(V) - \nabla_{\bar{z}_t} \hat{J}(V)\| \tag{29}$$

$$\ge \alpha c_t - \epsilon_J. \tag{30}$$

Thus, whenever $\alpha c_t > \epsilon_J$, criticality provides a positive lower bound on the sensitivity of the true conditional return under the assumed predictor-gradient fidelity. To connect latent return sensitivity to local action improvement, we further assume

$$\|\nabla_a Q_P^H(z_t, a_\pi)\| \ge \kappa \|\nabla_{\bar{z}_t} J^*(V)\|, \tag{31}$$

for some local controllability constant $\kappa > 0$.

A.5 Proof of Theorem 1

Let $g_a = \nabla_a Q_P^H(z_t, a_\pi)$. Local L-smoothness gives, for any $\|\delta a\| \le \varepsilon$,

$$Q_P^H(z_t, a_\pi + \delta a) \ge Q_P^H(z_t, a_\pi) + g_a^\top \delta a - \frac{L}{2}\|\delta a\|^2. \tag{32}$$

For $g_a \neq 0$, choose $\delta a = \varepsilon g_a / \|g_a\|$. Then

$$G_\varepsilon(z_t) \ge \varepsilon \|g_a\| - \frac{L}{2}\varepsilon^2. \tag{33}$$

Using Eq. equation 31 and the sensitivity bound above,

$$\|g_a\| \ge \kappa \|\nabla_{\bar{z}_t} J^*(V)\| \tag{34}$$

$$\ge \kappa(\alpha c_t - \epsilon_J). \tag{35}$$

Because $\delta a = 0$ is feasible in the definition of G_ε, we also have $G_\varepsilon(z_t) \ge 0$. Combining the two bounds yields

$$G_\varepsilon(z_t) \ge \max\left\{0,\, \varepsilon\kappa(\alpha c_t - \epsilon_J) - \frac{L}{2}\varepsilon^2\right\} =: B(c_t). \tag{36}$$

The result is a lower bound on the best local improvement under the stated regularity and controllability conditions; it does not impose a deterministic ordering on realized sampled improvements across arbitrary states.

A.6 Criticality-Priority Selection

Corollary 1 (Criticality-priority selection)**.** *For any replay-state set $\mathcal{S}$ and budget M, a top-M criticality subset $\mathcal{C}_M$ satisfies*

$$\mathcal{C}_M \in \arg\max_{\substack{\mathcal{C}\subseteq\mathcal{S} \\ |\mathcal{C}|=M}} \sum_{t\in\mathcal{C}} B(c_t). \tag{37}$$

Moreover, $c_t \ge \tau$ implies $G_\varepsilon(z_t) \ge B(\tau)$, and this certified bound is strictly positive whenever

$$\tau > \frac{\epsilon_J}{\alpha} + \frac{L\varepsilon}{2\alpha\kappa}. \tag{38}$$

By Eq. equation 36, $B(c)$ is non-decreasing in c and is strictly increasing once its affine branch is positive. Hence replacing a selected state by an unselected state with larger criticality cannot decrease the summed certified bound, which proves Eq. equation 37 up to possible ties in the zero-bound region. If $c_t \ge \tau$, monotonicity gives $G_\varepsilon(z_t) \ge B(\tau)$; solving $B(\tau) > 0$ yields Eq. equation 38.

A.7 Online Relevance-Selection Guarantee

For non-neutral polarity $p_t \in \{-1, +1\}$, $|\zeta_t| = c_t$. If the online signed-relevance head satisfies

$$|\hat{\zeta}_t - \zeta_t| \le \epsilon_H, \tag{39}$$

then the reverse triangle inequality gives

$$\big|\, |\hat{\zeta}_t| - c_t \,\big| \le |\hat{\zeta}_t - \zeta_t| \le \epsilon_H. \tag{40}$$

Therefore $|\hat{\zeta}_t| \ge \tau$ implies $c_t \ge \tau - \epsilon_H$ and hence

$$G_\varepsilon(z_t) \ge B(\tau - \epsilon_H). \tag{41}$$

For two states, a predicted-magnitude difference larger than $2\epsilon_H$ preserves their true criticality ordering. Moreover, if $c_t > \epsilon_H$, a sign reversal would require approximation error at least c_t, so $\operatorname{sign}(\hat{\zeta}_t) = p_t$. Neutral states require separate treatment because $|\zeta_t| = 0$ does not encode criticality.

A.8 Probability of Discovering a Better Local Alternative

In the local first-order regime,

$$\Delta Q_P^H = g_a^\top \delta a + \mathcal{O}(\|\delta a\|^2). \tag{42}$$

For an isotropic local perturbation $\delta a \sim \mathcal{N}(0, \sigma_a^2 I)$, the linear term is Gaussian with variance $\sigma_a^2 \|g_a\|^2$. For a positive improvement threshold $m > 0$,

$$\Pr[\Delta Q_P^H > m] \approx 1 - \Phi\left(\frac{m}{\sigma_a \|g_a\|}\right). \tag{43}$$

Combining this expression with $\|g_a\| \ge \kappa(\alpha c_t - \epsilon_J)$ gives the monotone approximation

$$\Pr[\Delta Q_P^H > m] \gtrsim 1 - \Phi\left(\frac{m}{\sigma_a \kappa(\alpha c_t - \epsilon_J)}\right) \tag{44}$$

whenever the denominator is positive and higher-order terms are negligible. The use of $m > 0$ is essential: for a symmetric first-order perturbation and zero threshold, the probability is approximately one half irrespective of gradient magnitude.

A.9 Polarity and Conditional-Return Progression

Let

$$J_k^* = \mathbb{E}[J_\tau \mid V_{1:k}]$$

denote the true conditional episodic return given trajectory prefix $V_{1:k}$. Suppose the prefix return predictor satisfies

$$\left|\hat{J}(V_{1:k}) - J_k^*\right| \le \epsilon_P$$

for adjacent prefixes.

Proposition 2 (Negative polarity indicates downward progression)**.** *If $d_{t_k} < -\delta_p$ and $\delta_p > 2\epsilon_P$, then*

$$J_k^* - J_{k-1}^* < -(\delta_p - 2\epsilon_P) < 0. \tag{45}$$

Proof. Let

$$e_k = \hat{J}(V_{1:k}) - J_k^*, \qquad |e_k| \le \epsilon_P. \tag{46}$$

Since $d_{t_k} = \hat{J}(V_{1:k}) - \hat{J}(V_{1:k-1})$,

$$J_k^* - J_{k-1}^* = d_{t_k} - e_k + e_{k-1}. \tag{47}$$

If $d_{t_k} < -\delta_p$, then

$$J_k^* - J_{k-1}^* < -\delta_p + 2\epsilon_P = -(\delta_p - 2\epsilon_P) < 0. \tag{48}$$

The positive case follows symmetrically. □

This result establishes only the direction of conditional-return progression. It does not imply that negative polarity causes a poor outcome or guarantees the existence of a better local action.

A.10 Local Ranking Preservation under Focused Error Reduction

Let $\mathcal{A}$ be a finite local candidate set, $a^* = \arg\max_{a \in \mathcal{A}} Q_P^H(z, a)$, and

$$\Delta_{\min} = \min_{a \neq a^*} \left[Q_P^H(z, a^*) - Q_P^H(z, a) \right] > 0. \tag{49}$$

If for every competitor $a \neq a^*$,

$$|b(z, a^*)| + |b(z, a)| < \Delta_{\min}, \tag{50}$$

then

$$\widehat{Q}^H(z, a^*) - \widehat{Q}^H(z, a) \geq \Delta_{\min} - |b(z, a^*)| - |b(z, a)| > 0, \tag{51}$$

so the true local best action remains top-ranked in imagination. Focused exploration and replay do not by themselves guarantee this error condition; rather, they are mechanisms for increasing support and optimization frequency at selected states so that the condition becomes easier to satisfy empirically.

A.11 Support and Focused-Update Effects

Let $\mu_n(z, a)$ denote replay support and $d_n(z, a)$ an action-evaluation distribution. For a nonnegative local error $e_n(z, a)$ and

$$C_n = \operatorname*{ess\,sup}_{z,a} \frac{d_n(z, a)}{\mu_n(z, a)}, \tag{52}$$

importance weighting gives

$$\mathbb{E}_{d_n}[e_n] \leq C_n \mathbb{E}_{\mu_n}[e_n]. \tag{53}$$

A large density ratio therefore weakens control of evaluation error in poorly supported regions. Focused exploration increases real support for nearby alternatives at selected states, while criticality-focused replay increases the expected number of model updates assigned to their local windows under a fixed update count. These observations complement the opportunity theorem: criticality specifies where additional accuracy has larger certified task relevance, whereas focused acquisition and replay improve the data and optimization support available there.

B Additional Benchmark: Atari100K

Atari100K provides a supplementary discrete-action setting that complements the continuous-control experiments in the main paper. The goal of this evaluation is not to reproduce the local continuous-action perturbations used in DMControl, but to examine whether trajectory-derived return relevance remains useful when each state provides a finite set of alternative actions. For the discrete-action variant, focused exploration increases sampling of return-relevant states and their alternative actions while keeping the total interaction budget fixed. Focused world-model learning and focused imagination follow the same relevance-based state selection principle as in the continuous-control setting.

B.1 Evaluation Protocol

We evaluate FocusDreamer on Atari100K under the standard 100K-interaction protocol. All methods use the same preprocessing, evaluation schedule, and number of seeds. FocusDreamer keeps the original backbone objectives and only changes how exploration, replay, and imagination are focused according to return relevance. We compare with EfficientZero (Wang et al., 2024), DreamerV3 (Hafner et al., 2025), MuDreamer (Burchi & Timofte, 2024), HarmonyDream (Ma et al., 2024), and Knowledge Boundary (Liu et al., 2026). We report the raw episodic returns for all methods.

FocusDreamer achieves competitive performance on Atari100K despite the restricted discrete action space. The improvement is more pronounced on tasks with complex temporal dependencies, such as Crazy Climber and Road Runner, where different action choices can lead to substantially different future outcomes. On tasks with more constrained behavioral alternatives, such as Amidar and Gopher, FocusDreamer remains comparable to existing allocation-based methods. This trend is consistent with the motivation of FocusDreamer: return relevance becomes more beneficial when preserving distinctions among local behaviors is important for long-term return optimization.

Table 4: Performance comparison on selected Atari100K tasks. Crazy, Road, and Up denote Crazy Climber, Road Runner, and Up N Down, respectively.

Method	**Amidar**	**Crazy**	**Gopher**	**Road**	**Up**
EfficientZero	102	80125	3518	18512	16096
DreamerV3	90	76093	2977	9972	11894
MuDreamer	114	80017	1347	9003	4766
HarmonyDream	107	83560	**4479**	15502	13361
Knowledge Boundary	118	86120	3360	18110	**18840**
FocusDreamer	**121**	**89574**	4325	**19887**	18260

Table 5: Fidelity of the online relevance predictor on held-out trajectory-derived targets. Top-25% denotes states with the highest retrospective criticality.

Task	MAE ↓	Sign Acc. ↑	Spearman ↑	Top-25% Sign Acc. ↑	Top-25% Spearman ↑
Quadruped Run	0.016	91.8%	0.82	96.3%	0.90
Finger Turn Hard	0.013	93.1%	0.85	97.0%	0.92
Acrobot Swingup	0.019	89.7%	0.79	94.8%	0.87
Average	0.016	91.5%	0.82	96.0%	0.90

C Relevance Predictor Fidelity

FocusDreamer's online focused exploration depends on the lightweight relevance predictor H_η, which estimates signed relevance before episode completion. This section directly evaluates whether the online predictor can faithfully approximate the retrospective trajectory-derived target $\zeta_t = c_t p_t$. The evaluation therefore isolates the reliability of the deployed relevance signal from its downstream control effect.

All results are computed on held-out completed episodes that are not used for updating H_η. We evaluate three complementary properties: (i) numerical fidelity through mean absolute error (MAE), (ii) directional fidelity through polarity sign accuracy, and (iii) ranking fidelity through Spearman correlation between predicted and retrospective relevance. Since FocusDreamer allocates additional resources primarily to highly consequential states, we additionally report the same directional and ranking metrics on the top 25% of states ranked by retrospective criticality.

For sign accuracy, very small-magnitude teacher targets are excluded using a fixed evaluation margin μ to avoid ambiguity near zero. This margin is used only for fidelity evaluation and does not modify the polarity dead zone δ_p used during training.

The relevance predictor achieves low numerical error while preserving both the direction and relative strength of trajectory-derived relevance. In particular, polarity accuracy exceeds 90% on average and further improves on high-criticality states, where FocusDreamer concentrates additional learning resources. The strong Spearman correlation also indicates that H_η preserves the relative ordering of relevance magnitude, supporting its use for online state selection before retrospective annotations become available.

D Return-Gap Preservation after FocusDreamer

We further examine whether FocusDreamer improves the preservation of local behavioral value differences in the learned world model. We apply the same frozen-policy counterfactual probe used in Sec. 5.4 to the FocusDreamer world model, using the same restored states, candidate actions, and branch continuations as in the main-paper analysis. This allows the difference between DreamerV3 and FocusDreamer to be attributed to the learned world model rather than to changes in the evaluated behaviors.

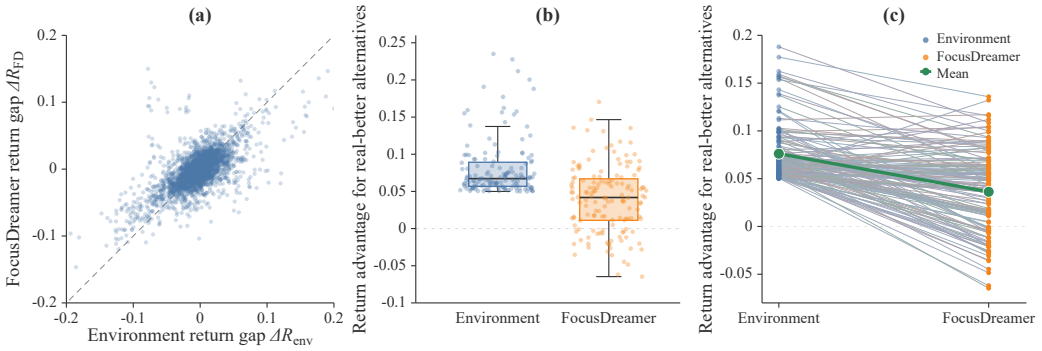

Figure 5: Return-gap preservation with FocusDreamer under the same counterfactual protocol as Fig. 4.

D.1 Counterfactual Evaluation Protocol

For each selected DMControl state, we save the complete simulator state and the corresponding posterior latent state. The deterministic actor action is used as the reference action, and $N = 32$ locally perturbed alternatives are evaluated over an $H = 15$ step horizon. For each candidate, the simulator is restored to the same state, the candidate action is applied at the first step, and the frozen actor then continues from real observations. The resulting branch-specific real action sequence is replayed through the learned world model from the saved posterior. This fixed-continuation design ensures that the comparison measures return-gap prediction rather than divergence caused by different imagined action sequences.

For candidate action a_i, we define

$$\Delta R_{\text{real}} = R^H_{\text{real}}(s_t, a_i) - R^H_{\text{real}}(s_t, a_\pi), \qquad \Delta R_{\text{FD}} = R^H_{\text{FD}}(z_t, a_i) - R^H_{\text{FD}}(z_t, a_\pi), \tag{54}$$

where ΔR_{FD} denotes the return gap predicted by the FocusDreamer world model.

Figure 5 visualizes the resulting gap preservation from three complementary perspectives. Panel (a) compares environment and FocusDreamer return gaps over all sampled local alternatives, where better alignment with the diagonal indicates more faithful preservation of behavioral differences. Panel (b) restricts the analysis to alternatives that are better in the real environment and compares their return-advantage distributions. Panel (c) pairs the environment and FocusDreamer gaps for the same real-better alternatives, making the degree of contraction or recovery visible at the individual-branch level.

D.2 Results

FocusDreamer substantially reduces the value-gap contraction observed with DreamerV3 while not eliminating model error entirely. For alternatives that are better in the real environment, the mean real return advantage is 0.0808. DreamerV3 reduces this value to only 0.0051 in imagination, whereas FocusDreamer preserves an average advantage of 0.0397. Thus, FocusDreamer recovers a substantial portion of the behavioral distinction that is lost by the original world model.

The same trend is reflected in prediction error and sign preservation. Averaged across the four criticality quartiles, the mean absolute value-gap error decreases from 0.0758 with DreamerV3 to 0.0455 with FocusDreamer, while the better-alternative miss rate decreases from 49.3% to 31.3%. Panel (a) therefore shows a noticeably stronger alignment between real and imagined gaps, although the remaining spread around the diagonal confirms that the learned model is still imperfect. In Panels (b) and (c), FocusDreamer retains substantially more positive advantage for real-better alternatives and fewer branches cross below zero.

These results complement the criticality-based localization analysis in the main paper. Criticality identifies states with larger local improvement opportunities, while focused exploration and world-model learning provide additional evidence around these states. The resulting world model therefore better preserves policy-relevant return differences rather than simply improving prediction uniformly over the state space.

Table 6: Comparison of alternative state-relevance signals under matched learning budgets. Return is averaged over Quadruped Run, Finger Turn Hard, and Acrobot Swingup, following the main ablation setting.

Relevance Source	**Return** ↑	**Spearman** ↑	**Top-**25% **Opp. Ratio** ↑
Random Allocation	453.3	0.01	1.03
Ensemble Uncertainty	624.7	0.28	1.28
RUDDER Contribution	651.8	0.41	1.40
RRD Proxy Reward	666.4	0.45	1.47
Prefix Change $\lvert d_t \rvert$	681.7	0.50	1.53
Last-Layer CLS Attention	707.6	0.57	1.61
Attention-Rollout Criticality	**748.0**	**0.72**	**1.77**

E Ablation of Relevance Acquisition

FocusDreamer uses attention-rollout criticality to identify states where limited learning resources are more valuable. We further examine whether this specific construction is more effective than alternative state-relevance signals. All variants retain the same exploration, world-model learning, and imagination pipeline and differ only in the nonnegative score used for state selection. The same prefix-based polarity is retained.

E.1 Alternative Relevance Signals

We compare several alternative signals for acquiring state relevance, including random allocation, ensemble world-model uncertainty, return decomposition, prefix return change, and direct attention. For RUDDER (Arjona-Medina et al., 2019), we use the magnitude of its redistributed return contribution as the location score; for RRD (Ren et al., 2022), we use the magnitude of its learned per-step proxy reward. These two variants adapt trajectory-return decomposition signals for state selection rather than reproducing their original control objectives. Prefix Change uses $|d_t|$, and Last-Layer CLS Attention directly uses the normalized CLS-to-token attention from the final layer of the return predictor. Attention-Rollout Criticality is the proposed score c_t, obtained by propagating attention across predictor layers.

Table 6 reports both downstream control performance and localization quality on the same three representative DMControl tasks used in the main ablation. *Return* is the average final return across these tasks. *Spearman* measures the rank correlation between each relevance score and the sampled local improvement opportunity G_N. *Top-*25% *Opp. Ratio* is the mean G_N of the highest-ranked 25% of states divided by the mean G_N over the complete evaluation pool, measuring how strongly each signal concentrates learning resources on states with larger local improvement opportunities.

Return-based signals consistently provide stronger localization than random allocation and uncertainty alone. RUDDER and RRD improve over uncertainty, showing that completed-trajectory return information is useful for identifying consequential states. Prefix return change further improves localization by using trajectory-level return prediction, while direct last-layer attention provides a stronger state-selection signal by exploiting the predictor's trajectory context.

Attention-rollout criticality achieves the strongest correlation with G_N and concentrates the highest-ranked states on approximately $1.77\times$ the average local improvement opportunity. It also yields the highest downstream return. The improvement over Last-Layer CLS Attention indicates that aggregating attention propagation across Transformer layers provides a more effective relevance signal than relying on the final attention map alone. Together, these results support attention-rollout criticality as the common location signal used throughout FocusDreamer.

F Additional Hyperparameter Analysis

We further examine the sensitivity of FocusDreamer to two groups of hyperparameters: those controlling how return relevance allocates learning resources, and those controlling how trajectory-level

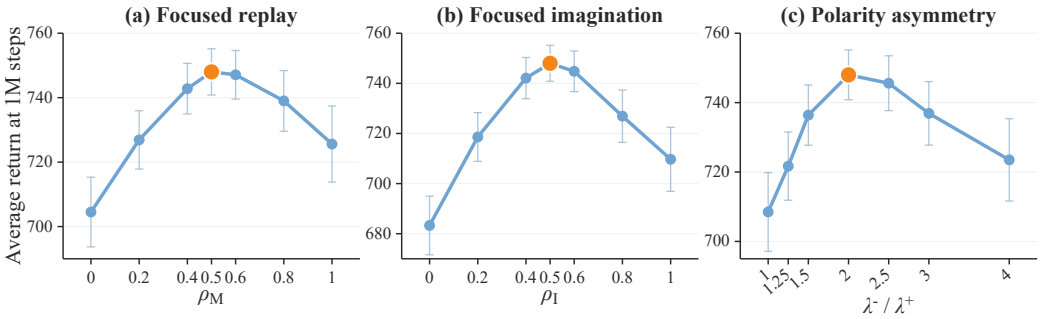

Figure 6: Sensitivity to the main FocusDreamer allocation hyperparameters. We vary (a) the focused replay ratio ρ_M, (b) the focused imagination ratio ρ_I, and (c) the shared negative-to-positive ratio λ^-/λ^+. Results report the average return over the three representative DMControl tasks at 1M environment steps.

return relevance is estimated. Unless otherwise specified, all experiments use the same three representative DMControl tasks as the main ablation, with all remaining training settings fixed.

F.1 Resource-Allocation Hyperparameters

We first examine whether FocusDreamer depends on a narrow choice of resource allocation strength. The focused replay ratio ρ_M controls the mixture between standard Dreamer replay and criticality-prioritized world-model updates, while ρ_I controls the corresponding mixture for imagination start states. The shared ratio $\lambda^-/\lambda^+ = \lambda_E^-/\lambda_E^+ = \lambda_I^-/\lambda_I^+$ controls the negative-to-positive asymmetry in both policy-side stages. When sweeping this ratio, the exploration and imagination ratios are varied jointly while all other settings are fixed. This experiment therefore tests whether the benefit comes from moderate relevance-guided allocation rather than from aggressively replacing the original Dreamer sampling distributions.

Figure 6 shows that FocusDreamer is robust over broad intermediate ranges of both ρ_M and ρ_I. Performance decreases when relevance-based allocation is either removed or made dominant, indicating that criticality is most effective as a soft focusing mechanism that complements standard Dreamer sampling. The polarity ratio exhibits a similar pattern. Moderate negative-to-positive asymmetry performs best, supporting stronger intervention at unfavorable progression while retaining positive critical states for further policy improvement.

F.2 Return-Relevance Predictor Configuration

We next test whether return-relevance estimation requires a sensitive or overly large Transformer. We vary three aspects of the trajectory-level return predictor: the number of attention heads, Transformer depth, and temporal sampling configuration. These experiments test, respectively, whether multi-head attention is necessary for criticality estimation, how much trajectory interaction capacity is required, and how temporal coverage should be balanced against redundancy. For each group, only the corresponding configuration is changed.

Attention configuration. Increasing the number of attention heads does not provide additional benefits. While four heads only cause a small degradation, eight heads lead to a larger performance drop. Since FocusDreamer extracts criticality through attention rollout, excessive head diversity may introduce less consistent relevance patterns. We therefore use a single attention head as the default.

Transformer depth. A single Transformer layer produces a noticeable performance drop, suggesting that shallow token interaction is insufficient for modeling trajectory-level return information. Increasing the depth to six layers provides only marginal improvement compared with three layers, indicating that additional model capacity is not critical for the return-relevance prediction objective. Therefore, we retain three layers as a more efficient default configuration.

Table 7: Sensitivity to the return-relevance predictor configuration at 1M environment steps. Results report the average return over the three representative DMControl tasks. The default configuration is highlighted in bold.

Parameter	Setting	Quadruped	Finger	Acrobot	Average
Attention heads	**1**	**877**	**916**	**451**	**748.0**
	4	872	910	445	742.3
	8	850	892	432	724.7
Transformer layers	1	863	902	439	734.7
	3	**877**	**916**	**451**	**748.0**
	6	881	919	452	750.7
Sampling configuration	$K = 32,\ h = 14$	868	908	443	739.7
	$\mathbf{K = 64,\ h = 7}$	**877**	**916**	**451**	**748.0**
	$K = 128,\ h = 3$	853	895	434	727.3

Temporal sampling. Temporal sampling exhibits a clearer coverage–redundancy trade-off. With $K = 32$ and $h = 14$, only a limited number of states represent the trajectory, which can omit information useful for episodic-return prediction. Increasing the sampling density to $K = 128$ and $h = 3$ introduces many temporally adjacent and therefore redundant states, producing a larger performance decrease. The default $K = 64, h = 7$ spans approximately 441 steps of a 500-step episode, providing broad trajectory coverage while maintaining sufficient temporal separation between sampled states.

Overall, these results show that FocusDreamer does not rely on a highly tuned or high-capacity return predictor. A lightweight configuration with one attention head, three Transformer layers, and moderately sparse trajectory sampling provides a favorable balance between return-relevance estimation, temporal coverage, and computational efficiency.

G Implementation Details and Computational Cost

This section reports the key implementation settings and computational overhead of FocusDreamer. Detailed implementation configurations are provided in the released code. FocusDreamer uses the same DreamerV3 backbone, interaction budget, and base optimization objectives as the baseline. The additional components are limited to trajectory-level return-relevance estimation, online relevance prediction, and relevance-guided sampling.

G.1 Training and Method Settings

Table 8 summarizes the backbone and FocusDreamer-specific configurations. Unless explicitly stated otherwise, FocusDreamer inherits the corresponding DreamerV3 setting without modification.

The trajectory-level return predictor operates on detached RSSM posterior features, so its optimization does not modify the Dreamer representation. Criticality is obtained by attention rollout over the trajectory predictor, while polarity is obtained from adjacent prefix return-prediction changes. The lightweight online relevance head is used only during real interaction; focused world-model learning and focused imagination directly use retrospective replay annotations.

Importantly, FocusDreamer does not introduce new world-model, actor, or value objectives. Focused world-model learning changes only the replay-window distribution, and focused imagination changes only the distribution of imagination start states. The total environment-interaction budget and the base DreamerV3 optimization objectives therefore remain unchanged.

G.2 Hardware

All experiments were run on a desktop machine equipped with an Intel Core i7 (14th generation) processor, 64 GB system memory, and a single NVIDIA RTX 5090D GPU with 32 GB memory.

Table 8: Key implementation settings of DreamerV3 and FocusDreamer. Detailed implementation is provided in the released code.

Setting	**DreamerV3**	**FocusDreamer**
Environment steps	1M	1M
Replay capacity	5×10^6	5×10^6
Batch size	16	16
Batch length	64	64
Optimizer	LaProp	LaProp
Learning rate	4×10^{-5}	4×10^{-5}
Imagination horizon	15	15
Discount factor	0.99	0.99
Trajectory tokens	–	64
Sampling interval h	–	7
Return predictor layers	–	3
Attention heads	–	1
Transformer hidden dimension	–	128
Return target	–	Undiscounted episodic return
Criticality extraction	–	Attention rollout
Polarity estimation	–	Prefix return change
Polarity dead zone δ_p	–	0.05
Online relevance head	–	LN-Linear(128)-SiLU-Linear(1)-tanh
Focused replay ratio ρ_M	–	0.5
Focused imagination ratio ρ_I	–	0.5
Polarity intervention ratio	–	$\lambda^-/\lambda^+ = 2$
Exploration warm-up	–	10,000 updates

Table 9: Computational cost under matched hardware and training budgets.

Method	**Time (1M Steps)**	**Peak VRAM**	**Params.**
DreamerV3	∼9.4 h	∼26.2 GB	41.659 M
FocusDreamer	∼10.6 h	∼27.9 GB	43.891 M

DreamerV3 and FocusDreamer were evaluated on the same hardware with identical environment-interaction budgets, evaluation frequency, and logging configuration.

G.3 Computational Cost

FocusDreamer introduces additional computation primarily through periodic trajectory-level return prediction and relevance annotation. These operations are performed on sampled completed trajectories rather than at every environment step. During online interaction, only the lightweight relevance head is queried. Table 9 reports wall-clock time, peak GPU memory, and parameter count under matched hardware.

The additional cost of FocusDreamer is expected to remain moderate because the base Dreamer world model, actor, and value networks are unchanged. Moreover, return-relevance annotations are updated from sampled completed trajectories during auxiliary updates, while standard interaction requires only a single forward pass through the lightweight online relevance head.