Beyond Single-Speed Reasoning: Coordinating Fast and Slow Dynamics for Efficient World Modeling

Hongwei Wang[1*], **Yangru Huang**[2*†], **Guangyao Chen**[2], **Xu Wang**[1], **Yi Jin**[1†]

[1]Key Laboratory of Big Data and Artificial Intelligence in Transportation, Ministry of Education, the State Key Laboratory of Advanced Rail Autonomous Operation, and the School of Computer Science and Technology, Beijing Jiaotong University, Beijing , China

[2]School of Computer Science, Peking University, Beijing, China

hongwei.wang@bjtu.edu.cn, yrhuang@stu.pku.edu.cn, gy.chen@pku.edu.cn, xu.wang@bjtu.edu.cn, yjin@bjtu.edu.cn

Abstract

Model-based reinforcement learning (MBRL) enables efficient decision-making by learning predictive world models of environment dynamics. Despite recent advances, existing models often struggle to reconcile accurate short-term transitions with coherent long-term planning, especially in partially observable or long-horizon settings. We argue that this limitation often stems from modeling all transitions at a single temporal resolution, which makes it challenging to simultaneously capture fine-grained local dynamics and abstract global structures. To this end, we propose SF-RSSM (Slow-Fast Recurrent State-Space Model), a novel method that decouples short-term and long-term dynamics via a dual-branch design. The fast branch captures short-horizon transitions using residual prediction, while the slow branch models long-range dependencies with a GRU-based recurrent pathway. A distillation mechanism is developed to enable cooperation across timescales, with the slow model providing soft targets to guide the fast model. Additionally, a curiosity module encourages exploration by promoting learning in regions where the fast and slow branches exhibit divergent dynamics. Experiments on CARLA, DMControl and Atari benchmarks show that SF-RSSM outperforms strong baselines in policy performance.

Code — https://github.com/wn772/sf-rssm

Introduction

Model-based reinforcement learning (MBRL) has emerged as a promising paradigm for enhancing sample efficiency and generalization in complex decision-making tasks. By learning a predictive world model of environmental dynamics, MBRL empowers agents to reason about future trajectories without requiring interaction with the environment at each step (Hao et al. 2021; Moerland et al. 2023; Frauenknecht et al. 2025; Wang et al. 2024). Recent advancements in latent dynamics modeling have demonstrated remarkable results in visual control, long-horizon reasoning, and embodied decision-making (Hafner et al. 2019a, 2025; Lei, Schölkopf, and Posner 2022; Gupta et al. 2024; Zhang et al. 2024). These breakthroughs not only validate the efficacy of learned world models but also highlight their potential to serve as compact simulators, laying a solid foundation for advancements in planning and behavior learning.

Standard world models often suffer from model prediction errors and difficulty in modeling complex temporal dependencies (Ha and Schmidhuber 2018b; Asadi, Misra, and Littman 2018). To address the limitations, numerous approaches have been proposed to boost model expressiveness and robustness. One class of methods focuses on learning latent dynamics to abstract away irrelevant details and enable long-term prediction in compact representations (Hafner et al. 2019b), but may sacrifice short-term accuracy due to overly coarse abstractions (Starre, Loog, and Oliehoek 2022). Other approaches, including uncertainty-aware and ensemble-based methods, aim to enhance robustness under distributional shifts, but often require additional computation during the planning process (Wu, Huang, and Lv 2022; Huang et al. 2024). Meanwhile, planning-aware training objectives attempt to align model learning with downstream control, though their performance is still constrained by the trade-off between model fidelity and temporal abstraction. Collectively, these efforts advance the field but fall short of resolving the fundamental conflict between accurate short-term prediction and coherent long-term planning—two demands that often compete with each other across different time scales.

We argue that this conflict stems from ignoring the multi-scale nature of dynamics modeling: most existing approaches typically rely on a single temporal resolution, making it arduous to simultaneously capture fine-grained short-term interactions and high-level abstractions (Hafner et al. 2019b; Shaj Kumar et al. 2023). Methods optimized for long-term planning often lose temporal granularity, leading to inaccurate short-term predictions, while those prioritizing immediate transitions fail to capture the global context needed for robust long-term decision-making (Gumbsch et al. 2023; Chen, Ma, and Lin 2021). As illustrated in Figure 1, conventional models usually treat transitions uniformly, failing to separate fast-changing local dynamics

[*]These authors contributed equally.

[†]Corresponding authors.

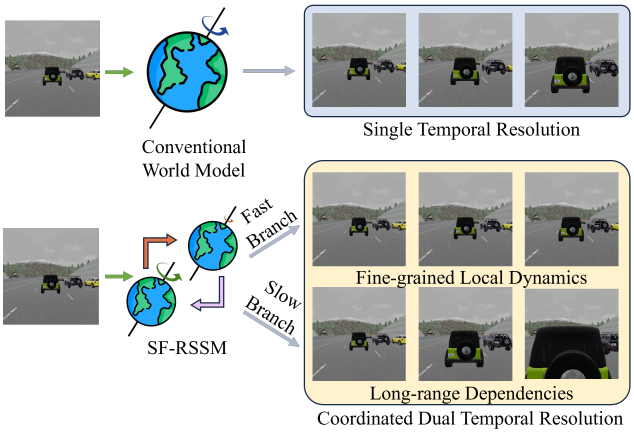

Figure 1: Comparison between conventional world model and SF-RSSM. Instead of modeling environmental dynamics at a single temporal resolution, SF-RSSM adopts a coordinated dual-branch design that effectively captures both fine-grained local dynamics and long-range dependencies.

from slow-changing global context. Such a limitation becomes particularly problematic in environments with partial observability or long horizons, where reasoning across multiple time scales is critical.

To tackle the above issues, we propose SF-RSSM (Slow-Fast Recurrent State-Space Model), a novel method improving transition accuracy by capturing both short-term and long-term dependencies in state dynamics. It introduces a dual-network structure: a fast dynamics model handling short-horizon transitions, and a slow dynamics model capturing long-range, cross-state dependencies. A mutual supervision mechanism is developed to bridge immediate reactivity and long-term reasoning, allowing the fast model to learn from the slow model's temporally abstract representations, inheriting long-term transition patterns without extra inference overhead. During imagination-based planning, only the enhanced fast model predicts future states, maintaining efficiency while preserving long-term consistency in imagined rollouts. Furthermore, we integrate a curiosity module in the behavior learning phase, which provides intrinsic rewards based on representational divergence between the fast and slow branches. This reward signal encourages exploration in regions where the fast and slow dynamics are inconsistent, prompting the agent to focus on under-explored areas. As a result, the learned policy favors sampling from states where the fast and slow models disagree, enhancing learning efficiency and ultimately leading to robust decisions.

In summary, our main contributions are as follows:

- We propose SF-RSSM, a novel model-based RL method that separates short-term dynamics and long-range dependencies into fast and slow temporal branches. We further introduce a cross-timescale distillation mechanism, enabling the fast model to inherit long-term dependencies from the slow model with minimal computational overhead.
- We develop a curiosity module in the behavior learning phase of SF-RSSM, rewarding exploration based on the divergence between fast and slow model dynamics.
- Extensive experiments on CARLA, DMControl and Atari benchmarks show that SF-RSSM outperforms state-of-the-art methods in policy performance.

Related Work

World Models from Visual Observations

World models aim to learn compact latent dynamics from high-dimensional observations, enabling agents to predict future outcomes and plan effectively. Early work by Ha and Schmidhuber (Ha and Schmidhuber 2018b) demonstrated the feasibility of learning world models from pixels by combining a variational autoencoder, a recurrent model, and a controller. Building on this idea, PlaNet (Hafner et al. 2019b) introduced a stochastic latent dynamics model that supports long-term planning in pixel-based environments. The Dreamer family of algorithms (Hafner et al. 2019a, 2020, 2025) further advanced this line by enabling end-to-end policy learning from imagined trajectories. Meanwhile, methods like CURL (Laskin, Srinivas, and Abbeel 2020) and the DrQ family (Yarats, Kostrikov, and Fergus 2021; Yarats et al. 2021) improved the quality of visual representations using contrastive learning and data augmentation. Nevertheless, these methods remain limited in their ability to capture long-term dependencies and maintain predictive accuracy, thereby driving research toward more expressive and robust visual world models.

Temporal Dynamics Modeling

Temporal dynamics modeling remains challenging due to conflicting demands for short-term precision and long-term coherence. Early efforts like Recurrent World Models (Ha and Schmidhuber 2018a) rely on RNNs but are constrained by a single temporal resolution. Latent state models such as the Dreamer series (Hafner et al. 2020) advance latent state modeling but integrate temporal scales within a unified recurrent framework, while Transformer-based methods like TWISTER (Burchi and Timofte 2025) focus on static feature alignment over explicit timescale separation. Memory-enhanced models (e.g., R2I (Henaff et al. 2023)) strengthen long-term recall but lack dedicated short/long pathways, and contrastive frameworks like TACO (Lee et al. 2023) improve representation learning without explicit long-term dynamics modeling. Our SF-RSSM seeks to mitigate these limitations with a dual-branch design: a residual fast branch captures fine-grained short transitions, while a GRU slow branch models long-range dependencies.

Preliminary

We focus on vision-based RL, where an agent interacts with an environment over discrete time steps. At each step, the agent receives a high-dimensional observation o_t, selects an action a_t, and receives a reward r_t. The goal is to learn a policy that maximizes the expected cumulative reward,

$\mathbb{E}\left[\sum_{t=1}^{T}\gamma^{t-1}r_t\right]$, where γ is the discount factor. Given the complexity of high-dimensional pixel-based observations, directly interacting with the environment can be inefficient. To address this, the agent exploits a learned world model that captures the environment's dynamics and reward structure. This model allows the agent to simulate future states and rewards, enabling more efficient planning and decision-making, and ultimately improving the policy's performance.

World Model Learning Our method builds on the DreamerV3 (Hafner et al. 2025) algorithm, using the Recurrent State Space Model (RSSM) (Hafner et al. 2019b) to model environment dynamics. The RSSM maintains a latent state s_t that evolves over time and is used for both prediction and planning. The generative process is factorized as:

$$p(o_{1:T}, r_{1:T}, s_{1:T}) = \prod_{t=1}^{T} p(o_t \mid s_t)p(r_t \mid s_t)p(s_t \mid s_{t-1}, a_{t-1}), \tag{1}$$

where $s_t = (h_t, z_t)$ represents a deterministic hidden state h_t and a stochastic latent state z_t. The hidden state h_t is updated via a recurrent network, while z_t is sampled from a learned distribution conditioned on h_t. The key components of the model are:

$$\text{RSSM}\begin{cases} \text{Sequence model: } h_t = f_\phi(h_{t-1}, z_{t-1}, a_{t-1}) \\ \text{Encoder: } z_t \sim q_\phi(z_t \mid h_t, o_t) \\ \text{Dynamics predictor: } \hat{z}_t \sim p_\phi(\hat{z}_t \mid h_t) \end{cases} \\ \begin{aligned} \text{Reward predictor: } \hat{r}_t &\sim p_\phi(\hat{r}_t \mid h_t, z_t) \\ \text{Continue predictor: } \hat{c}_t &\sim p_\phi(\hat{c}_t \mid h_t, z_t) \\ \text{Decoder: } \hat{o}_t &\sim p_\phi(\hat{o}_t \mid h_t, z_t). \end{aligned} \tag{2}$$

The sequence model updates the hidden state h_t using the previous hidden state h_{t-1}, latent representation z_{t-1}, and action a_{t-1}. The encoder converts input observations o_t into a latent variable z_t. The dynamics predictor estimates future latent states $\hat{z}_t$, while the reward and continue predictors model the expected reward and episode continuation probability. Finally, the decoder reconstructs observations from the latent state. End-to-end optimization of the model parameters $\boldsymbol{\phi}$ is done by minimizing the composite loss function:

$$\mathcal{L}(\phi) = \mathbb{E}_{q_\phi}\left[\sum_{t=1}^{T}\left(\beta_{\text{pred}}\mathcal{L}_{\text{pred}}(\phi) + \beta_{\text{dyn}}\mathcal{L}_{\text{dyn}}(\phi) + \beta_{\text{rep}}\mathcal{L}_{\text{rep}}(\phi)\right)\right], \tag{3}$$

where the prediction loss includes image reconstruction, reward estimation, and continuation prediction. The dynamics loss aligns the prior and posterior latent distributions and the representation term acts as an auxiliary regularization on latent states. We follow DreamerV3 and set $\beta_{\text{pred}} = 1$, $\beta_{\text{dyn}} = 1$, and $\beta_{\text{rep}} = 0.1$.

Behavior Learning We adopt an actor-critic framework for policy and value optimization. The critic estimates the return distribution $p_\psi(R_t^\lambda|s_t)$, where R_t^λ is the λ-return (Sharma et al. 2017), computed as a mixture of bootstrapped value and future returns:

$$\begin{aligned} \mathcal{L}(\psi) &= -\sum_{t=1}^{T}\ln p_\psi(R_t^\lambda|s_t), \\ R_t^\lambda &= r_t + \gamma c_t((1-\lambda)v_t + \lambda R_{t+1}^\lambda), \\ R_T^\lambda &= v_T. \end{aligned} \tag{4}$$

The actor is optimized to maximize expected rewards, guided by the advantage signal from the critic:

$$\begin{aligned} \mathcal{L}(\theta) = -\sum_{t=1}^{T}\text{sg}\left(\frac{R_t^\lambda - v_t}{\max(1,S)}\right)\log\pi_\theta(a_t \mid s_t) \\ + \eta\mathbb{H}\left[\pi_\theta(a_t \mid s_t)\right] \end{aligned} \tag{5}$$

where the stop-gradient operation ($\texttt{sg}$) ensures that the advantage signal is not backpropagated, and the entropy term $\eta\mathbb{H}$ encourages exploration. The scaling factor S is updated as an exponential moving average of the percentiles of R_t^λ.

Method

We propose Slow-Fast Recurrent State-Space Models (SF-RSSM) with Consistency-Driven Curiosity, a framework designed to decompose latent dynamics over multiple temporal scales for more effective world modeling and behavior learning in reinforcement learning. SF-RSSM integrates two temporal branches trained concurrently to capture both short-term and long-term dynamics. Additionally, a consistency-based curiosity module evaluates the agreement between the branches and drives exploration accordingly—encouraging exploration in unfamiliar regions with low consistency and exploitation in well-modeled regions with high consistency. Figure 2 provides an overview of the proposed model and learning process.

Dual-branch World Model Learning

In the world modeling phase, SF-RSSM consists of two primary components: a fast residual branch and a slow recurrent branch. The fast branch models short-term, fine-grained transitions, while the slow branch captures long-term, stable abstractions in the environment. These components are trained concurrently, enabling complementary learning of both local and abstract dynamics through mutual learning.

Fast Branch by Residual Prediction In the fast network, we adopt a residual prediction mechanism to predict state transitions $s_{t-1}^{\text{fast}} \rightarrow s_t^{\text{fast}}$, where the model learns the difference (residual) between consecutive states instead of predicting the absolute state (He et al. 2016). This approach is chosen because the residuals capture the short-term changes between consecutive states, allowing the fast network to focus on modeling rapid, fine-grained transitions in the environment. By predicting residuals rather than absolute states, the model can more efficiently learn to respond to immediate, local dynamics, enhancing its ability to react quickly to changes in the environment.

Specifically, the latent state s_t^{fast} that models environmental dynamics is explicitly decomposed into two complementary components: a deterministic hidden state h_t^{fast} and a

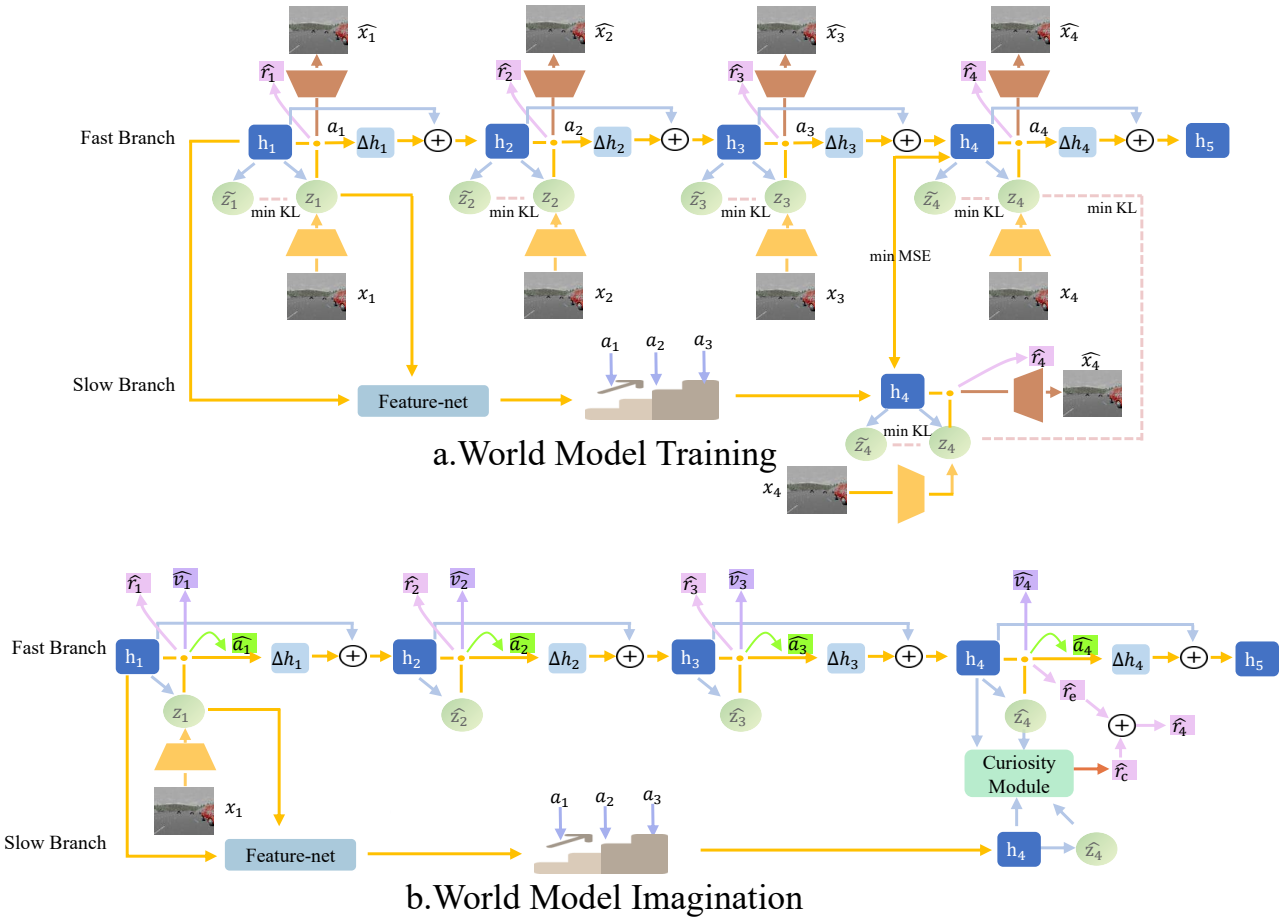

Figure 2: Overview of our method. (a) illustrates the proposed world model learning pipeline, where a dual-branch architecture decouples short-term and long-term dynamics. The fast branch models fine-grained transitions via residual prediction, while the slow branch captures coarse-grained temporal dependencies. A distillation mechanism enables cross-timescale knowledge transfer, guiding the fast branch with long-horizon signals. (b) describes the policy learning stage, which leverages imagined trajectories generated by the world model. Here, a curiosity module promotes exploration by using disagreement between fast and slow predictions as an intrinsic reward.

stochastic latent state z_t^{fast}, i.e., $s_t^{\text{fast}} = (h_t^{\text{fast}}, z_t^{\text{fast}})$. This structured decomposition enables the model to separate predictable temporal patterns from inherent environmental uncertainty, a key design choice for robust long-horizon prediction and model-based planning.

The deterministic state h_t^{fast} is updated via a residual mechanism to mitigate error accumulation in long sequences. Instead of directly predicting h_t^{fast}, we model the incremental change $\Delta h_t^{\text{fast}} = h_t^{\text{fast}} - h_{t-1}^{\text{fast}}$, leading to the recursive update:

$$h_t^{\text{fast}} = h_{t-1}^{\text{fast}} + \Delta h_t^{\text{fast}} \tag{6}$$

Here, the residual Δh_t^{fast} is predicted by a gated neural network g^{fast} that integrates past information:

$$\Delta h_t^{\text{fast}} = g^{\text{fast}}\left(a_{t-1}, h_{t-1}^{\text{fast}}, z_{t-1}^{\text{fast}}\right) \tag{7}$$

The network g^{fast} employs a GRU with gating mechanisms to regulate the magnitude of state updates. This ensures that Δh_t^{fast} captures meaningful temporal variations without excessive drift. This residual formulation reduces the learning difficulty in stable environments, where $\|\Delta h_t^{\text{fast}}\| \ll \|h_t^{\text{fast}}\|$, as the model focuses on incremental changes rather than predicting the absolute state values.

The stochastic state z_t^{fast} retains its dual-distribution design to model uncertainty, now conditioned on the residual-updated h_t^{fast}. For posterior inference (with observations), when o_t is available, z_t^{fast} is inferred from h_t^{fast} and the encoded observations $\text{Enc}(o_t)$:

$$z_t^{\text{fast}} \sim q_\phi(z_t^{\text{fast}} \mid h_t^{\text{fast}}, \text{Enc}(o_t)) \tag{8}$$

For prior imagination (without observations), z_t^{fast} is predicted solely from h_t^{fast}, enabling latent trajectory imagination for planning:

$$\hat{z}_t^{\text{fast}} \sim p_\phi(z_t^{\text{fast}} \mid h_t^{\text{fast}}) \tag{9}$$

This residual-enhanced decomposition preserves RSSM's core strengths—separating predictable dynamics from uncertainty—while improving the deterministic state's ability

to track long-range dependencies. The residual update ensures h_t^{fast} evolves smoothly, with Δh_t^{fast} focusing on task-relevant changes, thereby stabilizing multi-step prediction and model-planning. To train the fast network, we define its core dynamic regularization loss as:

$$\mathcal{L}_{\text{fast}} = D_{\text{KL}}\left(q\left(z_t^{\text{fast}} \mid h_t^{\text{fast}}, \text{Enc}(o_t)\right) \parallel p\left(z_t^{\text{fast}} \mid h_t^{\text{fast}}\right)\right) \tag{10}$$

This loss serves to regularize the stochastic component of the fast branch by penalizing divergence between two key distributions.

Slow Branch for Long-Range Dependencies The slow branch operates at a coarser temporal resolution. We set an update interval k so that the slow state $s_t^{\text{slow}} = (h_t^{\text{slow}}, z_t^{\text{slow}})$ is updated only once every k steps. This design enables the model to aggregate information over a k-step window, filtering out short-term noise and focusing on persistent, long-range patterns.

For deterministic state update, h_t^{slow} integrates the previous slow state and the sequence of actions over the last k steps to capture long-term trends:

$$h_t^{\text{slow}} = g^{\text{slow}}\left(a_{t-k:t-1}, h_{t-k}^{\text{slow}}, z_{t-k}^{\text{slow}}\right) \tag{11}$$

For the stochastic state z_t^{slow}, which captures uncertainty in long-range dependencies, we adopt the same probabilistic structure as in the fast branch. During the world model training , the posterior $z_t^{\text{slow}} \sim q_\phi(z_t^{\text{slow}} \mid h_t^{\text{slow}}, \text{Enc}(o_t))$ is used when observations are available, while during imagination, the prior $\hat{z}_t^{\text{slow}} \sim p_\phi(z_t^{\text{slow}} \mid h_t^{\text{slow}})$ guides long-horizon rollout. Both are parameterized similarly to those defined for the fast branch in Eq. 8 and Eq. 9.

Similar to the fast branch, we train the slow dynamic regularization loss $\mathcal{L}_{\text{slow}}$ with the corresponding z_t^{slow} and h_t^{slow}:

$$\mathcal{L}_{\text{slow}} = D_{\text{KL}}\left(q\left(z_t^{\text{slow}} \mid h_t^{\text{slow}}, \text{Enc}(o_t)\right) \parallel p\left(z_t^{\text{slow}} \mid h_t^{\text{slow}}\right)\right) \tag{12}$$

Cross-Timescale Interaction The fast network captures rapid state transitions, while the slow network encodes long-term dependencies in state dynamics. To transfer knowledge across time scales, we introduce a mutual supervision loss that also serves as a regularizer. Specifically, we align the deterministic states of both branches using mean squared error (MSE) and match their stochastic states via KL divergence (Gregor et al. 2018). The combined objective is denoted as $\mathcal{L}_{\text{cross}}$:

$$\mathcal{L}_{\text{cross}} = \underbrace{\|h_t^{\text{fast}} - h_t^{\text{slow}}\|_2^2}_{\mathcal{L}_{\text{MSE}}} + \underbrace{D_{\text{KL}}\left(z_t^{\text{fast}} \parallel z_t^{\text{slow}}\right)}_{\mathcal{L}_{\text{KL}}} \tag{13}$$

Training Objective In contrast to DreamerV3, which uses a single dynamics loss term between prior and posterior latent distributions, we replace this component with our proposed multi-timescale modeling loss:

$$\mathcal{L}_{\text{SF-RSSM}} = \mathcal{L}_{\text{fast}} + \mathcal{L}_{\text{slow}} + \mathcal{L}_{\text{cross}} \tag{14}$$

Curiosity Driven Behavior Learning

In the behavior learning phase, we introduce a consistency-based curiosity module that drives exploration by evaluating the agreement between predictions from the slow and fast branches (Nguyen et al. 2021; Colas et al. 2019). To reduce computational overhead during policy learning, only the fast branch is used for action selection, while the slow branch provides supervisory signals for curiosity estimation. This module encourages exploration in regions where the two branches disagree (low consistency) and promotes exploitation in well-understood regions where their predictions align (high consistency).

Specifically, we incorporate an intrinsic curiosity signal into the actor-critic framework as an auxiliary reward to guide the policy toward informative and uncertain states, which we formulate as follows:

$$r^i = Ce^{-\lambda t} \cdot d\left(\mathbf{h}_t^{\text{slow}}, \mathbf{h}_t^{\text{fast}}\right) \cdot \frac{r_{\text{max}}^e}{r_{\text{max}}^i}, \tag{15}$$

where $\mathbf{h}_t^{\text{slow}}$ and $\mathbf{h}_t^{\text{fast}}$ denote the deterministic states of the slow and fast branches, encoding long-term trends and short-term variations, respectively. $d(\cdot,\cdot)$ is the L2 distance function, quantifying the consistency gap between the two branches. C is a temperature weight scaling the intrinsic reward. $e^{-\lambda t}$ introduces exponential decay, reducing curiosity over time. r_{max}^e and r_{max}^i normalize the intrinsic reward with the maximum observed extrinsic and intrinsic rewards.

This curiosity reward is combined with the extrinsic reward r_t^e to form the total reward $\tilde{r}_t = r_t^e + r_t^i$, which drives policy optimization via λ-returns:

$$R_t^\lambda = \tilde{r}_t + \gamma c_t\left((1-\lambda)v_t + \lambda R_{t+1}^\lambda\right). \tag{16}$$

The critic predicts these λ-returns, and the actor maximizes expected returns using advantages $A_t = R_t^\lambda - v_t$. By rewarding exploration in areas where the slow and fast branches diverge, the curiosity mechanism accelerates policy learning by enhancing sample efficiency and directing exploration toward informative states, ultimately enabling more robust decision-making.

Experiments

Experimental Setup

Benchmarks and Implementation Details We evaluate our method on three widely used simulation environments: **CARLA** (Dosovitskiy et al. 2017), the **DeepMind Control Suite (DMC)** (Tassa et al. 2018), and **Atari** (Bellemare et al. 2013). In CARLA, we test the model in the `Town04` map with 64×64 downsampled images, running each episode for 1000 steps under six distinct weather settings (`Variable Weather`, `Default`, `HardRainNoon`, `MidRainSunset`, `WetCloudyNoon`, and `WetCloudySunset`) to reflect various environmental conditions. The task involves driving a vehicle smoothly and safely while avoiding collisions and excessive steering. In DMC, we evaluate the agent's ability to learn sample-efficient policies across six challenging physics-based continuous control tasks(see Tables 2), which require balance,

Method \ Task	Variable weather	Default	HardRainNoon	MidRainSunset	WetCloudyNoon	WetCloudySunset
DreamerV3	442	476	463	452	459	449
CURL	142	165	155	154	168	163
DrQ-v2	246	260	255	263	257	261
Iso-Dreamer++	252	265	267	272	280	263
TWISTER	501	532	526	548	516	535
SF-RSSM	**606**	**638**	**614**	**621**	**632**	**625**

Table 1: Performance comparison across different methods under varying weather conditions in CARLA.

Method \ Task	Hopper hop	Quadruped run	Finger turn hard	Cheetah run	Walker run	Cartpole swingup sparse
DreamerV3	352	586	962	728	757	792
CURL	353	490	928	502	360	373
DrQ-v2	365	698	952	716	539	773
Iso-Dreamer++	153	498	942	742	425	249
TWISTER	385	731	905	694	711	735
SF-RSSM	**508**	**832**	**966**	**813**	**798**	**827**

Table 2: Performance comparison of different methods in the DeepMind Control Suite.

locomotion, and precise motor control. All tasks are trained directly from high-dimensional pixel observations, with images sized at $64 \times 64 \times 3$, and agents are trained for up to 1 million steps. In Atari, we assess our method in discrete-action domains using $64 \times 64 \times 3$ RGB images from the Atari 100k benchmark, evaluating performance across six games(see Tables 3). All reported results are averaged over 5 random seeds to ensure statistical reliability. These benchmarks provide a comprehensive performance assessment across visual driving, continuous control, and discrete-action gameplay tasks, allowing us to thoroughly test the model's generalization and control capabilities.

Compared Methods SF-RSSM is evaluated in comparison with several representative visual reinforcement learning methods, including traditional model-free approaches **DrQ-v2** (Yarats et al. 2021), **CURL** (Laskin, Srinivas, and Abbeel 2020), as well as model-based methods **DreamerV3** (Hafner et al. 2025), **Iso-Dreamer++** (Pan et al. 2023), **IRIS** (Micheli, Alonso, and Fleuret 2022) and **TWISTER** (Burchi and Timofte 2025).

Comparision with State-Of-the-Art

Evaluation on CARLA Table 1 summarizes the episode returns across various weather conditions in CARLA. SF-RSSM consistently achieves the best performance across all settings, demonstrating strong adaptability to complex driving scenarios. While TWISTER performs competitively, especially under moderate conditions, it still falls behind SF-RSSM under more dynamic environments like HardRainNoon and WetCloudySunset. DreamerV3 shows stable but suboptimal performance, suggesting limitations in capturing fine-grained cues under visual disturbances. CURL and DrQ-v2 struggle in all scenarios, likely due to their limited capacity to reason over temporal structure in high-dimensional visual inputs. Iso-Dreamer++ slightly improves upon these model-free baselines but remains inferior to SF-RSSM. These results highlight the advantage of explicitly modeling multi-timescale dynamics in challenging, partially observable environments.

Evaluation on DeepMind Control Suite and Atari Tables 2 and 3 summarize mean episode returns on six DMC tasks and six Atari games, spanning locomotion/manipulation as well as dense- and sparse-reward settings. SF-RSSM ranks first on all tasks, indicating strong robustness to both fast-changing dynamics and long-horizon credit assignment. On DMC, the improvement is most pronounced on challenging high-speed locomotion tasks (e.g., Quadruped Run), where stable coordination is critical, yielding clear gains over strong baselines such as DreamerV3 and DrQ-v2. On Atari, SF-RSSM remains consistently strong across reward regimes; in particular, it improves performance on sparse-reward games (e.g., Amidar). Overall, these results support that explicitly modeling multi-timescale dynamics yields reliable benefits across diverse control and visual decision-making benchmarks.

Ablation Study

We perform an ablation study in Table 4 to evaluate the average performance of SF-RSSM and its variants across CARLA and DMC tasks, isolating the contribution of each component. Removing the slow branch leads to a significant performance drop, demonstrating its critical role in capturing long-term temporal dependencies for improved planning. Eliminating the residual prediction in the fast branch results in a reduced return, indicating that modeling short-term dynamics as residuals helps the model focus on local transitions more effectively. Without the cross-timescale distillation mechanism, episode return declines to 557, suggesting that the knowledge transfer between fast and slow branches enhances representation quality and sample efficiency. The ablated variant without the curiosity module shows reduced overall performance, suggesting that exploration driven by model disagreement contributes to better downstream policy learning. Finally, reverting to a stan-

Method \ Task	Atari Amidar	Atari Asterix	Atari Chopper Command	Atari Battle zone	Atari Frostbite	Atari Up N down
DreamerV3	94	1136	1626	12250	909	7600
CURL	142	734	1058	14870	1181	2955
Iso-Dreamer++	8	650	1600	12571	268	8402
IRIS	143	854	1565	13074	259	3546
TWISTER	184	1306	910	9920	305	7068
SF-RSSM	**191**	**1396**	**2089**	**14984**	**2283**	**9064**

Table 3: Performance comparison of different methods on Atari 100k benchmarks.

Variant	**Episode Return ↑**
SF-RSSM (Full)	**612**
w/o Slow Branch	523
w/o Residual in Fast	542
w/o Distillation	557
w/o Curiosity Module	575
Single-Scale RSSM	460

Table 4: Ablation performance of SF-RSSM. Episode returns are averaged over CARLA and DMC environments.

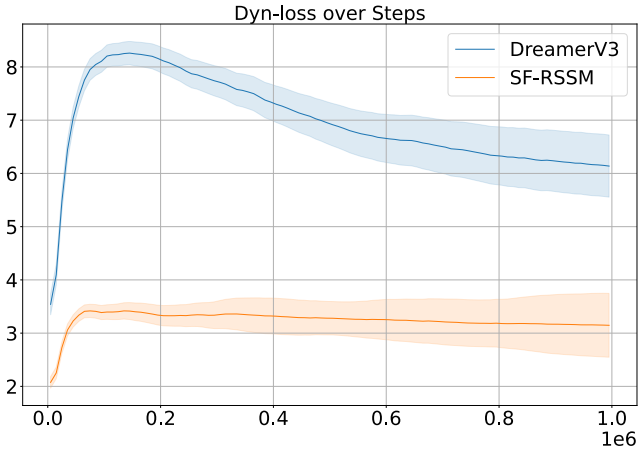

Figure 3: Comparison of dynamic loss between SF-RSSM and DreamerV3 on CARLA,where dynamic loss measures the gap between the posterior and prior latent states.

dard single-scale RSSM results in the lowest return, validating our hypothesis that a unified temporal resolution limits the model's ability to balance short- and long-term reasoning. These results collectively highlight the necessity of each module and the effectiveness of the proposed multi-timescale framework.

Further Analysis

Effect of Dual-Timescale Modeling To verify the effectiveness of dual-timescale modeling in latent dynamics, we assess latent quality by comparing the dynamics loss between the prior and posterior distributions of SF-RSSM and DreamerV3 on CARLA. As shown in Figure 3, SF-RSSM consistently yields lower dynamic loss than DreamerV3, indicating more coherent and predictable transitions. This reflects its superior imagination, where the prior alone can approximate the posterior. Such consistency stems from the dual-timescale design, which stabilizes long-term dynamics and enhances accuracy under partial observability.

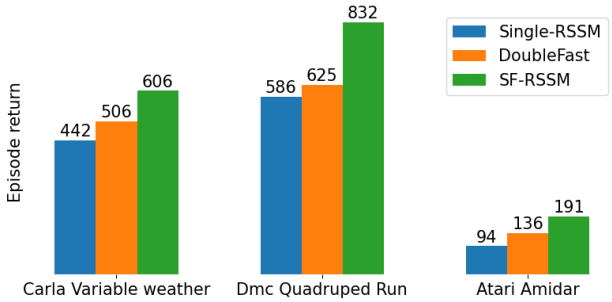

Figure 4: Performance comparison between SF-RSSM and a fast-fast dual-branch variant.

Effect of Temporal Decoupling Beyond Model Capacity To verify that SF-RSSM's performance gain does not simply stem from increased model capacity, we compare it with a dual-branch variant where both branches model only fast (short-horizon) dynamics. As shown in Figure 4, this fast-fast variant performs only slightly better than a single-branch baseline and consistently underperforms SF-RSSM on CARLA (Variable Weather), DMC Quadruped-run, and Atari Amidar. These results demonstrate that the performance gain arises not from parameter count, but from the explicit modeling of multi-timescale dynamics. The slow branch enhances temporal abstraction, yielding more stable predictions and improved decision-making.

Conclusion

We presented SF-RSSM, a novel world model that decouples short- and long-term dynamics via a dual-branch architecture with residual prediction and GRU-based modeling. To enable synergy across timescales, we introduced a distillation mechanism that guides the fast model using slow-branch knowledge, and a curiosity module that drives exploration based on their dynamic disagreement. Extensive experiments on CARLA, DMControl, and Atari benchmarks show that SF-RSSM consistently improves long-horizon prediction and policy performance. Our findings highlight the importance of multi-timescale modeling for robust and efficient decision-making in complex environments.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (62571027, 62402015); the China Railway Information Technology Group Co., Ltd. Technology Research and Development Plan (WJZGCKY-2024012, 2024K09); the Postdoctoral Fellowship Program of the China Postdoctoral Science Foundation (GZB20230024); and the China Postdoctoral Science Foundation (2024M750100).

References

Asadi, K.; Misra, D.; and Littman, M. 2018. Lipschitz continuity in model-based reinforcement learning. In *International conference on machine learning*, 264–273. PMLR.

Bellemare, M. G.; Naddaf, Y.; Veness, J.; and Bowling, M. 2013. The arcade learning environment: An evaluation platform for general agents. *Journal of artificial intelligence research*, 47: 253–279.

Burchi, M.; and Timofte, R. 2025. Learning Transformer-based World Models with Contrastive Predictive Coding. In *The Thirteenth International Conference on Learning Representations*.

Chen, Z.; Ma, Q.; and Lin, Z. 2021. Time-Aware Multi-Scale RNNs for Time Series Modeling. In *IJCAI*, 2285–2291.

Colas, C.; Fournier, P.; Chetouani, M.; Sigaud, O.; and Oudeyer, P.-Y. 2019. Curious: intrinsically motivated modular multi-goal reinforcement learning. In *International conference on machine learning*, 1331–1340. PMLR.

Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; and Koltun, V. 2017. CARLA: An open urban driving simulator. In *Conference on robot learning*, 1–16. PMLR.

Frauenknecht, B.; Subhasish, D.; Solowjow, F.; and Trimpe, S. 2025. On rollouts in model-based reinforcement learning. *arXiv preprint arXiv:2501.16918*.

Gregor, K.; Papamakarios, G.; Besse, F.; Buesing, L.; and Weber, T. 2018. Temporal difference variational auto-encoder. *arXiv preprint arXiv:1806.03107*.

Gumbsch, C.; Sajid, N.; Martius, G.; and Butz, M. V. 2023. Learning hierarchical world models with adaptive temporal abstractions from discrete latent dynamics. In *The Twelfth International Conference on Learning Representations*.

Gupta, T.; Gong, W.; Ma, C.; Pawlowski, N.; Hilmkil, A.; Scetbon, M.; Rigter, M.; Famoti, A.; Llorens, A. J.; Gao, J.; et al. 2024. The essential role of causality in foundation world models for embodied AI. *arXiv preprint arXiv:2402.06665*.

Ha, D.; and Schmidhuber, J. 2018a. Recurrent World Models Facilitate Policy Evolution. *NeurIPS Workshop on Evolution Strategies*.

Ha, D.; and Schmidhuber, J. 2018b. World models. *arXiv preprint arXiv:1803.10122*, 2(3).

Hafner, D.; Lillicrap, T.; Ba, J.; and Norouzi, M. 2019a. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*.

Hafner, D.; Lillicrap, T.; Fischer, I.; Villegas, R.; Ha, D.; Lee, H.; and Davidson, J. 2019b. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, 2555–2565. PMLR.

Hafner, D.; Lillicrap, T.; Norouzi, M.; and Ba, J. 2020. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*.

Hafner, D.; Pasukonis, J.; Ba, J.; and Lillicrap, T. 2025. Mastering diverse control tasks through world models. *Nature*, 1–7.

Hao, J.; Yuan, Y.; Wang, C.; and Wang, Z. 2021. ED2: Environment Dynamics Decomposition World Models for Continuous Control. *arXiv preprint arXiv:2112.02817*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Henaff, O. J.; Laskin, M.; McAllister, R.; Srinivasan, L.; Abbeel, P.; Efros, A. A.; and Finn, C. 2023. Mastering Memory Tasks with World Models. *Advances in Neural Information Processing Systems*, 36.

Huang, W.; Cui, Y.; Li, H.; and Wu, X. 2024. Practical probabilistic model-based reinforcement learning by integrating dropout uncertainty and trajectory sampling. *IEEE Transactions on Neural Networks and Learning Systems*.

Laskin, M.; Srinivas, A.; and Abbeel, P. 2020. Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning*, 5639–5650. PMLR.

Lee, K.; Ahn, S.; Pfaff, T.; Levine, S.; and Finn, C. 2023. Temporal Action-Driven Contrastive Learning for Visual Reinforcement Learning. In *International Conference on Machine Learning*, 19782–19795. PMLR.

Lei, A.; Schölkopf, B.; and Posner, I. 2022. Variational causal dynamics: Discovering modular world models from interventions. *arXiv preprint arXiv:2206.11131*.

Micheli, V.; Alonso, E.; and Fleuret, F. 2022. Transformers are sample-efficient world models. *arXiv preprint arXiv:2209.00588*.

Moerland, T. M.; Broekens, J.; Plaat, A.; Jonker, C. M.; et al. 2023. Model-based reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 16(1): 1–118.

Nguyen, T.; Luu, T. M.; Vu, T.; and Yoo, C. D. 2021. Sample-efficient reinforcement learning representation learning with curiosity contrastive forward dynamics model. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3471–3477. IEEE.

Pan, M.; Zhu, X.; Zheng, Y.; Wang, Y.; and Yang, X. 2023. Model-Based Reinforcement Learning With Isolated Imaginations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5): 2788–2803.

Shaj Kumar, V.; Gholam Zadeh, S.; Demir, O.; Douat, L.; and Neumann, G. 2023. Multi time scale world models. *Advances in Neural Information Processing Systems*, 36: 26764–26775.

Sharma, S.; Ramesh, S.; Ravindran, B.; et al. 2017. Learning to mix n-step returns: Generalizing lambda-returns for deep reinforcement learning. *arXiv preprint arXiv:1705.07445*.

Starre, R. A.; Loog, M.; and Oliehoek, F. A. 2022. Model-based reinforcement learning with state abstraction: A survey. In *Benelux Conference on Artificial Intelligence*, 133–148. Springer.

Tassa, Y.; Doron, Y.; Muldal, A.; Erez, T.; Li, Y.; Casas, D. d. L.; Budden, D.; Abdolmaleki, A.; Merel, J.; Lefrancq, A.; et al. 2018. Deepmind control suite. *arXiv preprint arXiv:1801.00690*.

Wang, W.; Dusparic, I.; Shi, Y.; Zhang, K.; and Cahill, V. 2024. Drama: Mamba-enabled model-based reinforcement learning is sample and parameter efficient. *arXiv preprint arXiv:2410.08893*.

Wu, J.; Huang, Z.; and Lv, C. 2022. Uncertainty-aware model-based reinforcement learning: Methodology and application in autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 8(1): 194–203.

Yarats, D.; Fergus, R.; Lazaric, A.; and Pinto, L. 2021. Mastering visual continuous control: Improved data-augmented reinforcement learning. *arXiv preprint arXiv:2107.09645*.

Yarats, D.; Kostrikov, I.; and Fergus, R. 2021. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International conference on learning representations*.

Zhang, Z.; Chen, R.; Ye, J.; Sun, Y.; Wang, P.; Pang, J.; Li, K.; Liu, T.; Lin, H.; Yu, Y.; et al. 2024. WHALE: Towards Generalizable and Scalable World Models for Embodied Decision-making. *arXiv preprint arXiv:2411.05619*.